



UNIVERSIDAD POLITÉCNICA SALESIANA

SEDE CUENCA

CARRERA DE BIOMEDICINA

**PROCESAMIENTO DE IMÁGENES PARA DETECCIÓN DE OSTEOARTRITIS DE
RODILLA UTILIZANDO IA**

Trabajo de titulación previo a la obtención del
título de Ingeniera Biomédica

AUTORA: TATIANA DOLORES CÁRDENAS GUARACA

TUTOR: ING. VLADIMIR ESPARTACO ROBLES BYKBAEV, PhD.

Cuenca - Ecuador

2024

CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE TITULACIÓN

Yo, Tatiana Dolores Cárdenas Guaraca con documento de identificación N° 0107372104, manifiesto que:

Soy la autora y responsable del presente trabajo; y, autorizo a que sin fines de lucro la Universidad Politécnica Salesiana pueda usar, difundir, reproducir o publicar de manera total o parcial el presente trabajo de titulación.

Cuenca, 15 de febrero del 2024

Atentamente,



Tatiana Dolores Cárdenas Guaraca

0107372104

**CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE
TITULACIÓN A LA UNIVERSIDAD POLITÉCNICA SALESIANA**

Yo, Tatiana Dolores Cárdenas Guaraca con documento de identificación N° 0107372104, expreso mi voluntad y por medio del presente documento cedo a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que soy autora del Proyecto técnico: “Procesamiento de imágenes para detección de osteoartritis de rodilla utilizando IA”, el cual ha sido desarrollado para optar por el título de: Ingeniera Biomédica, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En concordancia con lo manifestado, suscribo este documento en el momento que hago la entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

Cuenca, 15 de febrero del 2024

Atentamente,



Tatiana Dolores Cárdenas Guaraca

0107372104

CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN

Yo, Vladimir Espartaco Robles Bykbaev con documento de identificación N° 0300991817, docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: PROCESAMIENTO DE IMÁGENES PARA DETECCIÓN DE OSTEOARTRITIS DE RODILLA UTILIZANDO IA, realizado por Tatiana Dolores Cárdenas Guaraca con documento de identificación N° 0107372104, obteniendo como resultado final el trabajo de titulación bajo la opción Proyecto técnico que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Cuenca, 15 de febrero del 2024

Atentamente,



Ing. Vladimir Espartaco Robles Bykbaev, PhD.

0300991817

DEDICATORIA Y AGRADECIMIENTO

Quiero dedicar este proyecto primeramente a Dios y a quienes han sido mi cimiento en el camino hacia mi objetivo de ser Ingeniera Biomédica, muchos de ellos sin tener muy claro en que podría ejercer. Agradezco profundamente a los Ingenieros Juan Pablo Bermeo, Edmound Geround y Vladimir Robles por su soporte en este trabajo. Finalmente agradezco a la Universidad Politécnica Salesiana y a sus colaboradores por su constante apoyo y permitirme cumplir muchos sueños siendo estudiante. Cada día de estudio ha sido una aventura gracias a todos ustedes.

¡Gracias de corazón!

ÍNDICE DE CONTENIDO

1. Resumen	8
2. Abstract	8
3. Introducción	9
4. Problema.....	9
5. Objetivos Generales y Específicos	10
Objetivo general:	10
Objetivos específicos:	10
6. Marco Teórico - Revisión de la literatura o fundamentos teóricos.	11
7. Marco metodológico	17
8. Resultados	18
9. Cronograma.....	35
10. Presupuesto.....	36
11. Conclusiones	36
12. Recomendaciones.....	37
13. Referencias bibliográficas	37
14. Anexos.....	39

ÍNDICE DE FIGURAS

Figura 1. Clasificación radiológica de osteoartritis.....	12
Figura 2. Carga de imágenes y procesamiento en R.	18
Figura 3. Diagrama de red neuronal artificial.	19
Figura 4. Código de Modelo 1	20
Figura 5. Matriz de confusión de modelo 1	20
Figura 6. Código de Modelo 2	21
Figura 7. Matriz de confusión de modelo 2	23
Figura 8. Código de modelo 3	24
Figura 9. Matriz de confusión 3	24
Figura 10. Pestaña “Detección” en propuesta de diseño de interfaz.....	25
Figura 11. Creación de cuenta de usuario.	26
Figura 12. Inicio de sesión para los usuarios registrados.....	26
Figura 13. Programación en angular.	27
Figura 14. Pestaña “Home” de la interfaz.	28
Figura 15. Pestaña de “Detección” en la interfaz.....	29
Figura 16. Uso de Firebase para almacenar datos.	30
Figura 17. Pestaña de “Documentos” de la interfaz.....	30
Figura 18. Funciones de búsqueda	31
Figura 19. Código para interoperabilidad en los sistemas	32
Figura 20. Pruebas realizadas con Postman para verificar funcionalidad.....	32
Figura 21. Formulario de validación del proyecto.	33
Figura 22. Ejemplo de resultado del formulario.	34

ÍNDICE DE TABLAS

Tabla 1. Cronograma de actividades	35
Tabla 2. Presupuesto del proyecto.....	36

Resumen

El estudio se enfoca en la implementación de la Inteligencia Artificial (IA) para la detección de la osteoartritis de rodilla, un trastorno degenerativo que ocasiona dolor y restricciones en el movimiento. Su propósito principal consiste en desarrollar métodos de aprendizaje automático que permitan identificar esta enfermedad. Para ello, se emplea la IA para examinar imágenes médicas, en este caso radiografías, además, se propone una interfaz diseñada para simplificar la gestión de la información por parte del personal médico. La detección temprana es esencial en la prevención del avance de la osteoartritis, lo que aporta a la importancia de esta herramienta como apoyo para dicha detección.

En la investigación, se exploran diversos modelos de IA utilizando una base de datos de rayos X accesible. Se desarrolla una interfaz que simplifique la validación por parte de expertos médicos, lo que implica una adaptación cuidadosa del modelo para su integración efectiva en dicha interfaz, considerando la diversidad de entornos de programación.

Palabras clave: Radiología, Interfaz, Detección, Osteoartritis, Rehabilitación, Inteligencia artificial

Abstract

The study focuses on the implementation of Artificial Intelligence (AI) for the detection of knee osteoarthritis, a degenerative disorder that causes pain and movement restrictions. Its main purpose is to develop machine learning methods that allow the identification of this disease. To achieve this, AI is used to examine medical images, in this case, X-rays, and a user interface designed to simplify information management by healthcare personnel is proposed. Early detection is essential in preventing the progression of osteoarthritis, which underscores the importance of this tool as support for such detection.

The research explores various AI models using an accessible X-ray database. An interface is developed to streamline validation by medical experts, which involves careful adaptation of the model for effective integration into the interface, considering the diversity of programming environments.

Keywords: Radiology, Interface, Detection, Osteoarthritis, Rehabilitation, Artificial Intelligence

Introducción

La detección temprana de la osteoartritis de rodilla mediante el uso de IA representa un hito significativo en el ámbito de la atención médica contemporánea. Según la Organización Mundial de la Salud (OMS), aumento el número de personas afectadas por la osteoartritis a nivel global en 2019, alcanzando los 528 millones, refleja una creciente preocupación en la salud pública, especialmente conforme la población envejece. Este incremento en la incidencia de la enfermedad subraya la necesidad imperativa de soluciones innovadoras para abordar eficazmente este desafío de salud. Se desarrolla un sistema con inteligencia artificial capaz de identificar y clasificar la osteoartritis de rodilla en sus etapas iniciales no solo es una oportunidad revolucionaria, sino también una medida crucial para intervenir de manera efectiva y menos costosa. Los beneficios derivados de esta detección temprana son múltiples y abarcan tanto al sistema de atención médica como a los pacientes en su totalidad.

Uno de los aspectos más destacados de la detección de la osteoartritis de rodilla es el buen impacto que puede generar en la calidad de vida de los pacientes. Al identificar la enfermedad, se pueden implementar estrategias de manejo del dolor, fisioterapia y cambios en los hábitos que no solo ayuden a preservar la función de la articulación, sino también a reducir el malestar experimentado por los pacientes. Esta intervención anticipada podría permitir a los pacientes mantener su independencia y calidad de vida durante más tiempo, evitando así la discapacidad prematura y la necesidad de intervenciones médicas invasivas. La implementación de algoritmos avanzados de inteligencia artificial permite detectar cambios sutiles en las radiografías de las articulaciones, lo que es esencial para una intervención oportuna. Además, brinda a los médicos una herramienta adicional en su práctica clínica, lo que les permite realizar diagnósticos más precisos y tomar decisiones informadas sobre el tratamiento. Esta capacidad de la IA para optimizar el tiempo y los recursos del personal sanitario es especialmente relevante en un contexto de aumento de la carga de trabajo médico.

Problema

La detección de problemas en la rodilla es esencial debido a que existen patologías como la osteoartritis, una enfermedad que causa dolor, rigidez y problemas de movimiento en la articulación. El diagnóstico de la osteoartritis de rodilla se basa en la evaluación de síntomas y radiografías. La clasificación de Kellgren-Lawrence (KL) que es un sistema médico que evalúa la gravedad de la osteoartritis mediante radiografías de articulaciones, consta de 5 etapas, 4 principalmente para evaluar la osteoartritis: 0 (sin cambios), 1 (posible estrechamiento del espacio articular), 2 (ligero estrechamiento), 3 (moderado estrechamiento) y 4 (estrechamiento severo con deformidades óseas). En las primeras etapas, los pacientes que padecen osteoartritis pueden no tener síntomas notables, pero a medida que avanza la enfermedad, pueden experimentar problemas de movilidad y, en algunos casos, necesitar cirugía (Yaseen, 2023). Según un análisis realizado por Madaleno et al., (2018), la osteoartritis de rodilla no se limita a una edad específica; puede afectar a personas mayores de 50 años, así como a deportistas y jóvenes que

participan en actividades físicas intensas y repetitivas. Los deportistas enfrentan desafíos significativos, ya que esta afección puede limitar su participación en deportes y afectar su rendimiento. Por lo tanto, detectarla tempranamente es esencial para prevenir su progresión y mejorar la calidad de vida de los pacientes. Según la OMS en 2019, cerca de 528 millones de personas en todo el mundo tenían osteoartritis, un aumento del 113% desde 1990, la prevalencia general de osteoartritis de rodilla sintomática diagnosticada por un médico en adultos osciló entre 1,55% en Perú y 7,4% en Ecuador. La prevalencia total de osteoartritis de rodilla radiográfica de grado ≥ 2 fue de 22% para los ≥ 39 años en Brasil y 25,5% entre los ≥ 40 años en México (Medscape, 2022). Se obtiene una detección temprana y utilizando la inteligencia artificial para analizar imágenes y detectar problemas en la articulación de la rodilla es trascendental, lo que podría mejorar la vida de los pacientes y buscar una interfaz de fácil manejo para el médico.

La IA es una tecnología que permite a las computadoras aprender y crear decisiones basadas en datos, en el contexto médico, la IA se utiliza para procesar imágenes médicas, como radiografías, y ayudar en el diagnóstico temprano de enfermedades. Esto significa que la IA podría ayudar a los médicos a identificar la enfermedad en sus primeras etapas, incluso antes de que los pacientes experimenten síntomas notables. La detección temprana de la osteoartritis de rodilla es fundamental porque permite intervenir antes de que la enfermedad avance y cause más daño, con la ayuda de la IA los médicos pueden identificar pequeños cambios en las radiografías que indican la presencia de la enfermedad, lo que significa que los pacientes pueden recibir tratamiento más rápido. Además, la IA puede ayudar a predecir cómo progresará la enfermedad en cada paciente, lo que permite una atención personalizada, cuando esta enfermedad se detecta en sus etapas iniciales, los tratamientos pueden ser menos invasivos y costosos para los pacientes (Gore, 2020).

Esta iniciativa se delimita en un sistema de IA capaz de analizar imágenes médicas, como radiografías, con una interfaz diseñada para manejar fácilmente a los médicos, incluyendo fisioterapeutas, se espera que esta solución beneficie tanto a los pacientes, proporcionando tratamientos más rápidos e invasivos, como a los profesionales de la salud, facilitando diagnósticos más precisos y personalizados.

Objetivos Generales y Específicos

Objetivo general:

Desarrollar una interfaz basada en técnicas de inteligencia artificial para facilitar el diagnóstico de osteoartritis de rodilla siendo una herramienta de apoyo en la detección temprana.

Objetivos específicos:

- Analizar y seleccionar una base de imágenes médicas que contenga datos relevantes para la identificación de la osteoartritis de rodilla.
- Seleccionar, ejecutar y evaluar un algoritmo de clasificación que permita determinar si un paciente tiene o no osteoartritis de rodilla basándose en las imágenes médicas.

- Diseñar y validar una interfaz web que permita a los médicos interactuar con el sistema proporcionando acceso a la información almacenada.

Marco Teórico - Revisión de la literatura o fundamentos teóricos.

La osteoartritis de la rodilla, una afección deteriorante que impacta la articulación, puede manifestarse con síntomas como dolor, rigidez y problemas de movilidad. Sin embargo, hay un rayo de esperanza en el horizonte: la IA está emergiendo como una herramienta valiosa en la detección de esta condición a través del análisis de imágenes de rayos X. (Gore, 2020). El equipo editorial de MedImaging ha difundido un artículo sobre un software de detección de osteoartritis de rodilla mediante inteligencia artificial. Este software se basa en la red neuronal desarrollada por investigadores de la Universidad de Jyväskylä, (Jyväskylä, Finlandia) el método de inteligencia artificial puede identificar la osteoartritis de rodilla en sus etapas iniciales utilizando imágenes de rayos X. Este enfoque se ha entrenado para reconocer una característica radiológica específica asociada con la osteoartritis en los rayos X. En una evaluación de su fiabilidad, se encontró que este método coincidía con el diagnóstico realizado por médicos en el 87% de los casos. (Medimaging, 2023).

El uso de IA para la detección de enfermedades a través de imágenes médicas está en constante crecimiento. Google Health (2023) está dedicando esfuerzos a desarrollar modelos de IA con el fin de proporcionar asistencia a especialistas médicos en el proceso de diagnóstico de enfermedades.

La radiología, como especialidad médica, ha presenciado avances técnicos significativos en el pasado, que han influenciado cómo se usan las imágenes médicas. La inteligencia artificial (IA) representa potencialmente otro de estos avances, que podría introducir cambios fundamentales en la práctica de la radiología. (Gore, 2020).

En el marco teórico se pretende abarcar más a detalle los siguientes temas:

1. Osteoartritis de rodilla:

La osteoartritis de la rodilla es un trastorno articular caracterizada por el desgaste y la disminución del cartílago, así como por cambios en el hueso que yace debajo de este. Los cambios histológicos que afectan la calidad y el grosor del cartílago son signos distintivos de esta condición. A medida que el cartílago se reduce en las articulaciones, el hueso subcondral puede aumentar de tamaño y pueden desarrollarse crecimientos óseos en los bordes de las superficies articulares. El dolor, siendo el síntoma más prominente, limita la movilidad en los pacientes que tienen osteoartritis. (Physiotutors, 2021). Se tienen los siguientes grados:

Cuadro IV. Clasificación radiológica de osteoartritis (Kellgren y Lawrence)	
Grado	Características
0	Normal
1	(Dudoso) Dudoso estrechamiento del espacio articular Posible osteofitos
2	(Leve) Posible disminución del espacio articular Osteofitos
3	(Moderado) Estrechamiento del espacio articular Osteofitos Leve esclerosis Posible deformidad de los extremos de los huesos
4	(Grave) Marcado disminución del espacio articular Abundantes osteofitos Esclerosis grave Deformidad de los extremos de los huesos

Figura 1. Clasificación radiológica de osteoartritis.

García San Román, F. J., & Calcerrada Díaz-Santos, N. (2006). [Imagen]. En Guía de Práctica Clínica del manejo del paciente con osteoartritis de rodilla en Atención Primaria. Unidad de Evaluación de Tecnologías Sanitarias (UETS), Área de Investigación y Estudios Sanitarios. Agencia Laín Entralgo.

2. Inteligencia Artificial en la medicina

La integración de la IA en el área médica ofrece la posibilidad de proporcionar diagnósticos con mayor precisión, mejorar el monitoreo de pacientes crónicos y acelerar el desarrollo de vacunas. Además de su capacidad para mejorar la precisión diagnóstica y la detección de enfermedades, la IA puede agilizar la atención médica, ayudar a la investigación y respaldar iniciativas de salud pública. (Organización Mundial de la Salud, 2021).

3. Algoritmos de clasificación en IA

Los algoritmos de clasificación son una categoría de algoritmos de aprendizaje automático diseñados para clasificar datos o predecir la clase a la que pertenece una entrada determinada. Estos algoritmos trabajan con un conjunto de datos de entrenamiento, donde aprenden a asociar características específicas con clases particulares. Son útiles en situaciones en las que el resultado esperado puede ser uno de los resultados posibles. (The Black Box Lab, 2022).

Arquitectura de Redes Neuronales: Según Javid (2021), la arquitectura de la red neuronal implica el arreglo de neuronas para formar capas y el esquema de conexión entre y dentro de las capas. Se clasifica en arquitecturas de avance y retroalimentación, con redes de avance proporcionando un flujo de señal unidireccional, y las redes de retroalimentación permitiendo que las señales fluyan en ambas direcciones. Además, Javid (2021) destaca que una tesis doctoral se enfoca en desarrollar nuevas arquitecturas de redes neuronales, considerando limitaciones prácticas, y propone un enfoque de entrenamiento por capas para redes neuronales multicapa para garantizar una reducción de la pérdida de entrenamiento a medida que la red se vuelve más profunda.

Funciones de Activación: Son un componente crucial de la arquitectura de las redes neuronales, ya que introducen no linealidades, permitiendo que la red aprenda patrones complejos en los datos. Transforman la señal de entrada de un nodo en una señal de salida, lo que permite que la red modele relaciones no lineales entre las salidas y entradas. En una red neuronal, todas las capas ocultas suelen usar la misma función de activación, mientras que la de salida típicamente utiliza una función de activación diferente, dependiendo del objetivo específico o del tipo de predicción realizada por el modelo, la elección de esta es esencial para entrenar redes neuronales que generalicen bien, y existen varios tipos de funciones de activación, como la tangente hiperbólica, sigmoide, ReLU y softmax, cada una sirviendo para diferentes propósitos dentro de diferentes capas de una red y en diversos casos de uso. La selección de la función de activación adecuada es importante para el entrenamiento efectivo de redes neuronales. (Ali, 2023).

Algoritmos de Entrenamiento: Alzubaidi et al. (2021) destacan que la capacitación sobre modelos de redes neuronales implica la aplicación de diversos algoritmos y técnicas con el propósito de mejorar su rendimiento, disminuir el costo computacional y hacer frente a desafíos como el sobreajuste. En este sentido, se emplean estrategias como el descenso de gradiente mediante retropropagación, patrones de esparcimiento y seguimiento del proceso de aprendizaje. Por ejemplo, la técnica estándar de aprendizaje consiste en la inicialización aleatoria de los pesos y la aplicación del descenso de gradiente mediante retropropagación, pero se reconoce que este método puede generar soluciones subóptimas para redes profundas con múltiples capas ocultas. Para abordar esto, el entrenamiento de modelos dispersos emerge como una alternativa prometedora para reducir el costo computacional de las redes neuronales, especialmente aquellas con un exceso de parámetros, manteniendo al mismo tiempo sus ventajas de generalización. Además, se recurre a otras técnicas como la eliminación aleatoria, el aumento de datos y los modelos de atención con el fin de mejorar el proceso de capacitación y prevenir el sobreajuste. Estas estrategias contribuyen de manera significativa a los esfuerzos continuos destinados a mejorar la eficiencia y efectividad de los algoritmos de capacitación para modelos de redes neuronales.

Datos de entrada y salida: Huang y Kuo (2019) explican que los elementos de entrada de un modelo de red neuronal son adaptados según la aplicación específica y las características de los datos subyacentes. En el ámbito del reconocimiento de imágenes, la entrada suele consistir en valores de los píxeles, mientras que la salida de la capa resultante ofrece los resultados de clasificación correspondientes. En diferentes contextos de aplicación, las redes neuronales se usan con datos de salida y entrada. Por ejemplo, en el ámbito de la energía renovable, se implementa un modelo de red neuronal profunda conocido como PVPNet para realizar predicciones a corto plazo sobre la potencia fotovoltaica. En este caso, los datos de entrada incluyen información meteorológica y registros históricos del rendimiento del sistema fotovoltaico, mientras que la salida se centra en la predicción de la potencia generada. Por otro lado, en el control de procesos industriales, se proponen diversas estructuras de actor-critic con el fin de lograr un control óptimo basado en datos de salida y entrada en sistemas no lineales desconocidos. Aquí, los datos de salida y entrada son categorizados mediante una red neuronal artificial inhibitoria de derivación (SIANN), y se definen diferentes métricas de rendimiento para cada categoría.

Evaluación del Modelo: Zapeta Hernández et al. (2022) señalan que la evaluación de modelos de redes neuronales se lleva a cabo mediante métodos tales como la división de datos de entrenamiento, validación y prueba. En este proceso, se emplean métricas de rendimiento como precisión, exactitud y exhaustividad para medir la efectividad del modelo. La selección de métricas específicas está determinada por la naturaleza de la aplicación y la clase de datos en consideración. Por ejemplo, en el contexto de la clasificación de imágenes usando redes neuronales convolucionales, es común utilizar métricas como precisión y exhaustividad para evaluar el desempeño del modelo.

Tipos de procesamiento:

Otras características relevantes de los modelos de redes neuronales pueden incluir el utilizar de técnicas de procesamiento en imágenes como escala de grises y de reducción de dimensionalidad como PCA, aumento de datos y aprendizaje por transferencia para mejorar el rendimiento del modelo y la generalización a nuevos datos.

Palafox-Palafox et al. (2023) explican que el Análisis de Componentes Principales (PCA) es una técnica que se usa para reducir la complejidad de los datos mediante la disminución de su dimensionalidad. Su objetivo es extraer las características más importantes de un conjunto de datos y reducir el número de variables presentes en él. Esta técnica se aplica en el procesamiento de imágenes para reducir la cantidad de píxeles, lo que simplifica la imagen. Del mismo modo, se emplea en el análisis de datos para reducir la complejidad del análisis al disminuir la cantidad de variables presentes en el conjunto de datos.

4. Evaluación de algoritmos de IA

Los algoritmos de inteligencia artificial son secuencias de instrucciones diseñadas para dirigir a una computadora en la ejecución de cálculos específicos. La inteligencia artificial utiliza estos algoritmos para crear sistemas que aprenden de la experiencia, se adaptan a nuevas situaciones y realizan tareas de manera similar a los seres humanos. El uso de la inteligencia artificial plantea cuestiones éticas, como la recopilación y utilización poco éticas de los datos de salud, sesgos en los algoritmos y los riesgos para la seguridad del paciente, la ciberseguridad y el medio ambiente. (Organización Mundial de la Salud, 2021).

5. Base de datos

Se ha escogido el conjunto de imágenes "Knee OA - v2", que fue exportado a través de roboflow.com en septiembre de 2023, consiste en 21,631 imágenes de rodillas afectadas por osteoartritis. Roboflow es una plataforma especializada en visión por computadora que permite la colaboración, organización y mejora de conjuntos de datos, lo que facilita el trabajo con conjuntos de imágenes como este.

6. Interfaz de usuario

Lenguajes de programación: Para el presente proyecto se debe tener en cuenta los conceptos básicos de los siguientes lenguajes de programación:

JavaScript: Es un lenguaje de programación ampliamente utilizado en el desarrollo web. Se utiliza en el front-end para crear interactividad en las páginas web, permitiendo a los desarrolladores crear contenido dinámico (Ardila Muñoz, J.Y., & Molina Sosa, D.A., 2023).

R: R es un lenguaje de programación y un entorno de software utilizado para el análisis estadístico y la visualización de datos. Se emplea con frecuencia en el back-end para tareas de procesamiento y análisis de datos (Logroño-Naranjo, S.I., López-Paredes, C.R., Estrada-Brito, N.A., & Vásquez-Núñez, V.A., 2021).

JSON (JavaScript Object Notation): JSON es un formato ligero de intercambio de datos que se utiliza comúnmente para transmitir datos entre un servidor y un cliente en aplicaciones web (Mazaeda, R., de la Fuente López, E., González, J., & Moya de la Torre, E.J., 2022).

Para poder desarrollar la interfaz se requieren de programas como, por ejemplo:

Firebase, según Hili & Oliveira (2022), es una plataforma de desarrollo de aplicaciones

móviles y web que ofrece una amplia gama de herramientas y servicios para la creación de aplicaciones de alta calidad.

Angular, de acuerdo con Goh, Ho, & Abas (2022), es un framework de JavaScript de código abierto diseñado específicamente para el desarrollo de aplicaciones web de una sola página (SPA), facilitando la creación de aplicaciones web dinámicas y escalables en el lado del cliente al proporcionar una estructura sólida.

R, según García-Quiroga et al. (2022), es un lenguaje de programación y un entorno de software usado para el análisis estadístico y la visualización de datos, y es una herramienta muy utilizada en la investigación académica y en la industria para el análisis de datos y la creación de modelos estadísticos.

Swagger, según Saioc & Vlad (2022), es una herramienta de código abierto que facilita el diseño, construcción, documentación y consumo de servicios web RESTful. Ofrece una interfaz de usuario que permite interactuar con los servicios web, además de permitir a los desarrolladores generar documentación y código de cliente para dichos servicios web.

Postman, también mencionado por Saioc & Vlad (2022), es una herramienta de colaboración de desarrollo de API que permite a los desarrolladores probar, documentar y compartir servicios web. Proporciona una interfaz de usuario para enviar solicitudes HTTP a los servicios web y visualizar las respuestas recibidas.

Además, es necesario tener claro ciertos conceptos como los siguientes:

Front-end, según Atefi et al. (2020), hace referencia a la parte visible de una aplicación o sitio web, con la que el usuario interactúa directamente. Esto incluye elementos como la interfaz de usuario, el diseño, la navegación y la interactividad.

Back-end, también mencionado por Atefi et al. (2020), se refiere a la parte de una aplicación o sitio web que no es visible para el usuario final. Aquí se encuentra la lógica de negocio, la gestión de bases de datos, la autenticación de usuarios y la seguridad.

Los métodos HTTP más comunes son GET, POST, PUT, DELETE, HEAD, OPTIONS y PATCH, según Martínez & Cecilia (2020). GET se usa para solicitar datos de un recurso, POST para enviar datos a un recurso para su procesamiento, PUT para actualizar un existente, DELETE para eliminar uno, HEAD para solicitar solo los encabezados de un recurso, OPTIONS para obtener información sobre los métodos HTTP admitidos por un recurso y PATCH para actualizar parcialmente un existente.

Interoperabilidad, según Vargas & Cecilia (2019), se refiere a la capacidad de diferentes sistemas o dispositivos para comunicarse y trabajar juntos de manera efectiva. La interoperabilidad se puede lograr mediante la definición de normas y estándares comunes, así como mediante la creación de arquitecturas y modelos de referencia que permitan la integración y la comunicación entre sistemas y dispositivos.

Marco metodológico

El proyecto adopta un enfoque descriptivo para evaluar tanto la utilidad como la satisfacción derivada del uso de la interfaz. En la fase inicial, conocida como la validación de la funcionalidad del dispositivo, se emplea una base de datos de imágenes de articulaciones con diversos grados de osteoartritis. El sistema de detección se entrena utilizando esta base de datos y se verifica su desempeño comparando los diagnósticos con los de un grupo de expertos registrados en otra base de datos.

Además de validar el rendimiento del sistema, es crucial evaluar la facilidad de uso de la interfaz. Para ello, se realizará una encuesta a usuarios del ámbito médico, centrada en aspectos como la claridad de las instrucciones, la facilidad de uso de las funciones y la utilidad del proyecto. Este enfoque se elige debido a las limitaciones para realizar pruebas experimentales con pacientes, dado que el dispositivo no es físico.

La solución propuesta consta de varias etapas:

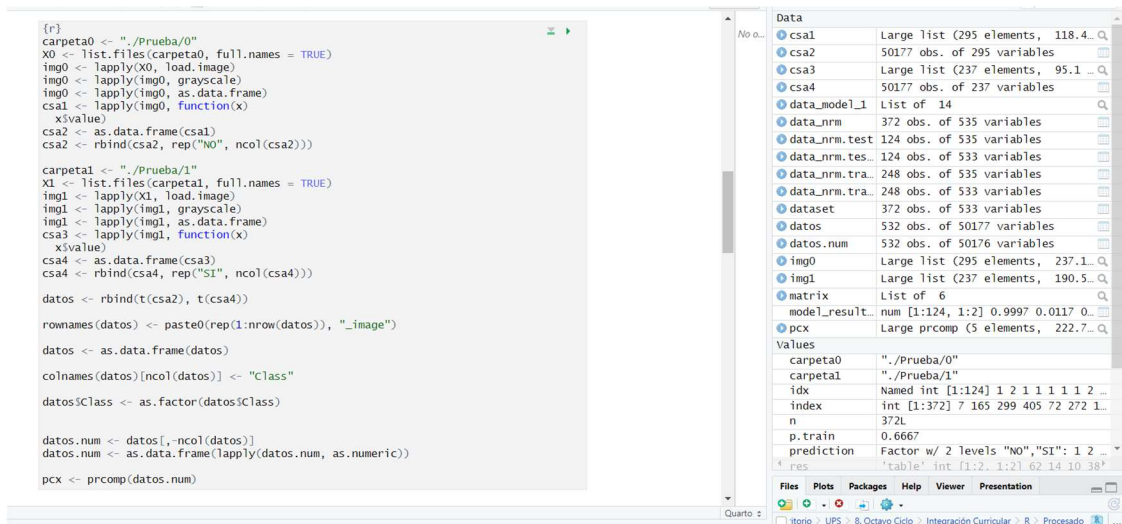
1. **Ingreso de Imágenes:** Búsqueda y carga de imágenes como punto de inicio del proceso.
2. **Procesamiento de Imágenes:** Convertir a escala de grises y realizar una Aplicación de PCA para reducir la dimensionalidad de las imágenes.
3. **Entrenamiento del Modelo:** Realización del entrenamiento del modelo en R utilizando la red neuronal seleccionada.
4. **Desarrollo de Interfaz:** Construcción de una interfaz de usuario en Angular. Diseño de tres pestañas principales en la interfaz para diferentes funciones.
 - Primera-Bienvenida y Explicación: Presentación de información de bienvenida al paciente. Explicación del formato de las imágenes.
 - Segunda - Carga de Imágenes y Registro: Posibilidad de cargar imágenes. Registro de información relevante del paciente.
 - Tercera- Visualización de Datos de Pacientes: Mostrar los datos ingresados de los pacientes de manera organizada.

- 5. Formulario de Encuesta:** Incorporación de un formulario de encuesta al final del trabajo. Evaluación de la eficacia de la elección realizada. Aseguramiento de que la solución cumple con las expectativas del usuario.

Resultados

Desarrollo del Modelo:

La fase de diseño detallado se centra en definir con precisión la funcionalidad del proyecto, especialmente enfocándose en la implementación de redes neuronales mediante el uso del paquete neuralnet. Durante el proceso, al ingresar las imágenes, se realiza un procesamiento inicial de las mismas. Luego, se aplica un PCA, una herramienta que permite retener la información esencial de la imagen al reducir sus dimensiones, optimizando así el proceso de entrenamiento del modelo, esto se realizó a los 3 modelos entrenados.



```
{r}
carpeta0 <- "./Prueba/0"
X0 <- list.files(carpeta0, full.names = TRUE)
img0 <- lapply(X0, load.image)
img0 <- lapply(img0, grayscale)
img0 <- lapply(img0, as.data.frame)
csa1 <- lapply(img0, function(x)
  x$value)
csa2 <- as.data.frame(csa1)
csa2 <- rbind(csa2, rep("NO", ncol(csa2)))

carpeta1 <- "./Prueba/1"
X1 <- list.files(carpeta1, full.names = TRUE)
img1 <- lapply(X1, load.image)
img1 <- lapply(img1, grayscale)
img1 <- lapply(img1, as.data.frame)
csa3 <- lapply(img1, function(x)
  x$value)
csa4 <- as.data.frame(csa3)
csa4 <- rbind(csa4, rep("SI", ncol(csa4)))

datos <- rbind(t(csa2), t(csa4))
rownames(datos) <- paste0(rep(1:nrow(datos)), "_image")

datos <- as.data.frame(datos)
colnames(datos)[ncol(datos)] <- "Class"
datos$Class <- as.factor(datos$Class)

datos.num <- datos[, -ncol(datos)]
datos.num <- as.data.frame(lapply(datos.num, as.numeric))
pcx <- prcomp(datos.num)
```

Data	
csa1	Large list (295 elements, 118.4... Q
csa2	50177 obs. of 295 variables
csa3	Large list (237 elements, 95.1... Q
csa4	50177 obs. of 237 variables
data_model_1	List of 14
data_nrm	372 obs. of 535 variables
data_nrm.test	124 obs. of 535 variables
data_nrm.test	124 obs. of 533 variables
data_nrm.train	248 obs. of 535 variables
data_nrm.train	248 obs. of 533 variables
dataset	372 obs. of 533 variables
datos	532 obs. of 50177 variables
datos.num	532 obs. of 50176 variables
img0	Large list (295 elements, 237.1... Q
img1	Large list (237 elements, 190.5... Q
matrix	List of 6
model_result	num [1:124, 1:2] 0.9997 0.0117 0... Q
pcx	Large prcomp (5 elements, 222.7... Q
Values	
carpeta0	"./Prueba/0"
carpeta1	"./Prueba/1"
idx	Named int [1:124] 1 2 1 1 1 1 1 2 ...
index	int [1:372] 7 165 299 405 72 272 1... 372L
n	372L
p.train	0.6667
prediction	Factor w/ 2 levels "NO","SI": 1 2 ...
res	'table' int [1:2, 1:2] 62 14 10 38'

Figura 2. Carga de imágenes y procesamiento en R.

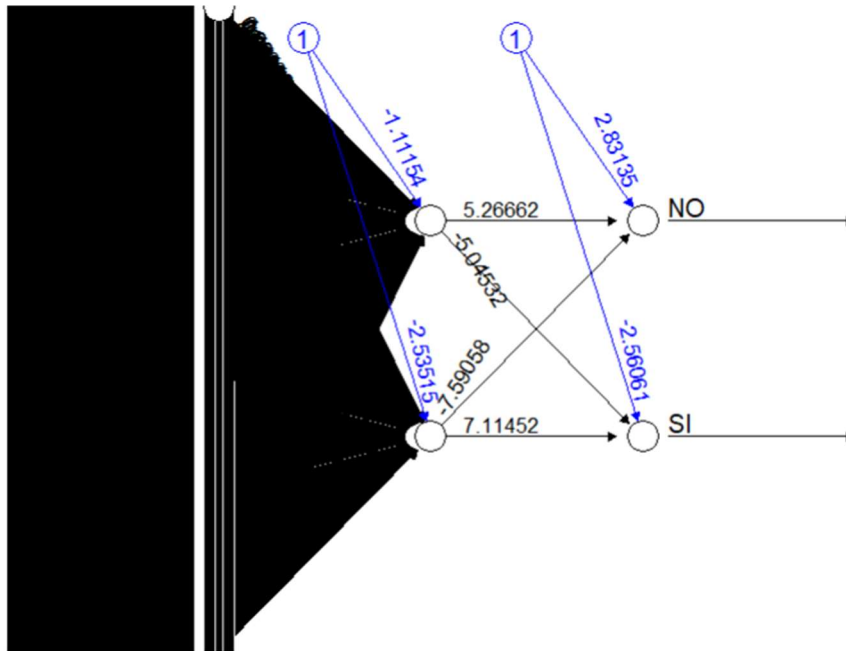


Figura 3. Diagrama de red neuronal artificial.

El entrenamiento del modelo se ejecuta en el entorno de programación R, dónde se busca conseguir más del 80% de precisión además de una buena sensibilidad y especificidad, además se probaron con otros modelos en los que se ha conseguido el 76%.

Se tienen las siguientes consideraciones con respecto a los modelos:

Modelo 1:

El modelo en la figura 4 es una **red neuronal artificial (RNA) perceptrón multicapa**. Este tipo de redes se caracteriza por tener una estructura organizada en capas, donde cada una de ellas está compuesta por un conjunto de neuronas artificiales.

En la imagen se observa una red neuronal con **una capa oculta de una neurona**. La de entrada tiene **n neuronas**, donde **n** es el número de variables predictoras en el conjunto de datos. La de salida tiene **dos neuronas**, una para cada clase a predecir (en este caso, "NO" y "SI").

El código muestra cómo se implementa la red neuronal utilizando la biblioteca neuralnet en el lenguaje R. Los pasos principales son:

Preparación de los datos: Se parten los datos en dos conjuntos: **entrenamiento (75%)** y **prueba (25%)**.

Definición del modelo: Se define la estructura de la red neuronal, incluyendo el número de neuronas por capa, el número de capas, y la función de activación.

```

set.seed(126734)
index <- sample(1:nrow(datos), nrow(datos) * 0.75)
train.d <- data.frame(pcx$x[index,], clase = datos[index, "clase"])
test.d <- data.frame(pcx$x[-index,], clase = datos[-index, "clase"])

#Modelo1
data_model_1 <-
  neuralnet(clase ~ .,
            data = train.d,
            hidden = 1,
            linear.output = FALSE)

model_results_1 <-
  compute(data_model_1, test.d[, -ncol(test.d)])$net.result

maxidx <- function(arr) {
  return(which(arr == max(arr)))
}

idx <- apply(model_results_1, 1, maxidx)
prediction <- as.factor(c("NO", "SI")[idx])
print(paste("Modelo 1"))

[1] "Modelo 1"

confusionMatrix(data = prediction, reference = test.d$clase)

```

Figura 4. Código de Modelo 1

Con estas consideraciones se tiene el siguiente resultado:

```

Confusion Matrix and Statistics

          Reference
Prediction NO SI
NO      64  20
SI      11  38

      Accuracy : 0.7669
      95% CI   : (0.6858, 0.8358)
No Information Rate : 0.5639
P-Value [Acc > NIR] : 8.553e-07

      Kappa   : 0.5176

McNemar's Test P-Value : 0.1508

      Sensitivity : 0.8533
      Specificity : 0.6552
      Pos Pred Value : 0.7619
      Neg Pred Value : 0.7755
      Prevalence   : 0.5639
      Detection Rate : 0.4812
      Detection Prevalence : 0.6316
      Balanced Accuracy : 0.7543

      'Positive' Class : NO

```

Figura 5. Matriz de confusión de modelo 1

El modelo analizado demuestra una precisión general del 76.69%, lo que indica que aproximadamente tres cuartas partes de las predicciones fueron correctas. Destaca su capacidad para identificar casos positivos con una sensibilidad del 85.33%, aunque su rendimiento en la identificación de casos negativos es menos destacado, con una especificidad del 65.52%. Con un valor predictivo positivo del 76.19%, el modelo acierta en alrededor del 76.19% de las veces que pronostica un resultado afirmativo. Asimismo, su valor predictivo negativo alcanza el 77.55%, lo que significa que cerca del 77.55% de las veces que pronostica un resultado negativo es acertado. La prevalencia de la clase "NO" en los datos es del 56.39%, lo que indica que más de la mitad de las instancias pertenecen a esta categoría. La tasa de detección, que indica qué proporción de los casos positivos es detectada por el modelo, se sitúa en el 48.12%. Por otro lado, la prevalencia de detección, que mide qué porcentaje de las predicciones positivas son realmente positivas, alcanza el 63.16%. Finalmente, la precisión equilibrada del modelo, que considera tanto la sensibilidad como la especificidad, es del 75.43%.

Modelo 2:

```
set.seed(1345276)
index <- sample(1:nrow(datos), nrow(datos) * 0.7)
train.d <- data.frame(pcx$x[index,], clase = datos[index, "clase"])
test.d <- data.frame(pcx$x[-index,],clase = datos[-index, "clase"])

#Modelo2
data_model_3 <-
  neuralnet(clase ~ .,
            data = train.d,
            hidden = 7,
            linear.output = FALSE,algorithm = "slr",act.fct = "logistic")

model_results_3 <-
  compute(data_model_3, test.d[, -ncol(test.d)])$net.result
# Put multiple binary output to categorical output
maxidx <- function(arr) {
  return(which(arr == max(arr)))
}
idx <- apply(model_results_3, 1, maxidx)
prediction <- as.factor(c("NO", "SI")[idx])

print(paste("Modelo 2"))

[1] "Modelo 2"

matrix2<-caret::confusionMatrix(data = prediction, reference = test.d$clase)
print(matrix2)
```

Confusion Matrix and Statistics

Figura 6. Código de Modelo 2

El modelo 2 es igualmente una Red neuronal artificial (RNA), con las siguientes características:

Algoritmo: Regresión lineal simple (SLR)

Función de activación: Logística

Capas ocultas: 7

Salida: Binaria (NO/SI)

Datos:

- train.d: Conjunto de entrenamiento se usa el 70% de los datos
- test.d: Conjunto de prueba se usa el 30% de los datos

Evaluación: Matriz de confusión

Librería: caret

En primer lugar, se emplea la función `data.frame` para organizar las variables de salida y entrada en un marco de datos. Luego, la función `neuralnet` es utilizada para entrenar la red neuronal con los datos proporcionados. Una vez entrenada, la función `compute` se emplea para calcular las salidas de la red neuronal para el conjunto de prueba. Posteriormente, la función `maxidx` es utilizada para determinar la clase predicha para cada instancia en el conjunto de prueba.

Finalmente, la función `confusionMatrix` del paquete `caret` es utilizada para generar y visualizar la matriz de confusión, lo que brinda una evaluación detallada del rendimiento del modelo, obteniendo así los siguientes resultados:

Confusion Matrix and Statistics

```
Reference
Prediction NO SI
NO 70 15
SI 23 52

Accuracy : 0.7625
95% CI : (0.6889, 0.8261)
No Information Rate : 0.5812
P-Value [Acc > NIR] : 1.21e-06

Kappa : 0.5201

McNemar's Test P-Value : 0.2561

Sensitivity : 0.7527
Specificity : 0.7761
Pos Pred Value : 0.8235
Neg Pred Value : 0.6933
Prevalence : 0.5813
Detection Rate : 0.4375
Detection Prevalence : 0.5312
Balanced Accuracy : 0.7644

'Positive' Class : NO
```

Figura 7. Matriz de confusión de modelo 2

La precisión del modelo es del 0.7625, lo que significa que el 76.25% de los puntos de datos se clasificaron correctamente. La precisión es del 0.8235, lo que significa que el 82.35% de los puntos de datos que se predijo que eran positivos realmente lo eran. La revocación es del 0.7527, lo que significa que el 75.27% de los puntos de datos positivos fueron identificados correctamente por el modelo.

Modelo 3

Finalmente se obtuvo un modelo con el 80% de precisión a continuación se detallará una breve explicación de las características de este modelo:

Tipo de modelo: Red neuronal artificial con dos capas ocultas.

Función de activación: Sigmoidea.

Algoritmo de entrenamiento: Descenso del gradiente.

Datos de entrenamiento: Se tiene un conjunto de datos de 532 variables numéricas y una variable objetivo binaria (enfermedad presente o ausente).

División del conjunto de datos: 66,67% para entrenamiento y 33,33% para probar.

Precisión del modelo: 80% (según la matriz de confusión)

Se ha utilizado la validación cruzada para ajustar los hiperparámetros del modelo. Y se ha utilizado la función plot() del paquete neuralnet para visualizar el modelo.

```

data_nrm.test2 <- data_nrm.test[,-c(534,535)]
model_results_1 <- compute(data_model_1,data_nrm.test2[,,-ncol(data_nrm.test2)])$net.result
maxidx <- function(arr) {
  return(which(arr == max(arr)))
}
idx <- apply(model_results_1, 1, maxidx)
prediction <- c("NO", "SI")[idx]
res <- table(prediction, dataset$Class[-train])
prediction <- as.factor(prediction)
matrix<-caret::confusionMatrix(prediction,dataset$Class[-train])
print(matrix)

```

Figura 8. Código de modelo 3

Se tienen así lo siguientes resultados:

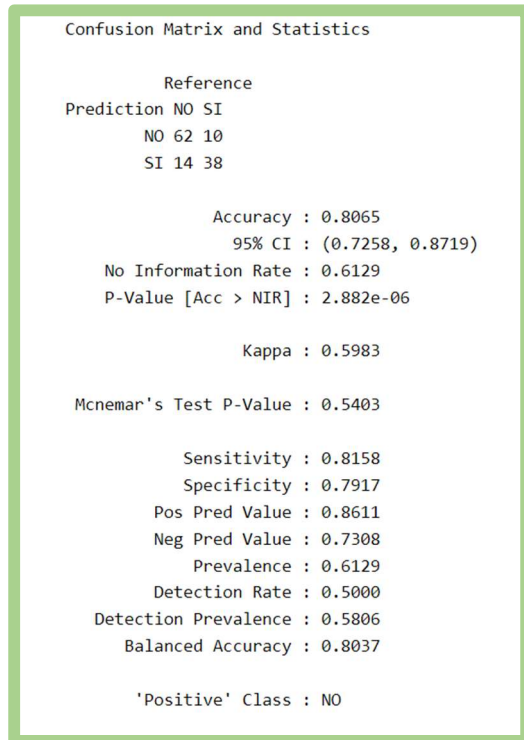


Figura 9. Matriz de confusión 3

La matriz de confusión muestra que hubo 62 verdaderos positivos (VP) y 53 verdaderos negativos (VN), lo que indica que el modelo fue efectivo para clasificar correctamente una buena cantidad de casos positivos y negativos. Sin embargo, también se observaron 14 falsos positivos (FP) y 11 falsos negativos (FN), lo que indica cierta propensión del modelo a errores en la clasificación.

Al examinar la tabla de estadísticas, se observa que la precisión del modelo fue del 80.65%, lo que significa que aproximadamente el 80.65% de todas las predicciones hechas por el modelo fueron

correctas. La sensibilidad fue del 81.58%, lo que indica la capacidad del modelo para identificar bien los casos positivos. La especificidad del modelo fue del 79.17%, lo que muestra su habilidad para identificar bien los casos negativos.

Además, el valor predictivo positivo (PPV) fue del 86.11%, lo que indica la probabilidad de que las instancias predichas como positivas si sean positivas, mientras que el valor predictivo negativo (NPV) fue del 82.86%, lo que sugiere la probabilidad de que las instancias predichas como negativas si sean negativas.

Desarrollo de la Interfaz:

Durante la fase inicial de desarrollo de esta interfaz, se creó un diseño preliminar utilizando el software libre Balsamiq, el cual ofreció una representación de la apariencia y funcionalidades esperadas de la aplicación final.



Figura 10. Pestaña “Detección” en propuesta de diseño de interfaz

Posterior a ello con un objetivo claro, la interfaz de usuario se desarrolló en Angular para que la interacción con el modelo sea más accesible para los usuarios. Se ha prestado especial atención para que el entorno sea amigable, dividiendo la interfaz en tres pestañas fundamentales y un inicio de sesión para llevar el registro del uso de la interfaz además de cuidar los datos de información médica ingresada.

A continuación, se puede observar un inicio de sesión cuando un usuario no se encuentra registrado tendrá la siguiente ventana para registrarse:

CREAR NUEVA CUENTA

Formulario de creación de nueva cuenta de usuario. Incluye tres campos de entrada: 'Email', 'Nombre y apellidos' y 'Password'. Debajo de los campos hay un checkbox 'Recordar mi usuario' y un enlace '¿Ya tienes cuenta? / Ingresar'. Al final hay un botón azul con el texto 'CREAR CUENTA'.

Figura 11. Creación de cuenta de usuario.

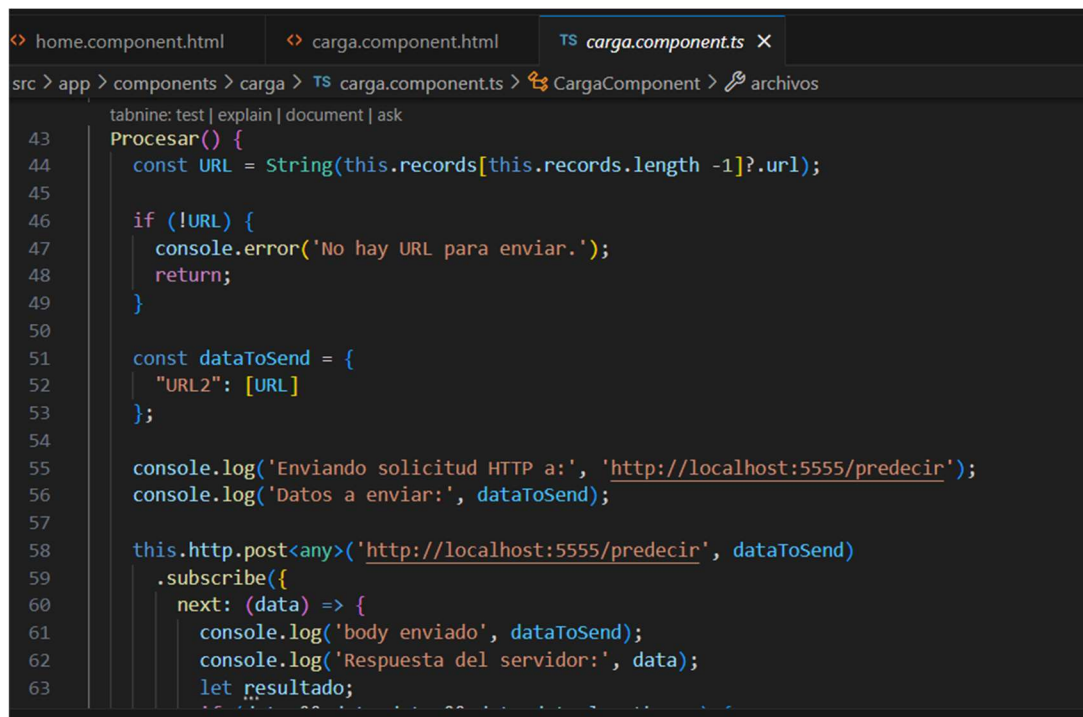
Una vez que el registro ha sido completado, el usuario puede iniciar sesión normalmente y seleccionar la opción para que su nombre de usuario sea recordado.

LOGIN

Formulario de inicio de sesión para usuarios registrados. Incluye dos campos de entrada: el primero contiene el correo electrónico 'tatianalola@hotmail.com' y el segundo muestra caracteres ocultos con puntos. Debajo de los campos hay un checkbox 'Recordar mi usuario' y un enlace '¿No tienes cuenta?'. Al final hay un botón azul con el texto 'INGRESAR'.

Figura 12. Inicio de sesión para los usuarios registrados

La primera pestaña da la bienvenida al paciente y proporciona información detallada sobre el formato necesario para las imágenes. En la segunda pestaña, se permite la carga de imágenes y el registro del paciente. La tercera pestaña presenta los datos ingresados de los pacientes, todo esto desarrollado en angular como se puede ver a continuación.



```
43 Procesar() {
44   const URL = String(this.records[this.records.length - 1]?.url);
45
46   if (!URL) {
47     console.error('No hay URL para enviar.');
```

```
48     return;
49   }
50
51   const dataToSend = {
52     "URL2": [URL]
53   };
54
55   console.log('Enviando solicitud HTTP a:', 'http://localhost:5555/predecir');
```

```
56   console.log('Datos a enviar:', dataToSend);
57
58   this.http.post<any>('http://localhost:5555/predecir', dataToSend)
59     .subscribe({
60       next: (data) => {
61         console.log('body enviado', dataToSend);
62         console.log('Respuesta del servidor:', data);
63         let resultado;
```

Figura 13. Programación en angular.

Pestaña “Home” informativa:

La documentación del proyecto constituye un elemento esencial para la comprensión y eficacia de la interfaz desarrollada. Este conjunto de recursos se convierte en un medio crucial para comunicar claramente la información sobre la interfaz, sus funciones y su uso, contribuyendo a la satisfacción del usuario, además en esta sección se informa al usuario el formato en el que se debe ingresar la imagen.

¡Bienvenido a la página web de Detección de Osteoartritis de Rodilla!



BIENVENIDO A JOINT CARE

La IA revoluciona la atención médica al identificar y clasificar la enfermedad en sus primeras etapas, ofreciendo beneficios significativos. Según la OMS, la artrosis afecta a 528 millones de personas, con un aumento del 113%. Nuestra herramienta no solo preserva la calidad de vida de los pacientes, sino que también reduce costos evitando tratamientos avanzados. Algoritmos avanzados permiten intervenciones oportunas al identificar cambios sutiles en radiografías.

INSTRUCCIONES DE USO

1. Verificar que la imagen sea formato .jpg
2. Verificar que el tamaño sea 224x224 px

Figura 14. Pestaña “Home” de la interfaz.

Pestaña de “Detección”:

En la pestaña de carga de imágenes, se lleva a cabo el proceso de predicción de manera simple y estructurada. En primer lugar, se registra la imagen, la cual se designará como la ficha médica del sistema. Luego, se puede seleccionar el archivo directamente o arrastrarlo a la ventana. Una vez que el archivo es cargado, se visualiza para su verificación. Posteriormente, se procede a completar los datos del paciente, incluyendo su nombre completo y edad. La ficha médica se completa

automáticamente al realizar este registro. Tras esto, se espera aproximadamente un minuto para que se realice la predicción. Una vez completada, se presiona el botón "Predecir y Guardar". Además, en caso de errores en la carga, existe la opción de limpiar el registro. Los datos también pueden ser editados libremente según sea necesario.

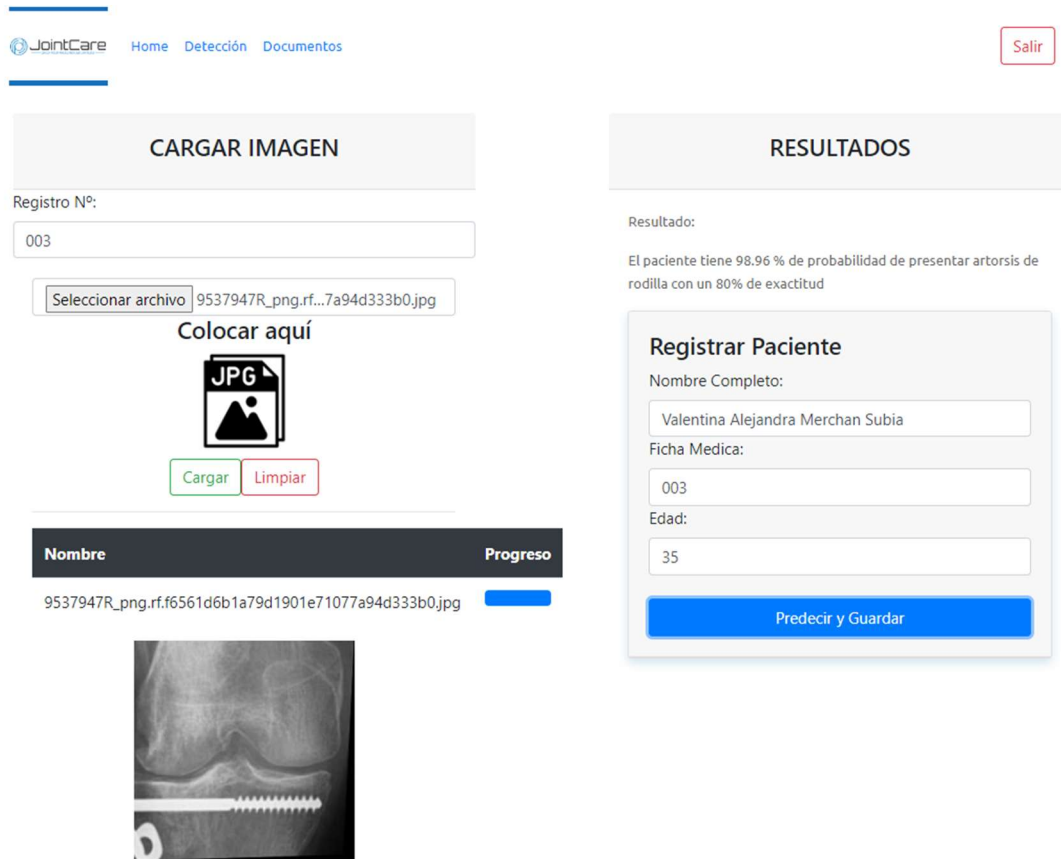


Figura 15. Pestaña de "Detección" en la interfaz.

Además, se requirió del uso de Firebase para almacenar toda esta información cargada:

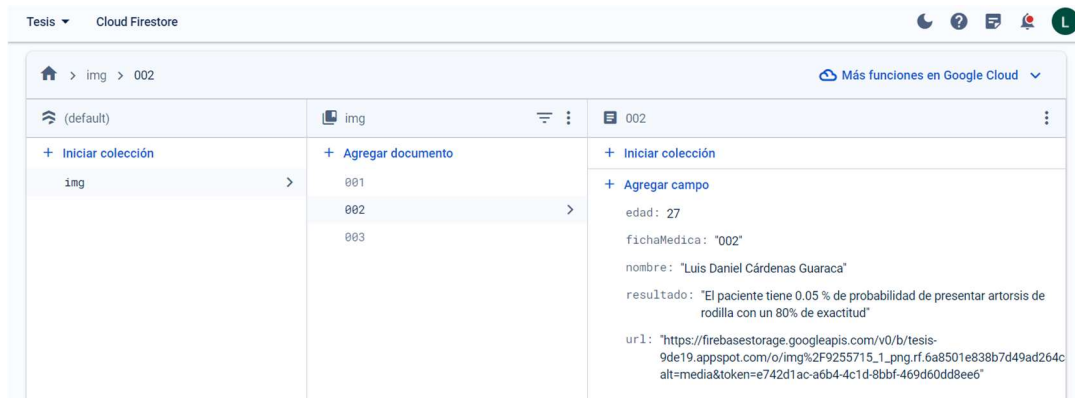


Figura 16. Uso de Firebase para almacenar datos.

Y así se almacena la información y el usuario puede tenerla a su disposición en “Documentos”.

Pestaña “Documentos”:

En esta pestaña, el personal de salud puede acceder a un registro completo de sus pacientes. Aquí se muestra la radiografía cargada, junto con el nombre del paciente, su ficha médica, su edad y la predicción realizada.

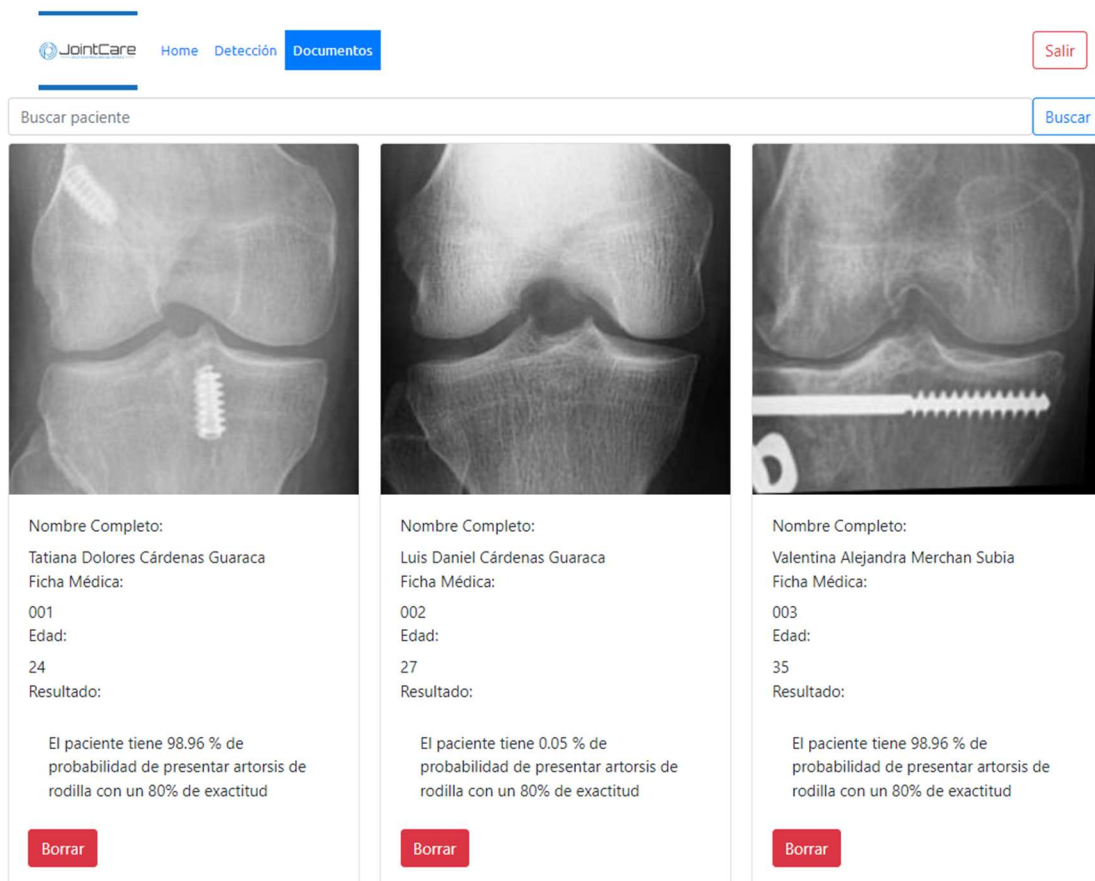


Figura 17. Pestaña de “Documentos” de la interfaz


En esta sección, se encuentran diversas funciones, incluyendo la búsqueda por nombre, ficha médica y edad pulsando “Buscar” la aplicación mostrara solo los datos de pacientes que coincidan con lo requerido. Además, se tiene la opción de eliminar todos los datos de un paciente pulsando “Borrar”.

JointCare Home Detección Documentos

Salir

Valentina

Buscar



Nombre Completo:
Valentina Alejandra Merchan Subia
Ficha Médica:
003
Edad:
35
Resultado:

El paciente tiene 98.96 % de probabilidad de presentar artrosis de rodilla con un 80% de exactitud

Borrar

Figura 18. Funciones de búsqueda

Interoperabilidad entre los sistemas:

Para garantizar una interoperabilidad adecuada entre R y Angular, se implementó R como servidor utilizando la biblioteca "plumber". Esta herramienta facilita la creación de un servidor web capaz de recibir solicitudes HTTP y llevar a cabo predicciones utilizando un modelo previamente entrenado, las pruebas se realizaron primeramente con Postman para garantizar la funcionalidad.

El funcionamiento del servidor implica recibir una solicitud POST en la ruta "/predecir", con una URL con la imagen radiológica en el cuerpo de la solicitud. Posteriormente, el servidor descarga la imagen radiológica, la convierte a escala de grises y la transforma en un vector de píxeles. Luego, carga un modelo de predicción y ejecuta una transformación de PCA sobre los datos de prueba. Finalmente, realiza la predicción de la clase de la imagen, en este caso, determinando si el paciente presenta o no osteoartritis de rodilla.

La respuesta del servidor incluye la predicción realizada y la probabilidad asociada a la presencia de osteoartritis de rodilla, con un nivel de precisión determinado.

```
# plumber.R
library(jsonlite)
library(plumber)
library(urltools)
library(dplyr)
library(imager)

options("plumber.port" = 5555)

# Filtro para habilitar CORS
#* @filter cors
cors <- function(res, req) {
  res$setHeader('Access-Control-Allow-Origin', '*') # Or whatever
  cat("Inicio de Servidor\n")

  if (req$REQUEST_METHOD == "OPTIONS") {
    res$setHeader("Access-Control-Allow-Methods", "*")
    res$setHeader("Access-Control-Allow-Headers", req$HTTP_ACCESS_CONTROL_REQUEST_HEADERS)
    res$status <- 200
    return(list())
  } else {
    plumber::forward()
  }
}

#* @preempt cors
#* @post /predecir
function(res, req) {
  # Obtener la URL del cuerpo de la solicitud
  url <- req$body$URL2
  header <- req$header

  # Imprimir la URL para propósitos de depuración
  cat("Cuerpo de la solicitud:", toJSON(req$body), "\n")
  cat("URL recibida:", url, "\n")
  cat("cabecera recibida:", header, "\n")

  url <-url
}
```

Figura 19. Código para interoperabilidad en los sistemas

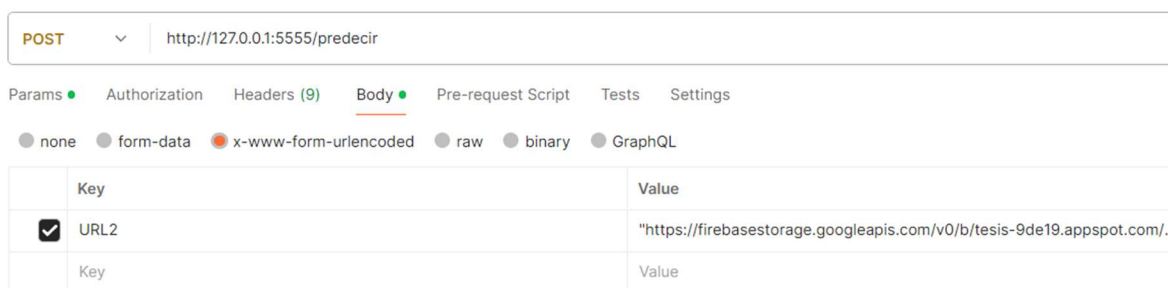


Figura 20. Pruebas realizadas con Postman para verificar funcionalidad.

Validación del proyecto

Al completar este proceso, se lleva a cabo una evaluación basada en la utilidad y satisfacción, utilizando un formulario de encuesta. Este enfoque asegura que la elección de la solución no solo cumpla con las expectativas del usuario, sino que también aborde

Preguntas Respuestas **3** Configuración

Sección 1 de 2

VALIDACIÓN DEL PROYECTO: PROCESAMIENTO DE IMÁGENES PARA DETECCIÓN DE OSTEOARTRITIS DE RODILLA UTILIZANDO IA

El formulario que se presenta a continuación desempeña un papel fundamental en la verificación de la pertinencia de una interfaz capaz de recibir imágenes radiológicas. A través de esta interfaz, se logra detectar con una precisión del 80% la presencia de artrosis de rodilla, al mismo tiempo que se almacena la información correspondiente al paciente.

Agradecemos sinceramente su tiempo y esfuerzo al participar en esta investigación, ya que su colaboración contribuirá de manera significativa a la validación de este estudio.

Nombre Completo *

Texto de respuesta corta

Figura 21. Formulario de validación del proyecto.

Este formulario aborda diversos aspectos sobre el uso de la inteligencia artificial (IA) en el diagnóstico y tratamiento de la osteoartritis de rodilla. Se busca entender opiniones de expertos sobre la utilidad de la IA para mejorar la precisión de los diagnósticos médicos, la fiabilidad de los resultados mediante métodos cuantitativos y analíticos, la personalización y seguimiento de pacientes en aplicaciones médicas, la fluidez de procesos en la interfaz de usuario, la visualización de datos para profesionales médicos, la capacidad de la IA para ofrecer resultados confiables en el análisis de imágenes médicas, la detección temprana de la osteoartritis de rodilla, la colaboración entre personal sanitario y sistemas de IA, la accesibilidad del diagnóstico presuntivo de la osteoartritis de rodilla, la adaptabilidad y mejora de la capacidad diagnóstica de la IA con más información clínica, la viabilidad económica de la implementación de IA en el diagnóstico, la percepción y confianza de los pacientes hacia la IA, si es correcto el tiempo de respuesta de la herramienta para procesar imágenes, la facilidad de uso de la herramienta y la claridad en la presentación y clasificación de resultados. Las respuestas se registran en una escala Likert que va desde el acuerdo hasta el desacuerdo.

La validación se llevó a cabo con médicos jóvenes, de género masculino, que ha interactuado con pacientes con osteoartritis de rodilla y que cuentan con años de experiencia. Se pueden destacar los siguientes resultados:

Satisfacción con la plataforma: La mayoría de los encuestados (66%) expresaron satisfacción general con la plataforma. Sin embargo, un segmento significativo (33%) manifestó insatisfacción, principalmente debido a problemas de usabilidad y funcionalidad.

Utilidad de la plataforma para la clasificación de lesiones: Dos tercios de los usuarios (66%) consideran útil la plataforma para clasificar el tipo de lesión detectada en las radiografías. Sin embargo, alrededor de un tercio no la encontró útil, resaltando la necesidad de mejorar la precisión y la facilidad de usabilidad.

Recomendación de la plataforma: Aproximadamente el 66% de los encuestados estarían dispuestos a recomendar la plataforma a otros profesionales de la salud, lo que indica un nivel considerable de confianza en su utilidad.

Contribución de métodos analíticos en investigación médica: Todos los encuestados (100%) reconocen la contribución positiva de los métodos cuantitativos y analíticos en la confiabilidad de los resultados de la constante investigación médica.

A continuación, se ilustra un ejemplo de como han sido respondidas estas preguntas:

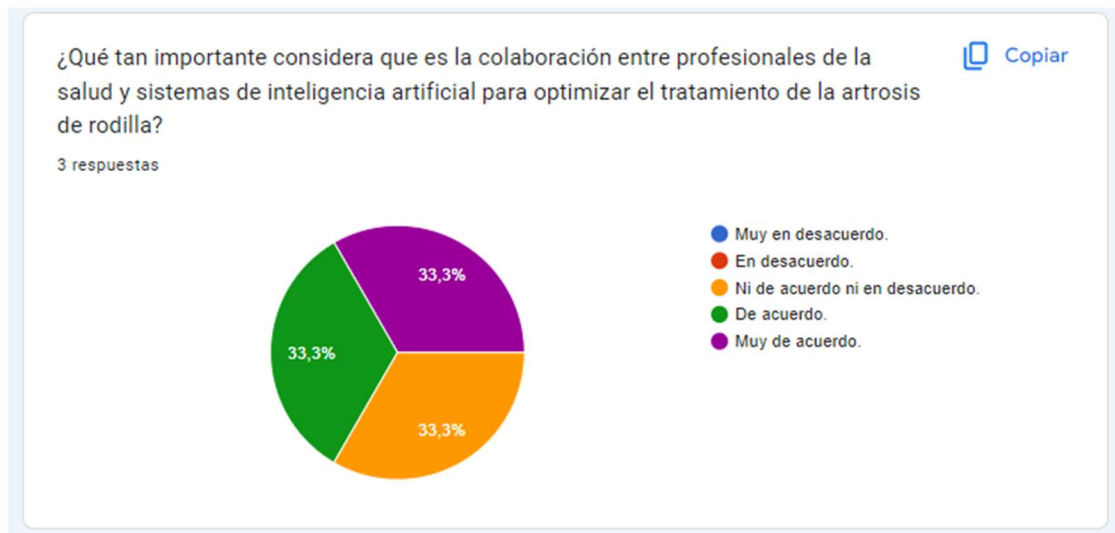


Figura 22. Ejemplo de resultado del formulario.

Cronograma

Tabla 1. Cronograma de actividades

CRONOGRAMA DE ACTIVIDADES														
OBJETIVOS ESPECIFICOS	ACTIVIDADES	Noviembre				Diciembre				Enero				Horas
		1	2	3	4	1	2	3	4	1	2	3	4	
Analizar y seleccionar una base de imágenes médicas que contenga datos relevantes para la identificación de la osteoartritis de rodilla.	Analizar características relevantes de las bases de imágenes médicas con fundamento teórico que respalde la identificación de la osteoartritis de rodilla.	X												20
	Seleccionar una base de imágenes médicas que cumpla los requerimientos para identificar osteoartritis de rodilla.		X											20
Seleccionar, ejecutar y evaluar un algoritmo de clasificación que permita determinar si un paciente tiene o no osteoartritis de rodilla basándose en las imágenes médicas.	Seleccionar un algoritmo de clasificación y uno de detección temprana para mejorar la identificación.			X	X	X								60
	Ejecutar y evaluar la identificación de osteoartritis de rodilla						X	X	X					50
Diseñar y validar una interfaz web que permita a los médicos interactuar con el sistema, proporcionando acceso a la información almacenada.	Diseñar una interfaz web que permita a los médicos interactuar con el sistema.								X	X	X			50
	Validar la interfaz web.											X	X	40
Horas totales													240	

Presupuesto

Tabla 2. Presupuesto del proyecto

Denominación	Cantidad	Costo Unitario	Costo total
Diseño del software	1	\$5x1h	200h=\$1000
Equipo de procesamiento de imagen	1	\$1000	\$1000
Pruebas y ajustes	1	\$5x1h	40h=200\$

Conclusiones

En conclusión, este estudio se basó en la implementación de redes neuronales utilizando el paquete *neuralnet* en R para la detección de osteoartritis a partir de imágenes médicas. Se diseñó un proceso que comprende la entrada de imágenes, seguido de un preprocesamiento mediante PCA para reducir la dimensionalidad y mejorar el entrenamiento del modelo.

El entrenamiento del modelo fue en R, mientras que la interfaz de usuario se desarrolló en Angular para ofrecer una experiencia amigable. Esta interfaz consta de un registro de usuario y tres pestañas principales: la primera para dar la bienvenida al paciente y explicar el formato de las imágenes, la segunda para cargar imágenes y registrar información del paciente, y la tercera para visualizar los datos ingresados de los pacientes.

Para asegurar la utilidad y satisfacción del usuario, se integró un formulario de encuesta al final del proceso. Este formulario evalúa la eficacia de la aplicación, garantizando que satisfaga las expectativas del usuario y aborde de manera eficiente la detección de osteoartritis.

El cumplimiento de los objetivos mencionados representa un avance significativo en el ámbito del diagnóstico de la osteoartritis. La combinación de una base de imágenes médicas relevantes, un algoritmo de clasificación preciso y una interfaz basada en técnicas de inteligencia artificial proporciona una herramienta integral y efectiva que facilita la detección temprana de esta condición. Esta solución mejora la precisión del diagnóstico y sirve como un valioso recurso de apoyo para los profesionales médicos, lo que puede llevar a una atención más temprana y eficaz para los pacientes afectados.

Recomendaciones

Para mejorar el rendimiento del modelo de detección de osteoartritis, se sugiere explorar otras opciones de modelos de redes neuronales disponibles en diferentes lenguajes de programación, como TensorFlow o PyTorch en Python, además de utilizar el paquete neuralnet en R. La evaluación comparativa de estos modelos puede ayudar a determinar cuál es el más adecuado para este problema y mejorar la precisión de las predicciones. Asimismo, se debe prestar especial atención a la interoperabilidad entre los diferentes lenguajes de programación utilizados, dedicando tiempo adicional para garantizar una integración fluida y una comunicación eficiente entre los componentes del sistema, especialmente en lo que respecta a la compatibilidad de datos y la transferencia de información.

Es crucial realizar pruebas exhaustivas en cada etapa del desarrollo del trabajo, con ajustes iterativos basados en los comentarios que brinden los usuarios y los resultados de las pruebas para mejorar continuamente la funcionalidad y la usabilidad de la aplicación.

Finalmente, se recomienda realizar una recolección y análisis periódico de datos de encuestas para evaluar la eficacia y la satisfacción del usuario, identificando áreas de mejora y realizando ajustes en consecuencia. Esto asegurará que la aplicación satisfaga de manera efectiva las expectativas del usuario y cumpla con sus necesidades de forma constante en el tiempo.

Referencias bibliográficas

Ali, M. (2023, 9 noviembre). Introduction to Activation Functions in Neural Networks. <https://www.datacamp.com/tutorial/introduction-to-activation-functions-in-neural-networks>

Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A. Q., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M. A., Al-Amidie, M., & Farhan, L. (2021). Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. *Journal Of Big Data*, 8(1). <https://doi.org/10.1186/s40537-021-00444-8>

Ardila Muñoz, J.Y., & Molina Sosa, D.A. (2023). Una Aproximación Investigativa Para El Uso De La Gamificación En El Aprendizaje De Lenguajes De Programación. *Latinoamericana de Estudios Educativos*.

Atefi, Y., Ahearne, M., Hohenberg, S., Hall, Z.R., & Zettelmeyer, F. (2020). Open Negotiation: The Back-End Benefits of Salespeople's Transparency in the Front End. *Journal of Marketing Research*, 57, 1076 - 1094.

García-Quiroga, M., Roig, D., Mok, C., & Cazorla, K. (2022). ¿Quién puede participar? Un análisis documental acerca de la participación de la infancia en cuidados alternativos. *REVISTA CUHSO*.

Goh, H., Ho, C., & Abas, F. S. (2022). Front-end deep learning web apps development and deployment: a review. *Applied Intelligence*, 53(12), 15923-15945. <https://doi.org/10.1007/s10489-022-04278-6>

Gore, J. C. (2020). Artificial intelligence in medical imaging. *Magnetic Resonance Imaging*, 68, A1-A4.

Google Health. (2023). *Uso de la IA en imágenes y diagnósticos* - Google Health.

Hili, N., & Oliveira, R.A. (2022). A light-weight low-code platform for back-end automation. *Proceedings of the 25th International Conference on Model Driven Engineering Languages and Systems: Companion Proceedings*.

Huang, C., & Kuo, P. (2019). Multiple-Input Deep Convolutional Neural Network Model for Short-Term Photovoltaic Power Forecasting. *IEEE Access*, 7, 74822-74834.

Javid, A. M. (2021). *Neural Network Architecture Design: Towards Low-complexity and Scalable Solutions* (Doctoral Thesis in Electrical Engineering). KTH Royal Institute of Technology, Stockholm, Sweden.

Logroño-Naranjo, S.I., López-Paredes, C.R., Estrada-Brito, N.A., & Vásquez-Núñez, V.A. (2021). Utilización de lenguajes de programación para generar plataformas informáticas en pro del desarrollo del sector turístico que ayude al Crecimiento Económico. *Dominio de las Ciencias*, 7, 1627-1636.

Madaleno, F. O., Santos, B. A., Araujo, V. L., Oliveira, V. C., & Resende, R. A. (2018). Prevalence of knee osteoarthritis in former athletes: a systematic review with meta-analysis. *Brazilian journal of physical therapy*, 22(6), 437-451.

Martínez, R., & Cecilia, C.A. (2020). *Contribuciones a los métodos STASIS basados en técnicas de aprendizaje no supervisado*.

Mazaeda, R., de la Fuente López, E., González, J., & Moya de la Torre, E.J. (2022). *Docencia en informática industrial: lenguajes de programación*. *Actas de las XXXVII Jornadas de Automática* 7, 8 y 9 de septiembre de 2016, Madrid.

Medimaging.es. (2023). *IA detecta osteoartritis de rodilla temprana a partir de imágenes de rayos X*.

Medscape. (2022, marzo 2). No se conoce el impacto epidemiológico ni la carga económica de la osteoartritis en Latinoamérica.

Organización Mundial de la Salud. (2021). La OMS publica el primer informe mundial sobre inteligencia artificial (IA) aplicada a la salud y seis principios rectores relativos a su concepción y utilización. Recuperado de <https://www.who.int/es/news/item/28-06-2021-who-issues-first-global-report-on-ai-in-health-and-six-guiding-principles-for-its-design-and-use>

Physiotutors. (2021). Osteoartritis de rodilla | Diagnóstico y tratamiento. Recuperado de <https://www.physiotutors.com/es/conditions/knee-osteoarthritis/>

Palafox-Palafox, D., Medina-Marín, J., Seck-Tuoh-Mora, J.C., Serna-Díaz, M.G., & Hernández-Romero, N. (2023). Modelo de pronóstico de cadena de suministro mediante redes neuronales. Pádi Boletín Científico de Ciencias Básicas e Ingenierías del ICBI.

Saioc, & Vlad, G. (2022). A Tool for Describing and Checking Natural Semantics Definitions of Programming Languages.

The Black Box Lab. (2022). Machine Learning: Algoritmos de clasificación y regresión. Recuperado de <https://theblackboxlab.com/2022/05/06/machine-learning-diferencias-entre-algoritmos-clasificacion-regresion/>

UNIR México. (2021). La inteligencia artificial en la medicina: usos y futuro. Recuperado de <https://mexico.unir.net/salud/noticias/inteligencia-artificial-medicina/>

Vargas, Y., & Cecilia, D.G. (2019). Arquitectura de Interoperabilidad de dispositivos físicos para el Internet de las Cosas (IoT).

Zapeta Hernández, A., Galindo Rosales, G.A., Juan Santiago, H.J., & Martínez Lee, M. (2022). Métricas de rendimiento para evaluar el aprendizaje automático en la clasificación de imágenes petroleras utilizando redes neuronales convolucionales. Ciencia Latina Revista Científica Multidisciplinar.

Anexos

- A. Códigos de modelos de predicción
- B. Propuesta de diseño de interfaz
- C. Códigos principales de interfaz web
- D. Código de interoperabilidad
- E. Formulario para validación

TESIS

```
library(dplyr)
```

Warning: package 'dplyr' was built under R version 4.3.2

Attaching package: 'dplyr'

The following objects are masked from 'package:stats':

filter, lag

The following objects are masked from 'package:base':

intersect, setdiff, setequal, union

```
library(imager)
```

Warning: package 'imager' was built under R version 4.3.2

Loading required package: magrittr

Attaching package: 'imager'

The following object is masked from 'package:magrittr':

add

The following object is masked from 'package:dplyr':

where

The following objects are masked from 'package:stats':

convolve, spectrum

The following object is masked from 'package:graphics':

frame

The following object is masked from 'package:base':

save.image

```
carpeta0 <- "./Prueba/0"  
X0 <- list.files(carpeta0, full.names = TRUE)
```

```
img0 <- lapply(X0, load.image)
img0 <- lapply(img0, grayscale)
img0 <- lapply(img0, as.data.frame)
csa1 <- lapply(img0, function(x)
  x$value)
csa2 <- as.data.frame(csa1)
csa2 <- rbind(csa2, rep("NO", ncol(csa2)))

carpeta1 <- "./Prueba/1"
X1 <- list.files(carpeta1, full.names = TRUE)
img1 <- lapply(X1, load.image)
img1 <- lapply(img1, grayscale)
img1 <- lapply(img1, as.data.frame)
csa3 <- lapply(img1, function(x)
  x$value)
csa4 <- as.data.frame(csa3)
csa4 <- rbind(csa4, rep("SI", ncol(csa4)))

datos <- rbind(t(csa2), t(csa4))

rownames(datos) <- paste0(rep(1:nrow(datos)), "_image")

datos <- as.data.frame(datos)

colnames(datos)[ncol(datos)] <- "clase"

datos$clase <- as.factor(datos$clase)

datos.num <- datos[,-ncol(datos)]
datos.num <- as.data.frame(lapply(datos.num, as.numeric))

pcx <- prcomp(datos.num)

#library(ggplot2)

#data.df <- data.frame(pcx$x, color = datos$clase)

#ggplot(data.df, aes(PC1, PC2, color = color)) + geom_point()
```

```
library(caret)
```

Warning: package 'caret' was built under R version 4.3.2

Loading required package: ggplot2

Warning: package 'ggplot2' was built under R version 4.3.2

Loading required package: lattice

```
library(neuralnet)
```

Warning: package 'neuralnet' was built under R version 4.3.2

Attaching package: 'neuralnet'

The following object is masked from 'package:dplyr':

compute

```
set.seed(126734)
index <- sample(1:nrow(datos), nrow(datos) * 0.75)
train.d <- data.frame(pcx$x[index,], clase = datos[index, "clase"])
test.d <- data.frame(pcx$x[-index,], clase = datos[-index, "clase"])

#Modelo1
data_model_1 <-
  neuralnet(clase ~ .,
            data = train.d,
            hidden = 1,
            linear.output = FALSE)

model_results_1 <-
  compute(data_model_1, test.d[, -ncol(test.d)])$net.result

maxidx <- function(arr) {
  return(which(arr == max(arr)))
}

idx <- apply(model_results_1, 1, maxidx)
prediction <- as.factor(c("NO", "SI")[idx])
print(paste("Modelo 1"))
```

[1] "Modelo 1"

```
confusionMatrix(data = prediction, reference = test.d$clase)
```

Confusion Matrix and Statistics

	Reference	
Prediction	NO	SI
NO	64	20
SI	11	38

Accuracy : 0.7669
 95% CI : (0.6858, 0.8358)
 No Information Rate : 0.5639

P-Value [Acc > NIR] : 8.553e-07

Kappa : 0.5176

Mcnemar's Test P-Value : 0.1508

Sensitivity : 0.8533

Specificity : 0.6552

Pos Pred Value : 0.7619

Neg Pred Value : 0.7755

Prevalence : 0.5639

Detection Rate : 0.4812

Detection Prevalence : 0.6316

Balanced Accuracy : 0.7543

'Positive' Class : NO



TESIS

```
library(dplyr)
```

Warning: package 'dplyr' was built under R version 4.3.2

Attaching package: 'dplyr'

The following objects are masked from 'package:stats':

filter, lag

The following objects are masked from 'package:base':

intersect, setdiff, setequal, union

```
library(imager)
```

Warning: package 'imager' was built under R version 4.3.2

Loading required package: magrittr

Attaching package: 'imager'

The following object is masked from 'package:magrittr':

add

The following object is masked from 'package:dplyr':

where

The following objects are masked from 'package:stats':

convolve, spectrum

The following object is masked from 'package:graphics':

frame

The following object is masked from 'package:base':

save.image

```
carpeta0 <- "./Prueba/0"  
X0 <- list.files(carpeta0, full.names = TRUE)
```



```
img0 <- lapply(X0, load.image)
img0 <- lapply(img0, grayscale)
img0 <- lapply(img0, as.data.frame)
csa1 <- lapply(img0, function(x)
  x$value)
csa2 <- as.data.frame(csa1)
csa2 <- rbind(csa2, rep("NO", ncol(csa2)))

carpeta1 <- "./Prueba/1"
X1 <- list.files(carpetas1, full.names = TRUE)
img1 <- lapply(X1, load.image)
img1 <- lapply(img1, grayscale)
img1 <- lapply(img1, as.data.frame)
csa3 <- lapply(img1, function(x)
  x$value)
csa4 <- as.data.frame(csa3)
csa4 <- rbind(csa4, rep("SI", ncol(csa4)))

datos <- rbind(t(csa2), t(csa4))

rownames(datos) <- paste0(rep(1:nrow(datos)), "_image")

datos <- as.data.frame(datos)

colnames(datos)[ncol(datos)] <- "clase"

datos$clase <- as.factor(datos$clase)

datos.num <- datos[,-ncol(datos)]
datos.num <- as.data.frame(lapply(datos.num, as.numeric))

pcx <- prcomp(datos.num)

#library(ggplot2)

#data.df <- data.frame(pcx$x, color = datos$clase)

#ggplot(data.df, aes(PC1, PC2, color = color)) + geom_point()
```

```
library(caret)
```

Warning: package 'caret' was built under R version 4.3.2

Loading required package: ggplot2

Warning: package 'ggplot2' was built under R version 4.3.2

Loading required package: lattice

```
library(neuralnet)
```

Warning: package 'neuralnet' was built under R version 4.3.2

Attaching package: 'neuralnet'

The following object is masked from 'package:dplyr':

compute

```
set.seed(1345276)
index <- sample(1:nrow(datos), nrow(datos) * 0.7)
train.d <- data.frame(pcx$x[index,], clase = datos[index, "clase"])
test.d <- data.frame(pcx$x[-index,], clase = datos[-index, "clase"])

#Modelo2
data_model_3 <-
  neuralnet(clase ~ .,
            data = train.d,
            hidden = 7,
            linear.output = FALSE, algorithm = "slr", act.fct = "logistic")

model_results_3 <-
  compute(data_model_3, test.d[, -ncol(test.d)])$net.result
# Put multiple binary output to categorical output
maxidx <- function(arr) {
  return(which(arr == max(arr)))
}
idx <- apply(model_results_3, 1, maxidx)
prediction <- as.factor(c("NO", "SI")[idx])

print(paste("Modelo 2"))
```

[1] "Modelo 2"

```
matrix2 <- caret::confusionMatrix(data = prediction, reference = test.d$clase)
print(matrix2)
```

Confusion Matrix and Statistics

	Reference	
Prediction	NO	SI
NO	70	15
SI	23	52

Accuracy : 0.7625
95% CI : (0.6889, 0.8261)

No Information Rate : 0.5812
P-Value [Acc > NIR] : 1.21e-06

Kappa : 0.5201

Mcnemar's Test P-Value : 0.2561

Sensitivity : 0.7527
Specificity : 0.7761
Pos Pred Value : 0.8235
Neg Pred Value : 0.6933
Prevalence : 0.5813
Detection Rate : 0.4375
Detection Prevalence : 0.5312
Balanced Accuracy : 0.7644

'Positive' Class : NO



MODELO ANN ACCEPTABLE TESIS

```
# Cargar bibliotecas  
library(dplyr)
```

Warning: package 'dplyr' was built under R version 4.3.2

Attaching package: 'dplyr'

The following objects are masked from 'package:stats':

filter, lag

The following objects are masked from 'package:base':

intersect, setdiff, setequal, union

```
library(imager)
```

Warning: package 'imager' was built under R version 4.3.2

Loading required package: magrittr

Attaching package: 'imager'

The following object is masked from 'package:magrittr':

add

The following object is masked from 'package:dplyr':

where

The following objects are masked from 'package:stats':

convolve, spectrum

The following object is masked from 'package:graphics':

frame

The following object is masked from 'package:base':

save.image

```
library(ggplot2)
```

Warning: package 'ggplot2' was built under R version 4.3.2

```
library(caret)
```

Warning: package 'caret' was built under R version 4.3.2

Loading required package: lattice

```
library(reshape2)
```

Warning: package 'reshape2' was built under R version 4.3.2

```
library(ggcorrplot)
```

Warning: package 'ggcorrplot' was built under R version 4.3.2

```
library(neuralnet)
```

Warning: package 'neuralnet' was built under R version 4.3.2

Attaching package: 'neuralnet'

The following object is masked from 'package:dplyr':

compute

```
library(NeuralNetTools)
```

Warning: package 'NeuralNetTools' was built under R version 4.3.2

```
carpeta0 <- "./Prueba/0"  
X0 <- list.files(carpeta0, full.names = TRUE)  
img0 <- lapply(X0, load.image)  
img0 <- lapply(img0, grayscale)  
img0 <- lapply(img0, as.data.frame)  
csa1 <- lapply(img0, function(x)  
  x$value)  
csa2 <- as.data.frame(csa1)  
csa2 <- rbind(csa2, rep("NO", ncol(csa2)))  
  
carpeta1 <- "./Prueba/1"  
X1 <- list.files(carpeta1, full.names = TRUE)  
img1 <- lapply(X1, load.image)  
img1 <- lapply(img1, grayscale)  
img1 <- lapply(img1, as.data.frame)  
csa3 <- lapply(img1, function(x)  
  x$value)
```

```
csa4 <- as.data.frame(csa3)
csa4 <- rbind(csa4, rep("SI", ncol(csa4)))

datos <- rbind(t(csa2), t(csa4))

rownames(datos) <- paste0(rep(1:nrow(datos)), "_image")

datos <- as.data.frame(datos)

colnames(datos)[ncol(datos)] <- "Class"

datos$Class <- as.factor(datos$Class)

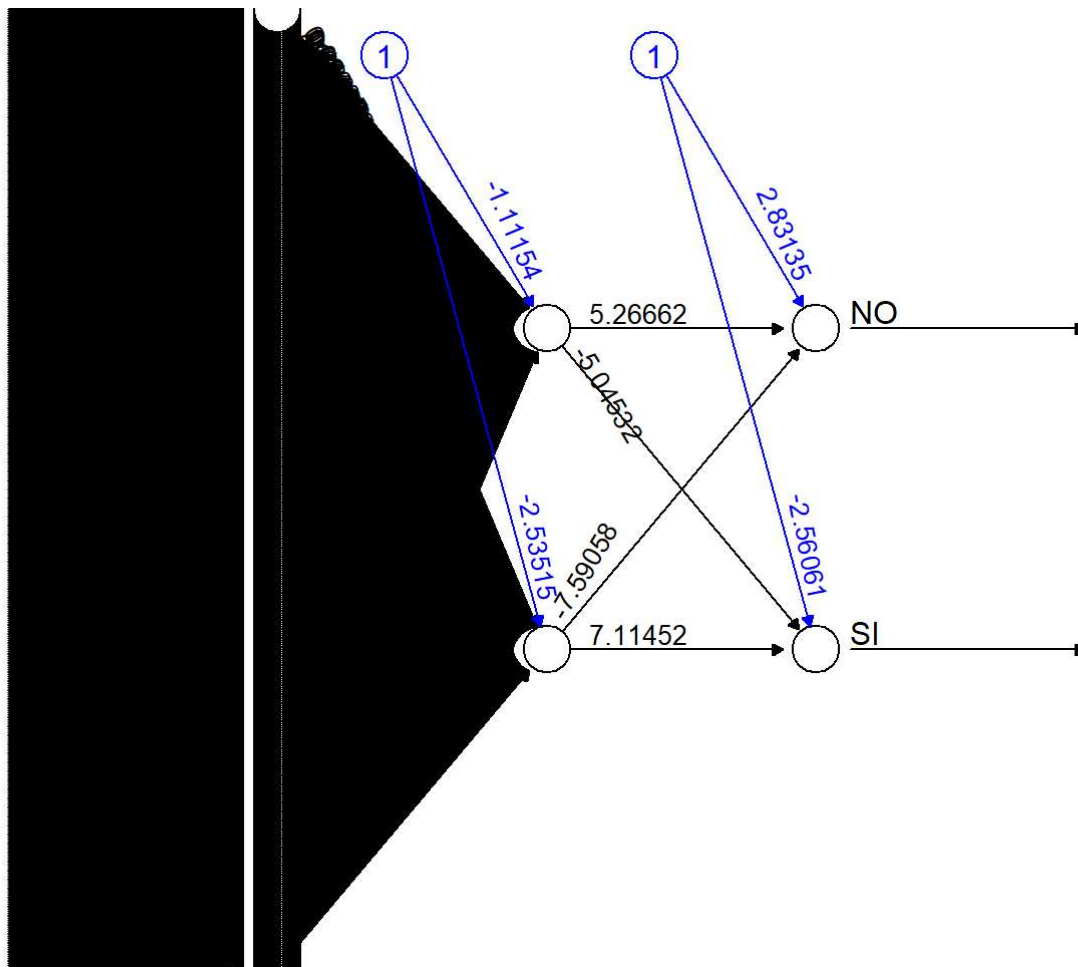
datos.num <- datos[,-ncol(datos)]
datos.num <- as.data.frame(lapply(datos.num, as.numeric))

pcx <- prcomp(datos.num)
```

```
p.train<-0.6667
set.seed(1745362)
index <- sample(1:nrow(datos), nrow(datos) * 0.7)
dataset <- data.frame(pcx$x[index,], Class = datos[index, "Class"])
length(complete.cases(dataset))
```

[1] 372

```
data_nrm <- dataset
data_nrm$M <- dataset$Class=="SI"
data_nrm$B <- dataset$Class=="NO"
n <- nrow(data_nrm)
train <- sample(n,floor(n*p.train))
data_nrm.train <- data_nrm[train,]
data_nrm.test <- data_nrm[-train,]
data_nrm.train2 <- data_nrm.train[,-c(534,535)]
data_model_1 <- neuralnet(Class ~., data = data_nrm.train2, hidden = 2, linear.output = FALSE)
plot(data_model_1, rep = "best")
```



```

data_nrm.test2 <- data_nrm.test[,-c(534,535)]
model_results_1 <- compute(data_model_1,data_nrm.test2[,-ncol(data_nrm.test2)])$net.result
maxidx <- function(arr) {
  return(which(arr == max(arr)))
}
idx <- apply(model_results_1, 1, maxidx)
prediction <- c("NO", "SI")[idx]
res <- table(prediction, dataset$Class[-train])
prediction <- as.factor(prediction)
matrix<-caret::confusionMatrix(prediction,dataset$Class[-train])
print(matrix)

```

Confusion Matrix and Statistics

Reference
Prediction NO SI
NO 62 10
SI 14 38

Accuracy : 0.8065
95% CI : (0.7258, 0.8719)
No Information Rate : 0.6129
P-Value [Acc > NIR] : 2.882e-06

Kappa : 0.5983

Mcnemar's Test P-Value : 0.5403

Sensitivity : 0.8158

Specificity : 0.7917

Pos Pred Value : 0.8611

Neg Pred Value : 0.7308

Prevalence : 0.6129

Detection Rate : 0.5000

Detection Prevalence : 0.5806

Balanced Accuracy : 0.8037

'Positive' Class : NO

Course Title Here

<http://coursehomepage.com/>



<input type="text" value="Email"/>	<input type="text" value="Password"/>	<input type="checkbox"/> Remember me	<input type="button" value="Sign in"/>
			Create an account



BIENVENIDO A JOINT CARE

Bienvenidos a Joint Care para la detección temprana de la osteoartritis de rodilla. La IA revoluciona la atención médica al identificar y clasificar la enfermedad en sus primeras etapas, ofreciendo beneficios significativos. Según la OMS, la artrosis afecta a 528 millones de personas, con un aumento del 113%. Nuestra herramienta no solo preserva la calidad de vida de los pacientes, sino que también reduce costos evitando tratamientos avanzados. Algoritmos avanzados permiten intervenciones oportunas al identificar cambios sutiles en radiografías.

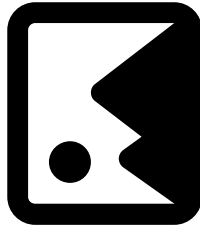
INSTRUCCIONES DE USO

1. Verificar que la imagen sea formato .jpg

The image will be imported into this project.

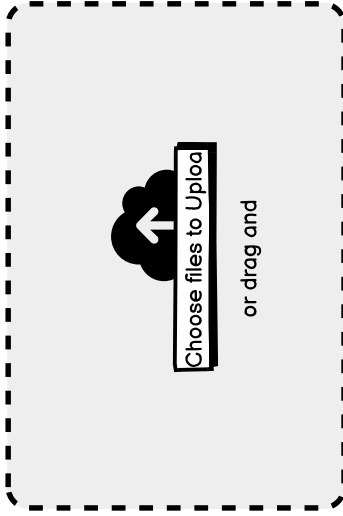
2. Verificar que el tamaño sea 224x224

Hold down the SHIFT key to maintain its aspect ratio.



RESULTADO DE LA DETECCIÓN

En la imagen proporcionada, se observa una evidencia de osteoartritis de rodilla en su cuarto grado en el paciente



Registrar pac



Comentarios sobre el caso

Nombre Completo

Edad

Ficha Médica

Guardar

Course Title Here



<http://coursehomepage.com/>



 **JointCare**

[Home](#)

[Detección](#)

[Documentos](#)

[username@gmail.co](#) | [Logout](#)



```

//auth.service.ts (Autenticación)
import { HttpClient } from '@angular/common/http';
import { Injectable } from '@angular/core';
import { UsuarioModel } from '../Models/usuario.model';
import { map } from 'rxjs/operators';

@Injectable({
  providedIn: 'root'
})
export class AuthService {
  //Para registrarse-crear cuenta de usuario
  //https://identitytoolkit.googleapis.com/v1/accounts:signUp?key=[API_KEY]

  //Para ingresar

  //https://identitytoolkit.googleapis.com/v1/accounts:signInWithPassword?key=[API_KEY]
  ]
  private url='https://identitytoolkit.googleapis.com/v1/accounts: ';
  private apikey='AIzaSyDFqJtfA9T4j4e09bSR8pG664oB41P9kYM';
  tokenDeUsuario:string;
  private guardarToken(idToken:string){
    let fe_tope=new Date();
    fe_tope.setTime(fe_tope.getTime()+3600*1000); //definimos fecha una hora despues
    localStorage.setItem('topeP63', fe_tope.getTime().toString());

    this.tokenDeUsuario=idToken;
    localStorage.setItem('token',idToken);
  }
  private readToken(){
    if(localStorage.getItem('token')){
      this.tokenDeUsuario=localStorage.getItem('token');
    }else{
      this.tokenDeUsuario='';
    }
    return this.tokenDeUsuario;
  }

  constructor(private http:HttpClient) {
    this.readToken();
  }
  login(usuario:UsuarioModel){
    const authData={
      email:usuario.email,
      password:usuario.password,
      returnSecureToken:true
    }
    return
    this.http.post(`${this.url}signInWithPassword?key=${this.apikey}`, authData)
      .pipe(map(resp=>{
        console.log('Login con password');

```

```

        this.guardarToken(resp['idToken']);
        return resp;
    }));
}
nuevoUsuario(usuario:UsuarioModel){
    const authData={
        email: usuario.email,
        password: usuario.password,
        returnSecureToken: true
    }
    return this.http.post(`${this.url}signUp?key=${this.apikey}`,authData)
        .pipe(map(resp=>{
            console.log('Creación de Nuevo Usuario con password');
            this.guardarToken(resp['idToken']);
            return resp;
        })));
}
logOut(){
    localStorage.removeItem('token');
    localStorage.removeItem('topeP63');
}
estaAutenticado(){
    if(this.tokenDeUsuario.length<2 || !localStorage.getItem('topeP63')){
        return false;
    }
    const fe_tope=new Date();
    const tope=Number(localStorage.getItem('topeP63'));
    fe_tope.setTime(tope);
    if(new Date()< fe_tope){
        return true;
    }else{
        return false;
    }
}
}
}

```

```

//carga.component.ts (Código de ventana "Detección")
import { Component, OnInit, EventEmitter, Output } from '@angular/core';
import { CargaImagenesService } from 'src/app/services/carga-imagenes.service';
import { Paciente, item } from '../fotos/fotos.component';
import { FileItem } from 'src/app/Models/file-item';
import { AngularFireStore } from '@angular/fire/compat/firestore';
import { HttpClient } from '@angular/common/http';

@Component({
  selector: 'app-carga',
  templateUrl: './carga.component.html',
  styleUrls: ['./carga.component.css']
})
export class CargaComponent implements OnInit {
  public item: Paciente = item;
  archivos: FileItem[] = [];
  estaSobreElemento = false;
  file: any;
  imageUrl: string | ArrayBuffer | null = null;
  public resultado: any;
  records: any[] = [];
  texto: string;
  @Output() resultadoEvent = new EventEmitter<any>();

  showResultado() {
    this.resultadoEvent.emit(this.resultado);
  }

  constructor(
    public _cargaImagenes: CargaImagenesService,
    private firestore: AngularFireStore,
    private http: HttpClient,
  ) {}

  ngOnInit() {
    this.firestore.collection('img').valueChanges().subscribe(records =>
this.records = records);
  }

  cargarImagenes() {
    this._cargaImagenes.cargarImagenesFirebase(this.archivos, this.item);
  }

  Procesar() {
    const URL = String(this.records[this.records.length -1]?.url);

    if (!URL) {
      console.error('No hay URL para enviar.');
```

```

const dataToSend = {
  "URL2": [URL]
};

console.log('Enviando solicitud HTTP a:', 'http://localhost:5555/predecir');
console.log('Datos a enviar:', dataToSend);

this.http.post<any>('http://localhost:5555/predecir', dataToSend)
  .subscribe({
    next: (data) => {
      console.log('body enviado', dataToSend);
      console.log('Respuesta del servidor:', data);
      let resultado;
      if (data && data.data && data.data.length > 0) {
        resultado = data.data[0]; // Tomamos el primer elemento del arreglo como
el mensaje
      } else {
        resultado = 'No se encontró un mensaje en la respuesta del servidor.';
      }
      this.resultado = resultado; // Asignamos el resultado
      this.resultadoEvent.emit(this.resultado); // Emitimos el evento con el
nuevo resultado
      this.actualizarResultadosEnFirestore(URL, resultado);
    },
    error: (error) => {
      console.error('Error en la solicitud:', error);
      this.resultado = 'Error al procesar la URL en R.';
      this.resultadoEvent.emit(this.resultado); // Emitimos el evento con el
mensaje de error
      this.actualizarResultadosEnFirestore(URL, this.resultado);
    },
    complete: () => {
      console.log('Finalizado Procesar');
    },
  });
}

```

```

actualizarResultadosEnFirestore(url: string, resultados: any) {
  // Actualizar el campo 'resultado' en Firestore junto con otros datos
  this._cargaImagenes.actualizarResultadosEnFirestore(resultados, url, this.item);
}

```

```

limpiarP63() {
  this.archivos = [];
  this.imageUrl = null; // Reiniciar la imagen de ejemplo
}

```

```

previewImage(file: File): void {
  const reader = new FileReader();
}

```



```
    reader.onload = (e) => {
      this.imageUrl = e.target?.result;
    };
    reader.readAsDataURL(file);
  }

  onChange(event) {
    this.file = event.target.files[0];
    if (this.file) {
      this.previewImage(this.file);
      this.archivos = [new FileItem(this.file)];
    }
  }

  resultadoKeys() {
    return Object.keys(this.resultado);
  }
}
```

```

<!-- // carga.component.html (Código para interfaz de ventana "Detección" -->
<app-navbar></app-navbar>

<div class="cards">
  <div class="row">
    <div class="container">
      <div class="col-left">
        <div class="card-header">
          <div class="text-center">
            <h4>CARGAR IMAGEN</h4>
          </div>
        </div>
        <label for="fichaMedica">Registro N°:</label>
        <input type="text" class="form-control" id="fichaMedica"
[(ngModel)]= "item.fichaMedica">
        <div class="card-body">
          <input class="form-control" type="file" accept=".jpg"
(change)="onChange($event)"
          [ngClass]='{"file-over": estaSobreElemento}'>
          <div class="well drop-zone"
            appNgDropFiles
            (mouseSobre)="estaSobreElemento=$event"
            [archivos]="archivos"
            [ngClass]='{"file-over": estaSobreElemento}' >
            <div class="text-center">
              <h4>Colocar aquí</h4>
              
            </div>
          </div>
          <h4></h4>
          <div class="text-center">
            <button class="btn btn-outline-success "
              (click)="cargarImagenes()"
              [disabled]="!archivos || archivos.length === 0">Cargar </button>
            <!-- <button class="btn btn-outline-primary "
              (click)="Procesar()"
              [disabled]="!archivos || archivos.length === 0">Procesar
</button> -->
            <button class="btn btn-outline-danger"
              (click)="limpiarP63()">Limpiar</button>
          </div>
          <hr>
          <table class="table">
            <thead class="thead-dark">
              <tr>
                <th>Nombre</th>
                <th>Progreso</th>
              </tr>
            </thead>
            <tbody>

```

```

    <tr *ngFor="let archivo of archivos">
      <td>{{archivo.nombreArchivo}}</td>
      <td>
        <div class="progress">
          <div class="progress-bar"
            role="progressbar"
            [ngStyle]="{'width': archivo.progreso + '%'}"></div>
        </div>
      </td>
    </tr>
  </tbody>
</table>
<div class="text-center">
  <div class="card-img-top">
    <img [src]="imageUrl" alt="ejemplo" *ngIf="imageUrl" >
  </div>
</div>
</div>
</div>
</div>
</div>
</div>
<div class="cards">
  <div class="row">
    <div class="container">
      <div class="col-left">
        <div class="card-header">
          <div class="text-center">
            <h4>RESULTADOS</h4>
          </div>
        </div>
        <div class="card-body">
          <div *ngIf="resultado">
            <p>Resultado:</p>
            <p>{{ resultado }}</p>
          </div>
          <div *ngIf="!resultado">
            <p>Resultado:</p>
          </div>
          <div class="form-group">
            <div class="card">
              <div class="card-header">
                <h4>Registrar Paciente</h4>
                <label for="nombre">Nombre Completo:</label>
                <input type="text" class="form-control" id="nombre"
[(ngModel)]="item.nombre">
                <label for="fichaMedica">Ficha Medica:</label>
                <td type="text" class="form-control">{{item.fichaMedica}}</td>
                <label for="edad">Edad:</label>
                <input type="number" class="form-control" id="edad"

```



```

// fotos.component.ts (Código para ventana "Documentos")
import { Component, OnInit, RendererFactory2 } from '@angular/core';
import { Router } from '@angular/router';
import { PacientesService } from 'src/app/services/pacientes.service';
import { AngularFirestore } from '@angular/fire/compat/firestore';
import { AngularFirestoreDocument } from '@angular/fire/compat/firestore'; //
Asegúrate de importar AngularFirestoreDocument
import { CargaImagenesService } from 'src/app/services/carga-imagenes.service';
import { CargaComponent } from 'src/app/components/carga/carga.component';

export interface Paciente {
  resultado: string;
  nombre: string;
  edad: number;
  fichaMedica: string;
  url: string;
}
export let item: Paciente = {
  resultado: '',
  nombre: '',
  edad: 0,
  fichaMedica: '',
  url: '',
};
export let records = {
  resultado: '',
  nombre: '',
  edad: 0,
  fichaMedica: '',
  url: '',
};
@Component({
  selector: 'app-fotos',
  templateUrl: './fotos.component.html',
  styles: [],
})
export class FotosComponent implements OnInit {
  buscarTexto: string = '';
  records: any[] = [];
  resultadoFromCargaComponent: any;
  constructor(
    private router: Router,
    private _pacienteService: PacientesService,
    private firestore: AngularFirestore,
    private cargaImagenesService: CargaImagenesService
  ) {}
  onResultadoReceived(resultado: any) {
    this.resultadoFromCargaComponent = resultado;
  }
}

```

```

ngOnInit() {
  this.firestore.collection('img').valueChanges().subscribe(records =>
this.records = records);
}

verPacientes(ind: number) {
  this.router.navigate(['/paciente', ind]);
}

buscarPaciente(txt:string) {
  this.buscarTexto=txt.trim();
  if (!this.buscarTexto || this.buscarTexto.length === 0) {
    // Si no hay texto de búsqueda, muestra todos los pacientes
    this.records = this.records;
  } else {
    // Filtra los pacientes que coincidan con el texto de búsqueda en nombre,
ficha médica o edad
    const textoBusqueda = this.buscarTexto.toLowerCase();
    this.records = this.records.filter((paciente: any) => {
      //return (
        var nombre:string=paciente.nombre.toLowerCase();
        var ficha:string=paciente.fichaMedica.toLowerCase();
        var edad:string=paciente.edad.toString().toLowerCase();
        //paciente.edad.toString().includes(textoBusqueda)
        if(nombre.indexOf(textoBusqueda)>=0 || ficha.indexOf(textoBusqueda)>=0 ||
edad.indexOf(textoBusqueda)>=0) return paciente;
        //
      //));
    });
    console.log('pacientes:',this.records);

  }
}

deleteRecord(record: Paciente){
  this.cargaImagenesService.eliminarImagen(record);
}
}

```

```

// carga-imagenes.service.ts
import { Injectable } from '@angular/core';
import { AngularFireStorage } from '@angular/fire/compat/storage';
import { AngularFirestore } from '@angular/fire/compat/firestore';
import { AngularFireDatabase } from '@angular/fire/compat/database';
import { FileItem } from '../Models/file-item';
import { Paciente } from '../components/fotos/fotos.component';
import { HttpClient } from '@angular/common/http';

@Injectable({
  providedIn: 'root'
})
export class CargaImagenesService {
  private CARPETA_IMAGENES = 'img';

  constructor(
    private db: AngularFirestore,
    private db2: AngularFireDatabase,
    private storage: AngularFireStorage,
    private http: HttpClient
  ) {}

  private guardarImagen(imagen: {
    nombre: string,
    url: string,
    edad: number,
    fichaMedica: string,
    resultado: any
  }) {
    this.db.collection(this.CARPETA_IMAGENES).doc(imagen.fichaMedica).set(imagen).then(() => {
      console.log('Documento creado con nombre de archivo:', imagen.fichaMedica);
    }).catch(error => {
      console.error('Error al crear documento:', error);
    });

    this.db2.list(this.CARPETA_IMAGENES).push(imagen).then(ref2 => {
      console.log('GuardarImagen en DATABASE, ref2:', ref2);
    });

    this.enviarUrlAlServidor(imagen.url);
  }

  enviarUrlAlServidor(url: string) {
    this.http.post('http://localhost:8000/carga', url, { responseType: 'text' })
      .subscribe(
        response => {
          console.log('Respuesta del servidor:', response);
        },
      ),
  }
}

```

```

        error => {
            console.error('Error al enviar la URL al servidor:', error);
        }
    );
}

eliminarImagen(imagen: {
    nombre: string,
    url: string,
    edad: number,
    fichaMedica: string,
    paciente?: Paciente,
    resultado: string
}) {
    this.db
        .collection(this.CARPETA_IMAGENES, (ref) =>
            ref
                .where('nombre', '==', imagen.nombre)
                .where('url', '==', imagen.url)
                .where('edad', '==', imagen.edad)
                .where('fichaMedica', '==', imagen.fichaMedica)
                .where('resultado', '==', imagen.resultado)
                .limit(1)
        )
        .get()
        .subscribe((querySnapshot) => {
            if (!querySnapshot.empty) {
                const docId = querySnapshot.docs[0].id;
                this.db.collection(this.CARPETA_IMAGENES).doc(docId).delete().then(() => {
                    console.log('Imagen eliminada de Firestore');
                }).catch(error => {
                    console.error('Error al eliminar la imagen de Firestore:', error);
                });
            } else {
                console.log('No se encontró ninguna imagen para eliminar.');
```

```

            }
        });
}

cargarImagenesFirebase(imagenes: FileItem[], item: Paciente, ) {
    for (const img63 of imagenes) {
        img63.estaCargando = true;
        if (img63.progreso >= 100) {
            continue;
        }
        const filePath = `${this.CARPETA_IMAGENES}/${img63.nombreArchivo}`;
        const storageRef = this.storage.ref(filePath);
        const uploadTask = this.storage.upload(filePath, img63.archivo);

        uploadTask.percentageChanges().subscribe({
```



```

next: (snapShot_data) => {
  img63.progreso = snapShot_data;
},
error: err => {
  console.log('ERROR subiendo archivo:', err);
},
complete: () => {
  storageRef.getDownloadURL().subscribe(download_URL => {
    img63.url = download_URL;

    this.guardarImagen({
      nombre: item.nombre,
      url: img63.url,
      edad: item.edad,
      fichaMedica: item.fichaMedica,
      resultado: item.resultado
    });
    img63.estaCargando = false;
  });
}
});
}
}
}

```

```

actualizarResultadosEnFirestore(resultados: any, url: string, item: Paciente) {
  // Actualizar el campo 'resultado' en Firestore junto con otros datos
  this.guardarImagen({
    nombre: item.nombre,
    url: url,
    edad: item.edad,
    fichaMedica: item.fichaMedica,
    resultado: resultados
  });
}
}
}

```

D. Código de Interoperabilidad

```
# plumber.R
library(jsonlite)
library(plumber)
library(urltools)
library(dplyr)
library(imager)

options("plumber.port" = 5555)
# Filtro para habilitar CORS
#* @filter cors
cors <- function(res, req) {
  res$setHeader('Access-Control-Allow-Origin', '*') # Or whatever
  cat("Inicio de Servidor\n")

  if (req$REQUEST_METHOD == "OPTIONS") {
    res$setHeader("Access-Control-Allow-Methods", "*")
    res$setHeader("Access-Control-Allow-Headers",
req$HTTP_ACCESS_CONTROL_REQUEST_HEADERS)
    res$status <- 200
    return(list())
  } else {
    plumber::forward()
  }
}

#* @preempt cors
#* @post /predecir
function(res, req) {
  # Obtener la URL del cuerpo de la solicitud
  url <- req$body$URL2
  header <- req$header

  # Imprimir la URL para propósitos de depuración
  cat("Cuerpo de la solicitud:", toJSON(req$body), "\n")
  cat("URL recibida:", url, "\n")
  cat("cabecera recibida:", header, "\n")

  url <-url
# Descargar el archivo desde la URL
download.file(url, 'url.jpg', mode = 'wb')

# Leer la imagen
image <- load.image("url.jpg")

# Convertir la imagen a escala de grises
image_grayscale <- grayscale(image)

# Convertir la imagen a un dataframe
image_df <- as.data.frame(image_grayscale)
```

```

# Extraer los valores de píxeles como una lista
csa1 <- lapply(image_df, function(x) as.matrix(x))

# Convertir la lista en un dataframe
csa2 <- as.data.frame(csa1)

# Convertir el dataframe a un vector
vector_pixeles <- as.vector(t(csa2$value))

# Cargar modelo y matriz
model <- readRDS("./modelo_entrenado.rds")
testprueba <- read.csv("./datos.num.csv", header = TRUE)

# Añadir el vector a los datos de prueba (testprueba)
testprueba <- rbind(testprueba, as.list(vector_pixeles))

# Realizar PCA

datos.num <- testprueba
datos.num <- as.data.frame(lapply(datos.num, as.numeric))

pcx <- prcomp(datos.num)

# Predecir
# para_predecir <- as.numeric(tail(pcx$x, 1))
# prediccion <- predict(model, newdata = as.data.frame(para_predecir))
topred <- data.frame(nueva=(tail(pcx$x,1)))
prediccion <- predict(model,newdata=topred)
# Mostrar la respuesta
print(prediccion)

res <- paste0("El paciente tiene ",round(100*prediccion[2],2)," % de probabilidad de
presentar artrosis de rodilla con un 80% de exactitud")

# Preparar la respuesta
response <- list(
  data = res
)

# Imprimir la respuesta para propósitos de depuración
cat("Respuesta preparada:", toJSON(response), "\n")

# Devuelve la respuesta en formato JSON
return(response)
}

```

VALIDACIÓN DEL PROYECTO: PROCESAMIENTO DE IMÁGENES PARA DETECCIÓN DE OSTEoarTRITIS DE RODILLA UTILIZANDO IA

El formulario que se presenta a continuación desempeña un papel fundamental en la verificación de la pertinencia de una interfaz capaz de recibir imágenes radiológicas. A través de esta interfaz, se logra detectar con una precisión del 80% la presencia de artrosis de rodilla, al mismo tiempo que se almacena la información correspondiente al paciente.

Agradecemos sinceramente su tiempo y esfuerzo al participar en esta investigación, ya que su colaboración contribuirá de manera significativa a la validación de este estudio.

* Indica que la pregunta es obligatoria

1. Nombre Completo *

2. Edad *

3. Género *

4. Profesión *

5. Especialización/Posgrado *

6. Años de experiencia profesional *

7. Años de experiencia en el ámbito de diagnóstico o intervención en artrosis de rodilla *

8. ¿Qué tan útil considera la inteligencia artificial en el ámbito clínico para mejorar la eficacia de los diagnósticos médicos en general? *

Marca solo un óvalo.

- Muy en desacuerdo.
- En desacuerdo.
- Ni de acuerdo ni en desacuerdo.
- De acuerdo.
- Muy de acuerdo.

9. ¿Considera que la aplicación de métodos cuantitativos y analíticos en la investigación médica contribuye a la confiabilidad de los resultados? *

Marca solo un óvalo.

- Muy en desacuerdo.
- En desacuerdo.
- Ni de acuerdo ni en desacuerdo.
- De acuerdo.
- Muy de acuerdo.

10. ¿Considera que el registro de usuario en la interfaz es esencial para la personalización y seguimiento efectivo de pacientes en aplicaciones médicas? *

Marca solo un óvalo.

- Muy en desacuerdo.
- En desacuerdo.
- Ni de acuerdo ni en desacuerdo.
- De acuerdo.
- Muy de acuerdo.

11. ¿Considera útil la inclusión de la pesaña para explicar el formato de las imágenes que se cargará en la plataforma para realizar la clasificación del tipo de lesión que presenta la radiografía? *

Marca solo un óvalo.

- Muy en desacuerdo.
- En desacuerdo.
- Ni de acuerdo ni en desacuerdo.
- De acuerdo.
- Muy de acuerdo.

12. ¿Considera que el sistema realiza una clasificación precisa de la carga de imágenes y la recopilación de información del paciente en la segunda pestaña de la interfaz son procesos fluidos? *

Marca solo un óvalo.

- Muy en desacuerdo.
- En desacuerdo.
- Ni de acuerdo ni en desacuerdo.
- De acuerdo.
- Muy de acuerdo.

13. ¿Considera que la visualización de datos ingresados en la tercera pestaña de la interfaz es útil para los profesionales médicos en la evaluación de casos? *

Marca solo un óvalo.

- Muy en desacuerdo.
- En desacuerdo.
- Ni de acuerdo ni en desacuerdo.
- De acuerdo.
- Muy de acuerdo.

14. ¿Está de acuerdo en que la combinación de procesamiento de imágenes y redes neuronales proporciona resultados más confiables que otros enfoques convencionales? *

Marca solo un óvalo.

- Muy en desacuerdo.
- En desacuerdo.
- Ni de acuerdo ni en desacuerdo.
- De acuerdo.
- Muy de acuerdo.

15. ¿Cree que el uso de redes neuronales en el análisis de imágenes médicas es una opción prometedora para mejorar la detección de enfermedades como la osteoartritis? *

Marca solo un óvalo.

- Muy en desacuerdo.
- En desacuerdo.
- Ni de acuerdo ni en desacuerdo.
- De acuerdo.
- Muy de acuerdo.

16. ¿En qué medida está de acuerdo en que la aplicación de inteligencia artificial puede contribuir significativamente a la detección temprana de la artrosis de rodilla? *

Marca solo un óvalo.

- Muy en desacuerdo.
- En desacuerdo.
- Ni de acuerdo ni en desacuerdo.
- De acuerdo.
- Muy de acuerdo.

17. ¿En qué medida confía en la capacidad de la inteligencia artificial para diferenciar entre casos de artrosis de rodilla y condiciones similares en imágenes médicas? *

Marca solo un óvalo.

- Muy en desacuerdo.
- En desacuerdo.
- Ni de acuerdo ni en desacuerdo.
- De acuerdo.
- Muy de acuerdo.

18. ¿Qué tan importante considera que es la colaboración entre profesionales de la salud y sistemas de inteligencia artificial para optimizar el tratamiento de la artrosis de rodilla? *

Marca solo un óvalo.

- Muy en desacuerdo.
- En desacuerdo.
- Ni de acuerdo ni en desacuerdo.
- De acuerdo.
- Muy de acuerdo.

19. En términos de accesibilidad, ¿cree que la inteligencia artificial puede brindar apoyo en el diagnóstico presuntivo de la artrosis de rodilla a fin de atender a un mayor número de personas? *

Marca solo un óvalo.

- Muy en desacuerdo.
- En desacuerdo.
- Ni de acuerdo ni en desacuerdo.
- De acuerdo.
- Muy de acuerdo.

20. ¿Qué tan confiado se siente en la capacidad de la inteligencia artificial para adaptarse y mejorar sus capacidades de diagnóstico a medida que se acumula más información clínica? *

Marca solo un óvalo.

- Muy en desacuerdo.
- En desacuerdo.
- Ni de acuerdo ni en desacuerdo.
- De acuerdo.
- Muy de acuerdo.

21. ¿Qué tan optimista es sobre la posibilidad de que la inteligencia artificial desempeñe un papel en el desarrollo de tratamientos más personalizados y eficaces para la artrosis de rodilla? *

Marca solo un óvalo.

- Muy en desacuerdo.
- En desacuerdo.
- Ni de acuerdo ni en desacuerdo.
- De acuerdo.
- Muy de acuerdo.

22. En términos de costo-beneficio, ¿cree que la implementación de inteligencia artificial en el diagnóstico de la artrosis de rodilla sería económicamente viable? *

Marca solo un óvalo.

- Muy en desacuerdo.
- En desacuerdo.
- Ni de acuerdo ni en desacuerdo.
- De acuerdo.
- Muy de acuerdo.

23. ¿Cuál es su percepción sobre la aceptación y confianza de los pacientes hacia el uso de inteligencia artificial en el diagnóstico y tratamiento de la artrosis de rodilla? *

Marca solo un óvalo.

- Muy en desacuerdo.
- En desacuerdo.
- Ni de acuerdo ni en desacuerdo.
- De acuerdo.
- Muy de acuerdo.

24. ¿Considera que el tiempo de respuesta de esta herramienta para procesar imágenes en la detección de osteoartritis de rodilla utilizando IA es adecuado? *

Marca solo un óvalo.

- Muy en desacuerdo.
- En desacuerdo.
- Ni de acuerdo ni en desacuerdo.
- De acuerdo.
- Muy de acuerdo.

25. ¿Encuentra que la herramienta es fácil de usar para llevar a cabo el procesamiento de imágenes en la detección de osteoartritis de rodilla utilizando IA? *

Marca solo un óvalo.

- Muy en desacuerdo.
- En desacuerdo.
- Ni de acuerdo ni en desacuerdo.
- De acuerdo.
- Muy de acuerdo.

26. ¿Los resultados obtenidos de esta herramienta en el procesamiento de imágenes para la detección de osteoartritis de rodilla utilizando IA son claramente presentados y clasificados de manera comprensible? *

Marca solo un óvalo.

- Muy en desacuerdo.
- En desacuerdo.
- Ni de acuerdo ni en desacuerdo.
- De acuerdo.
- Muy de acuerdo.

27. ¿Basándose en su experiencia con la interfaz, considera que esta herramienta *
para la detección de osteoartritis de rodilla mediante procesamiento de imágenes
es fácil de usar?

Marca solo un óvalo.

- Muy en desacuerdo.
- En desacuerdo.
- Ni de acuerdo ni en desacuerdo.
- De acuerdo.
- Muy de acuerdo.

Este contenido no ha sido creado ni aprobado por Google.

Google Formularios