



TECNOLÓGICO NACIONAL DE MÉXICO
INSTITUTO TECNOLÓGICO DE ACAPULCO

DISEÑO DE UNA HERRAMIENTA WEB PARA MANIPULAR IMÁGENES
MÉDICAS

TESIS PROFESIONAL

QUE PARA OBTENER EL TÍTULO DE:
MAESTRO EN SISTEMAS COMPUTACIONALES

PRESENTA:
ING. MANUEL NIÑO NAVARRETE

DIRECTOR DE TESIS:
DR. JOSÉ ANTONIO MONTERO VALVERDE

CODIRECTOR DE TESIS:
M.T.I RAFAEL HERNÁNDEZ REYNA

ACAPULCO, GRO., DICIEMBRE 2019.

El presente trabajo de tesis fue desarrollado en la *División de Estudios de Posgrado e Investigación* del *Instituto Tecnológico de Acapulco*, perteneciente al Programa Nacional de Posgrados de Calidad (PNPC-CONACYT).

Con domicilio para recibir y oír notificaciones en Av. Instituto Tecnológico de Acapulco s/n, Crucero del Cayaco, Acapulco, Guerrero, México. C.P. 39905.

Becario:	Manuel Niño Navarrete
CVU:	851632
Número de apoyo:	627032
Grado:	Maestría



Descargo de responsabilidad Institucional

El que suscribe declara que el presente documento titulado “Diseño de una Herramienta Web para Manipular Imágenes Médicas” es un trabajo propio y original, el cual no ha sido utilizado anteriormente en institución alguna para propósito de evaluación, publicación y/o obtención de algún grado académico.

Además, se han recogido todas las fuentes de información utilizadas, las cuales han sido citadas en la sección de referencia bibliográficas de este trabajo.

Nombre: Ing. Manuel Niño Navarrete fecha y firma: _____

Dedicatoria

Todo este esfuerzo está dedicado a mi madre Estela, a pesar de no tener las mismas oportunidades de educación que yo he tenido, siempre me ha inculcado que la educación es una pieza importante en la vida y me ha dicho que lo mejor que me puede heredar es tener una buena educación, porque siempre me ha apoyado en todo momento y nunca ha dejado de creer en mí, porque ha sido mi inspiración en todo momento, este logro también es tuyo mamá.

Agradecimientos

A mis padres, Margarito y Estela

Gracias por sus consejos, el apoyo y el amor que me han dado a lo largo de mi vida, porque sin ello no lo hubiera podido lograr.

A mi asesor, Dr. José Antonio Montero Valverde

Agradezco por su tiempo, dedicación, enseñanza y paciencia que tuvo conmigo, gracias a ello pude finalizar este trabajo, mis respeto y admiración hacia usted.

Al CONACYT.

Agradezco a este organismo por el apoyo económico que me han brindado para poder concluir satisfactoriamente este grado.

A la banda Abejorro

Agradezco a mis amigos y compañeros de la generación, por el apoyo, convivencias y todos los momentos de estrés que pasamos y salimos adelante.

A los docentes del posgrado

Agradezco por las enseñanzas y dedicación que tuvieron conmigo.

Al IECan, departamento de informática.

Agradezco al jefe de informática el Ing. Israel Figueroa Palacios por el buen trato y abrirme las puertas y permitirme realizar este proyecto, por tener disposición en todo momento que surgieron dudas.

Resumen

El análisis de imágenes médicas se ha convertido en la base de los radiólogos y/o expertos médicos para emitir un diagnóstico de posibles enfermedades de pacientes. La calidad de la imagen analizada es determinante en la localización de regiones de interés y su posterior diagnóstico. El uso de aplicaciones informáticas que permiten la manipulación de tales imágenes es una herramienta de gran apoyo para el personal médico que realiza diagnósticos. El presente trabajo se realizó en el Instituto Estatal de Cancerología del estado de Guerrero, teniendo como objetivo desarrollar una herramienta web que apoye a los médicos a la manipulación de imágenes médicas. Dicha plataforma cuenta con diferentes funcionalidades para la manipulación de imágenes permitiendo el uso de diferentes técnicas para su mejoramiento, transmisión de imágenes, entre otras.

Abstract

The analysis of medical images has become the basis of radiologists and/or medical experts to issue a diagnosis of possible diseases of patients. The quality of the image analyzed is decisive in locating regions of interest and their subsequent diagnosis. The use of computer applications that allow the manipulation of such images is a tool of great support for medical specialist who performs diagnostics. The present work has been made in the Institute of Cancerology of Guerrero State, aims to develop a web platform that supports radiologist to manipulate medical images. This platform provides several functionalities, such as: image processing by applying different types of techniques for its improvement, allowing to use annotations and manipulation of the improved image, among others.

Índice de contenido

Dedicatoria.....	iii
Agradecimientos.....	iv
Resumen.....	vi
Abstract.....	vii
Capítulo 1. Introducción	1
1.1 Antecedentes	1
1.2 Planteamiento del problema.....	4
1.3 Objetivos	6
1.3.1 General	6
1.3.2 Específicos	6
1.4 Hipótesis	7
1.5 Justificación	7
Capítulo 2. Marco teórico.....	10
2.1 Formato DICOM.....	10
2.2 Procesamiento y análisis de imágenes digitales	15
2.3 Conformación de una imagen	16
2.4 Filtrado	18
2.5 Arquitectura Cliente-Servidor	23
2.6 ASP.NET Core MVC	24
2.6.1 HTML	27
2.6.2 CSS.....	27
2.6.3 JavaScript.....	28
2.6.4 C#	29
2.7 Metodología de desarrollo de Software.....	30
2.7.1 Kanban	30
Capítulo 3. Estado del arte.....	34
3.1 Plataformas existentes para la manipulación de imágenes	35
3.2 Herramientas existentes para el desarrollo de sistemas PACS.....	40
3.3 Metodología y arquitectura de sistemas PACS	43
Capítulo 4. Desarrollo e implementación.....	54
4.1 Levantamiento de requerimientos.	56
4.2 MockUp/Diseño de la Aplicación.....	62
4.2.1 Arquitectura de la herramienta Software	63

4.3 Desarrollo e implementación.....	65
4.3.1 Servidor	66
4.3.2 MiddleWare	71
4.3.3 App Entity	73
4.3.3.1 BackEnd	74
4.3.3.2 FrontEnd	76
4.4 Implementación	79
4.4.1 Configuración del Servidor.....	79
4.4.2 Generación de bibliotecas de enlace dinámico(DLL) para producción.....	87
Capítulo 5. Pruebas y resultados	91
5.1 Pruebas de conectividad	91
5.2 Pruebas de usabilidad	93
5.3 Pruebas de funcionalidad	98
5.4 Conclusiones	109
Bibliografía	111

Índice de figuras

Figura 2.1. Estructura de una carpeta DICOM (Leadtools, 2017).....	11
Figura 2.2. Estructura de un elemento de datos (Bojchko & Kroker, 2015).....	12
Figura 2.3. a) Imagen original. b) Fracción pequeña de la imagen original en píxeles. c) valores de la intensidad de píxeles.	17
Figura 2.4. Representación de imagen en matriz.	17
Figura 2.5. Representación de imagen en píxeles. a) Imagen original pixelada. b) Imagen pixelada con valores de los píxeles. c) Valores de intensidad de los píxeles (Yeung & Alahi, 2015).....	18
Figura 2.6. Imagen filtrada con algoritmo de Canny.	21
Figura 2.7. Filtro de paso bajo, a) Imagen original con ruido. b) Imagen filtrada con Gaussian filter.	22
Figura 2.8. Pixel que se procesa en aplicación de filtro.....	23
Figura 2.9. Modelo Cliente-Servidor.	23
Figura 2.10. Diagrama MVC.	26
Figura 2.11. Tablero Kanban.	32
Figura 3.1. Primera radiografía de rayos X.....	34
Figura 3.2. Modelo de Integración.....	46
Figura 3.3. Escenario de interconexión e interoperabilidad de RSB.	46
Figura 3.4. Módulos de patrones de recuperación.	50
Figura 4.1. Flujo de trabajo Kanban.	54
Figura 4.2. Diagrama de bloques.	55
Figura 4.3. Sharpen filter.	57
Figura 4.4. Filtrado de imagen. a) Imagen original, b) Resultado del Sharpen Filter.	58
Figura 4.5. Emboss filter.....	58
Figura 4.6. Filtrado de imagen. a) Imagen original, b) Resultado del Emboss Filter.....	58
Figura 4.7. Edge Detection Filter.....	59
Figura 4.8. Filtrado de imagen. a) Imagen original, b) Resultado del Edge Detection Filter.....	59
Figura 4.9. High-Pass Filter.	59
Figura 4.10. Filtrado de imagen. a) Imagen original, b) Resultado del High Pass Filter.	60
Figura 4.11. Unsharp Filter.	60
Figura 4.12. Filtrado de imagen. a) Imagen original, b) Resultado del Edge Unsharp Filter.....	60
Figura 4.13. Gaussian Blur.	61
Figura 4.14. Filtrado de imagen. a) Imagen original, b) Resultado del Gaussian Blur.	61
Figura 4.15. Box Blur.	61
Figura 4.16. Filtrado de imagen. a) Imagen original, b) Resultado del Box Blur.	62
Figura 4.17. Diagrama de despliegue.	63
Figura 4.18. Arquitectura de la Herramienta Web.....	64
Figura 4.19. Interfaz web de DCM4CHEE.....	70
Figura 4.20. Instalación de la librería Fo Dicom.	72
Figura 4.21. Configuración de la conexión.....	73
Figura 4.22. Código de la creación de la base de datos SQLite.....	75
Figura 4.23. Código del modelo de la base de datos SQLite.....	76
Figura 4.24. Código de la interfaz principal.	77
Figura 4.25. Interfaz principal del sistema.....	78

Figura 4.26. Interfaz de la herramienta para manipular imágenes.....	79
Figura 4.27. Agregar roles al servidor.	83
Figura 4.28. Selección del rol Servidor web(IIS).	83
Figura 4.29. Selección del servicio de rol ASP.NET.....	84
Figura 4.30. Servicios del rol Servidor web(IIS) que se instalarán.	85
Figura 4.31. Confirmación que el Servidor web (IIS) se haya instalado.....	86
Figura 4.32. Versión de ASP.NET Core donde se desarrolló la aplicación.	86
Figura 4.33. Descargar las aplicaciones Runtime para ejecutar la aplicación.	87
Figura 4.34. Limpieza del proyecto.	88
Figura 4.35. Compilación del proyecto.....	88
Figura 4.36. Generación de DLL.	89
Figura 4.37. Dirección donde se generan los DLL.	89
Figura 4.38. Almacenando los archivos DLL en la carpeta wwwroot del servidor.....	90
Figura 4.39. Herramienta web funcionando.	90
Figura 5.1. Configuración de conexión al servidor.....	92
Figura 5.2. Lista de imágenes de pacientes almacenadas en el servidor.	93
Figura 5.3. Graficas de resultados de la evaluación.....	96
Figura 5.4. Imagen médica visualizada en la herramienta.....	99
Figura 5.5. Información de una imagen médica.	100
Figura 5.6. Funcionalidad Enfocar de la herramienta.....	101
Figura 5.7. Funcionalidad Nivel B/C de la herramienta.	102
Figura 5.8. Funcionalidad Desplazar de la herramienta.	103
Figura 5.9. Funcionalidad Anotar de la herramienta.	103
Figura 5.10. Funcionalidad Invertir de la herramienta.	104
Figura 5.11. Funcionalidad Magnificar de la herramienta.....	105
Figura 5.12. Funcionalidad Interpolación de la herramienta.	105
Figura 5.13. Funcionalidad Marcar ROI de la herramienta.	106
Figura 5.14. Tipos de procesamiento de imagen en la herramienta.....	107
Figura 5.15. Procesamiento de detección de bordes en la imagen.....	108
Figura 5.16. Procesamiento de suavizado en la imagen.	109

Índice de tablas

Tabla 1.1 Dosis de radiación que son expuestos los pacientes (Europea, 2001).....	8
Tabla 2.1. Registros de elementos de datos DICOM (Digital Imaging and Communications in Medicine (DICOM), 2017).	13
Tabla 4.1. Tabla de servicios DCM4CHEE.....	67
Tabla 4.2 Tablas de la base de datos DCM4CHEE.	68
Tabla 4.3. Características del servidor HP.....	80
Tabla 5.1. Criterios de usabilidad de la herramienta web.....	93
Tabla 5.2. Tabla de resultados de la evaluación.	95

Capítulo 1. Introducción

1.1 Antecedentes

El análisis de las imágenes médicas se ha convertido en una parte integral del diagnóstico y tratamiento del paciente. Su utilización ya no queda relegada únicamente al ámbito de la Radiología, sino que cada vez es más común utilizar elementos de computación basados en imagen en el proceso previo a la cirugía (como el caso de VirSSPA¹) e incluso durante las propias intervenciones quirúrgicas, en las que los facultativos se valen de estas herramientas para tomar decisiones en tiempo real o para tomar imágenes y muestras que posteriormente soportarán el diagnóstico y la terapia recomendada (IMÁGENES MÉDICAS, 2008).

La radiología es una rama de la medicina que utiliza la tecnología imagenológica para diagnosticar y tratar una enfermedad.

La radiología diagnóstica ayuda a los profesionales de la salud observar estructuras dentro del cuerpo humano. Los médicos que se especializan en la interpretación de estas imágenes se conocen como radiólogos de diagnóstico. Mediante estas imágenes, el radiólogo u otros médicos con frecuencia pueden:

- Diagnosticar la causa de sus síntomas.
- Vigilar qué tan bien está respondiendo una persona a un tratamiento que está recibiendo para su enfermedad o afección.
- Detectar diferentes enfermedades, como cáncer de mama, cáncer de colon o cardiopatía.

¹ VirSSPA - herramienta de realidad virtual para el flujo de trabajo de planificación quirúrgica.

La radiología intervencionista es una subespecialidad de la radiología en la que se utilizan técnicas de imagen para ejecutar procedimientos mínimamente invasivos. Los radiólogos intervencionistas son médicos que utilizan imágenes tales como tomografía computarizada (TC), ecografía, resonancia magnética (RM) y fluoroscopia para ayudar a guiar los procedimientos. Las imágenes son útiles para el médico al introducir catéteres (sondas), alambres y otros instrumentos y herramientas pequeñas en el cuerpo. Esto particularmente se considera para incisiones (cortes) pequeñas. Los médicos pueden usar esta tecnología para diagnosticar o tratar afecciones en casi cualquier parte del cuerpo humano, en lugar de tener que mirar directamente dentro del mismo a través de un endoscopio (cámara) o con una cirugía abierta. Los radiólogos intervencionistas con frecuencia están involucrados en el tratamiento de cánceres o tumores, bloqueos en arterias y venas, miomas uterinos, dolor de espalda, problemas hepáticos y renales (MedlinePlus, 2019).

Existe una gama de modalidades de adquisición de imágenes médicas que ayuda a adquirir información vital estructural y funcional para la salud del paciente. El procesamiento apropiado y la visualización de dicha información juegan un papel importante en ayudar a un diagnóstico eficiente y preciso (Ukis et al., 2013).

En la actualidad existen diferentes tipos de patologías cancerígenas que se pueden diagnosticar mediante el análisis de radiografías. El diagnóstico por imágenes permite a los médicos observar el interior del cuerpo humano para buscar indicios sobre una condición médica. Una variedad de aparatos y técnicas pueden crear imágenes de las estructuras y actividades dentro del mismo (MedlinePlus, 2018). El tipo de imagen que use el médico dependerá de los síntomas y de la parte del cuerpo que debe examinarse, a continuación, se mencionan las fuentes más comunes para la generación de imágenes médicas:

- a) *Rayos X* - Imagen por Rayos X, donde un emisor externo al paciente emite una fuente de rayos X en dirección al mismo, para que un sensor situado tras el organismo detecte los rayos X que han conseguido atravesarlo. Mediante la utilización de este sistema se pueden obtener imágenes planares, pero a pesar de estar muy extendido en la práctica médica, debido fundamentalmente a su bajo coste, presenta el inconveniente de que sólo se visualiza una proyección del objeto. En los últimos años, la tendencia en este tipo de equipos ha estado marcada en un esfuerzo por obtener imágenes digitales, que puedan ser almacenadas, analizadas y transferidas mediante un sistema informático (ImagenYON, 2018).
- b) *Tomografías computarizadas* - La tomografía computarizada (TC) es una tecnología para diagnóstico con imágenes. Utiliza un equipo de rayos X especial para crear imágenes transversales del cuerpo. Entre los usos de la TC se incluye la exploración de:
- Huesos fracturados.
 - Cánceres.
 - Coágulos de sangre.
 - Signos de enfermedad cardiaca.
 - Hemorragia interna.
- c) *Imágenes por resonancia magnética* - Los estudios con imágenes por resonancia magnética (IRM) usan un gran imán y ondas de radio para observar órganos y estructuras que se encuentran al interior del cuerpo. Los profesionales de la salud utilizan estas imágenes para diagnosticar una variedad de afecciones, desde

rupturas de ligamentos hasta tumores. Las imágenes por resonancia magnética son muy útiles para examinar el cerebro y la médula espinal.

- d) *Mamografía* - Una mamografía es un tipo especial de radiografía de las mamas. Para la mamografía se utiliza radiación ionizante a fin de obtener imágenes del tejido mamario. El examen se lleva a cabo mediante la compresión firme de la mama entre dos placas; esta compresión despliega los tejidos superpuestos y reduce la cantidad de radiación necesaria para tomar la imagen.
- e) *Ecografía*, que utiliza el sonido como forma de examinar al paciente. Un pequeño transductor emite sonidos de muy alta frecuencia (ultrasonidos) que se reflejan (ecos) en las distintas estructuras del cuerpo humano con mayor o menor intensidad, en función de la densidad del medio que atraviesa. Dichos ecos son detectados por el propio transductor que envía la información recogida a un ordenador que se encarga de reconstruir la imagen.

1.2 Planteamiento del problema

Con el paso del tiempo la tecnología ha ido evolucionando, y para adaptarse a esa evolución las herramientas también tienen que ajustarse y adaptarse a esos cambios. Uno de estos cambios consiste en eliminar el almacenamiento de información tradicional, donde todo se archiva y se guarda de manera física, las impresiones de radiografías pueden ser costosas y ocasionar una gran pérdida de tiempo al momento de imprimir una gran cantidad de imágenes, además de que los químicos que se obtienen al revelar las imágenes radiológicas puede causar un serio daño al medioambiente (Ramon, 2013).

Tener una imagen de gran calidad es fundamental para realizar un diagnóstico certero, sin embargo, existen diferentes factores que pueden ocasionar que no se obtenga una imagen con buena calidad, la mala práctica de los principios metódicos y ordenados que deben realizarse en el cuarto oscuro, originan irremediablemente, radiografías defectuosas: manchadas, mal reveladas, mal fijadas o mal enjuagadas, raspaduras en la emulsión o películas pegadas entre sí (González Pérez et al., 2009). Los defectos de las radiografías pueden clasificarse según sus causas.

- *Falta de revelado* - Es un defecto muy frecuente cuando se realiza de forma manual, donde se persiste en revelar sin medir la temperatura de las soluciones, ni usar el tiempo de revelado adecuado (4 min). Las radiografías bien expuestas con factores electrotécnicos adecuados, pueden aparecer blancuzcas, con poco contraste, predominan los tonos blancos y grises (por falta de revelado), faltan los tonos negros.
- *Exceso de revelado* - Radiografía negruzca, sin contraste, con exceso de densidad, predominan los tonos negros y grises oscuros.
- *Velo* - La radiografía aparece ennegrecida, con ausencia de imagen.
- *Manchas* - Las manchas en las radiografías obedecen a diferentes causas y pueden afectar la película radiográfica de forma parcial o total.
- *Marcas de corriente estática* - Se produce al sacar las películas de la caja de protección o de su envoltura de papel, al agrupar varias películas unas sobre otras o a fricción de las mismas con las pantallas intensificadoras (al cargar o descargar los chasis).

Hay una gran cantidad de tipos de cáncer que se atienden en el Instituto Estatal de Cancerología del estado de Guerrero (IECan). Los especialistas del IECan pueden llegar a tener

demasiadas imágenes médicas que tienen que analizar, y no cuentan con una herramienta que les ayude a visualizar y manipular dichas imágenes para poder ofrecer un diagnóstico y así disminuir la carga de trabajo que tienen.

Actualmente en el mercado existen herramientas que apoyan el trabajo de los radiólogos, para utilizar dichas herramientas se necesita pagar una licencia, ya sea mensual o anual, el costo dependerá de varios factores, tales como las conexiones DICOM, capacidad de almacenamiento de información, cantidad de equipos en la cual se utilizará la herramienta y tipos de imágenes DICOM (CR, TC, MRI, etc.) que podrá analizar, el precio oscila alrededor de 5,000 a 100,000 dólares dependiendo de las especificaciones de la herramienta (Systems, 2013), es un precio considerable para una institución pública. Por lo tanto, cualquier aportación en el desarrollo e implementación de este tipo de herramienta web es de gran apoyo en este sector.

1.3 Objetivos

1.3.1 General

Desarrollar una herramienta web, que apoye a los especialistas del sector salud a visualizar y manipular imágenes médicas en formato DICOM desde cualquier sitio físico en que se encuentre.

1.3.2 Específicos

- ❖ Desarrollar una interfaz que permita la visualización de imágenes médicas en formato DICOM.
- ❖ Desarrollar una interfaz gráfica que permita la manipulación de las imágenes médicas con las siguientes funcionalidades:
 - Enfocar.

- Nivel de brillo y contraste.
- Desplazar.
- Anotar.
- Borrar anotaciones.
- Invertir.
- Rotar.
- Ángulo.
- Magnificar.
- Interpolación.
- Restablecer.
- Marcar región de interés (ROI).
- Procesamiento de imagen.

1.4 Hipótesis

El diseño de una herramienta computacional que permita la visualización y la manipulación de imágenes médicas en la web, permitirá que los especialistas puedan realizar diagnósticos desde cualquier sitio físico, aumentando con ello el número de diagnósticos.

1.5 Justificación

Con el desarrollo de la medicina moderna, las actividades de diagnóstico y tratamiento del hospital dependen cada vez más del examen de imágenes médicas (por ejemplo, rayos X, tomografía computarizada, resonancia magnética, ultrasonido, endoscopia, angiografía, etc.). Alrededor del 80% de los datos generados en los hospitales provienen de materiales de imágenes

médicas. Para gestionar los datos masivos generados, muchos hospitales han establecido el Sistema de comunicación y almacenamiento de imágenes (Wan Yanli et al., 2016).

Actualmente los radiólogos del Instituto Estatal de Cancerología del estado de Guerrero (IECan) no cuentan con una herramienta que les permita realizar su trabajo de una forma más flexible. Con el desarrollo de esta herramienta computacional se evitará pérdidas de información, debido a que dicha información se encontrará almacenada y ordenada en un lugar centralizado, además, se evitará pérdidas de recursos, ya que, al estar almacenadas, sólo se imprimirán cuando sean necesarias, los especialistas podrán acceder a la información desde cualquier lugar en que se encuentre, ya sea desde una computadora de escritorio, laptop o dispositivo móvil.

Así mismo con esta herramienta se reducirá la carga de trabajo de los especialistas, debido a que la herramienta ofrecerá con una serie de características para mejorar la calidad de las imágenes radiográficas, y así el especialista podrá dar un diagnóstico en menor tiempo. Con la mejora de la calidad de las imágenes, se evitará que los pacientes vuelvan a ser expuestos a radiación, cuando un paciente se expone a grandes dosis de radiación, puede causar graves daño a los tejidos y aumentar el riesgo de que padezca cáncer más adelante (American Cancer Society, 2016). En la tabla 1.1 se muestra la dosis de radiación que son expuestos los pacientes según sea el estudio, también se muestra el periodo equivalente aproximado a estar expuesto a radiación natural de fondo.

Tabla 1.1 Dosis de radiación que son expuestos los pacientes (Europea, 2001).

Procedimiento	Dosis efectiva características (mSv)	No. Equivalente de RX de tórax	Periodo equivalente aproximado de radiación natural de fondo
<i>Radiografía</i>			
<i>Extremidades y articulaciones</i>			
<i>(excluida la cadera)</i>	< 0,01	< 0,5	< 1,5 días

<i>Tórax simple (postero-anterior)</i>	0,02	1	3 días
<i>Cráneo</i>	0,07	3,5	11 días
<i>Columna dorsal</i>	0,7	35	4 meses
<i>Columna lumbar</i>	1,3	65	7 meses
<i>Cadera</i>	0,3	15	7 semanas
<i>Pelvis</i>	0,7	35	4 meses
<i>Abdomen</i>	1,0	50	6 meses
<i>Urograma excretor</i>	2,5	125	14 meses
<i>Esofagograma</i>	1,5	75	8 meses
<i>Esofagogastroduodenal</i>	3	150	16 mese
<i>Tránsito intestina</i>	3	150	16 meses
<i>Colon por enema</i>	7	350	3,2 años
<i>Tomografías computadas</i>			
<i>TC de cabeza</i>	2,3	115	1 año
<i>TC de tórax</i>	8	400	3,6 años
<i>TC de abdomen o pelvis</i>	10	500	4,5 años
<i>Medicina Nuclear</i>			
<i>Pulmonar de ventilación (Xe-133)</i>	0,3	15	7 semanas
<i>Pulmonar de perfusión (Tc-99m)</i>	1	50	6 meses
<i>Renal (Tc-99m)</i>	1	50	6 meses
<i>Tiroidea (Tc-99m)</i>	1	50	6 meses
<i>Ósea (Tc-99m)</i>	4	200	1,8 años
<i>Cardíaca dinámica (Tc-99m)</i>	6	300	2,7 años
<i>PET de la cabeza (F-18 FDG)</i>	5	250	2,3 años

Esta herramienta ofrece la manipulación de imágenes médicas, brinda la capacidad de evaluar imágenes de múltiples modalidades y realizar análisis comparativos.

La implementación de esta herramienta proporcionará un impulso general de productividad al Instituto Estatal de Cancerología y disminuirá sus costos operativos. También creará una excelente oportunidad para capacitar a los profesionales de la salud con la capacidad de trabajar de forma remota, proporcionando entornos de trabajo colaborativos y facilitando el acceso y el intercambio de información multimedia.

Capítulo 2. Marco teórico

En este capítulo se realiza una descripción de las técnicas, conceptos, arquitectura y teorías que se utilizan para el desarrollo de la herramienta computacional web.

2.1 Formato DICOM

En el Instituto Estatal de Cancerología se obtienen las imágenes de los dispositivos de adquisición de imagen en formato DICOM², este formato es un estándar internacional y se utiliza en todos los hospitales en el área de imagenología alrededor del mundo. DICOM se desarrolló de acuerdo con los procedimientos del Comité de estándares de DICOM. El estándar DICOM está estructurado como un documento de varias partes que utiliza directrices establecidas ([Directivas ISO/IEC, Parte 2] (PS3.1: DICOM PS3.1 2019b - Introduction and Overview, 2019)).

El estándar DICOM facilita la interoperabilidad de los equipos de imágenes médicas al especificar:

- Las comunicaciones de red, conjunto de protocolos que deben seguir los dispositivos que afirman cumplir con la Norma.
- La sintaxis y la semántica de los Comandos y la información asociada que se pueden intercambiar utilizando estos protocolos.
- La comunicación de medios, conjunto de servicios de almacenamiento que deben tener los dispositivos que afirman cumplir con el estándar, así como un formato de archivo y una estructura de directorio médico para facilitar el acceso a las imágenes y la información relacionada almacenada en los medios de intercambio.

² DICOM (Digital Imaging and Communications in Medicine) - Imagen digital y comunicaciones en medicina.

- La información que debe suministrarse con una implementación acorde con la norma.

Para realizar un diagnóstico por medio de imágenes, se utilizan imágenes en formato DICOM. Este formato es utilizado para transmitir, almacenar, recuperar, imprimir, procesar y mostrar información de las imágenes médicas. Utilizar este estándar tiene sus beneficios debido a que hace que la información de imágenes sea interoperable, integra los dispositivos de adquisición, PACS, estaciones de trabajo, los VNA e impresoras de diferentes fabricantes, aparte de que es desarrollado y mantenido para satisfacer las necesidades de las tecnologías y la evolución de la imagen médica activamente. En la Figura 2.1 se muestra la estructura de una carpeta en formato DICOM.

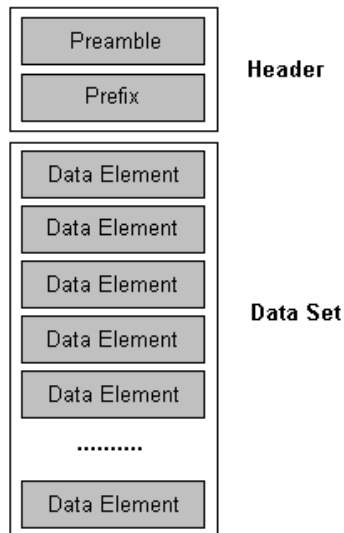


Figura 2.1. Estructura de una carpeta DICOM (Leadtools, 2017).

La parte del **Header** consta de un preámbulo de archivos de 128 bytes, seguido de un prefijo DICOM de 4 bytes, este Header puede o no estar incluido en el archivo.

- Preámbulo: 128 bytes = ??? ???
- Prefijo: 4 bytes = 'D', 'I', 'C', 'M'

El **Data Set** consta de varios Data Element, también llamado atributos DICOM donde cada uno de estos elementos de datos son únicos y están compuestos por etiquetas, representación del valor (VR), tamaño del valor y un campo del valor, como se muestra en la Figura 2.2.

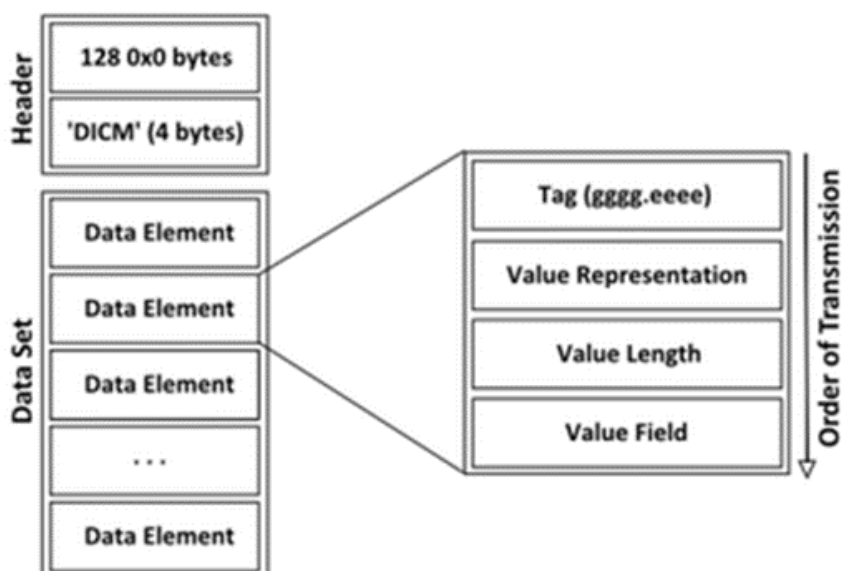


Figura 2.2. Estructura de un elemento de datos (Bojchko & Kroker, 2015).

Una imagen en formato DICOM contiene una gran variedad de **etiquetas** (*Tags*), estas son muy importantes, debido a que contiene información de la imagen. Una etiqueta identifica el atributo DICOM, generalmente en el formato (XXXX, XXXX) con números hexadecimales, y se puede dividir más adelante en Número de grupo DICOM y Número de elemento DICOM. En la tabla 2.1 se muestra algunos de los registros de elementos de datos DICOM más importantes.

Tabla 2.1. Registros de elementos de datos DICOM (Digital Imaging and Communications in Medicine (DICOM), 2017).

Attribute	Tag	VR	VL (hex)	Value
SOP Class UID	(0008,0016)	UI	001e	1.2.840.10008.5.1.4.1.1.88.33
SOP Instance UID	(0008,0018)	UI	0012	1.2.3.4.5.6.7.300
Study Date	(0008,0020)	DA	0008	19991029
Content Date	(0008,0023)	DA	0008	19991029
Study Time	(0008,0030)	TM	0006	154500
Content Time	(0008,0033)	TM	0006	154510
Accession Number	(0008,0050)	SH	0006	123456
Modality	(0008,0060)	CS	0002	SR
Manufacturer	(0008,0070)	LO	0004	WG6
Referring Physician's Name	(0008,0090)	PN	0014	Luke^Will^Dr.^M.D.
Referenced Study Component Sequence	(0008,1111)	SQ	fffffff	
Patient's Name	(0010,0010)	PN	000e	Homer^Jane^^^
Patient's ID	(0010,0020)	LO	0006	234567
Patient's Birth Date	(0010,0030)	DA	0008	19991109
Patient's Sex	(0010,0040)	CS	0002	F
Study Instance UID	(0020,000D)	UI	0012	1.2.3.4.5.6.7.100
Series Instance UID	(0020,000E)	UI	0012	1.2.3.4.5.6.7.200
Study ID	(0020,0010)	SH	0006	345678
Series Number	(0020,0011)	IS	0002	1
Instance (formerly Image) Number	(0020,0013)	IS	0002	1
Value Type	(0040,a040)	CS	000a	CONTAINER
Concept Name Code Sequence	(0040,a043)	SQ	fffffff	
Code Value	(0008,0100)	SH	0006	333300
Coding Scheme Designator	(0008,0102)	SH	0006	LNdemo
Code Meaning	(0008,0104)	LO	000c	Chest X-Ray
Continuity Of Content	(0040,a050)	CS	0008	SEPARATE
Verifying Observer Sequence	(0040,a073)	SQ	fffffff	
Verifying Organization	(0040,a027)	LO	0004	WG6

Debido a la gran cantidad de información que guarda una imagen DICOM, es utilizada por los médicos del sector de salud público y privado para ofrecer un diagnóstico por medio de imágenes.

Para que se pueda establecer una comunicación entre dos dispositivos se tiene que tener en cuenta los conceptos básicos de la comunicación DICOM. Dos entidades de aplicación (dispositivos) que desean comunicarse entre sí a través de una red mediante el protocolo DICOM primero deben establecer la comunicación a través de un *handshake* (saludo) inicial de la red. Uno de los dos dispositivos debe iniciar una asociación (conexión con el otro dispositivo), y preguntar si puede admitir servicios específicos, información y codificación.

DICOM especifica una serie de servicios de red y tipos de objetos de información, cada uno de los cuales se denomina sintaxis abstracta para la negociación. DICOM también especifica una variedad de métodos para codificar datos, denotados como sintaxis de transferencia. La negociación permite que la entidad de aplicación iniciadora proponga combinaciones de sintaxis abstracta y de transferencia que se utilizarán en la asociación; estas combinaciones se denominan contextos de presentación. La entidad de aplicación receptora acepta los contextos de presentación que admite.

Para cada contexto de presentación, la negociación de asociación también permite que los dispositivos acuerden los roles, uno de ellos es el usuario de clase de servicio (SCU, por sus siglas en inglés) y el proveedor de clase de servicio (SCP, por sus siglas en inglés). Normalmente, el dispositivo que inicia la conexión es la SCU, es decir, el sistema cliente llama al servidor, pero no siempre.

La negociación de asociación finalmente permite el intercambio de tamaño máximo de paquete de red (PDU), información de seguridad y opciones de servicio de red (llamada información de negociación extendida).

2.2 Procesamiento y análisis de imágenes digitales

Procesamiento de imágenes digitales significa procesar imágenes digitales por medio de una computadora digital. También se puede decir que es el uso de algoritmos informáticos, con el fin de obtener una imagen mejorada o para extraer información útil de una imagen.

El procesamiento de imágenes incluye principalmente los siguientes pasos:

1. Importar la imagen a través de herramientas de adquisición de imágenes;
2. Analizar y manipular la imagen;
3. Salida en la cual el resultado puede ser una imagen alterada o un informe que se basa en el análisis de esa imagen.

Una imagen se define como una función bidimensional, $F(x, y)$, donde x, y son coordenadas espaciales, y la amplitud de F en cualquier par de coordenadas (x, y) se denomina intensidad de esa imagen en ese punto. Cuando (x, y) y los valores de amplitud de F son finitos, se le llama imagen digital.

En otras palabras, una imagen puede definirse por una matriz bidimensional específicamente dispuesta en filas y columnas (Figura 2.3).

La imagen digital se compone de un número finito de elementos, cada uno de los cuales tiene un valor particular en una ubicación específica, a estos elementos también se les conocen como píxeles.

Tipos de una imagen

Imagen binaria: la imagen binaria como su nombre lo indica, contiene solo dos valores para los de píxeles, es decir, 0 y 1, donde 0 se refiere a negro y 1 se refiere a blanco. Esta imagen también se conoce como monocromo.

Imagen en blanco y negro: la imagen que consiste solo en color blanco y negro se llama imagen en escala de grises.

Formato de color de 8 bits: es el formato de imagen más conocido. Tiene 256 tonos diferentes de colores y se conoce comúnmente como imagen en escala de grises. En este formato, 0 significa negro, y 255 significa blanco y 127 significa gris.

Modelo RGB: Es un formato de imagen en color de 3 bytes. Tiene 16,7 millones colores diferentes (256x256x256). También es conocido como formato de color alto. En este formato, la distribución del color no es la misma que la imagen en escala de grises.

2.3 Conformación de una imagen

Para comprender mejor las propiedades inherentes de las imágenes y el procedimiento técnico utilizado para manipularlas y procesarlas, podemos pensar en una imagen, que se compone de píxeles individuales, como una función f . Cada píxel tiene su propio valor. Para una imagen en escala de grises, cada píxel tendría una intensidad entre 0 y 255, con 0 en negro y 255 en blanco. $f(x, y)$ luego daría la intensidad de la imagen en la posición del píxel (x, y) , asumiendo que está definida sobre un rectángulo, con un rango finito: $f: [a, b] \times [c, d] \rightarrow [0, 255]$.

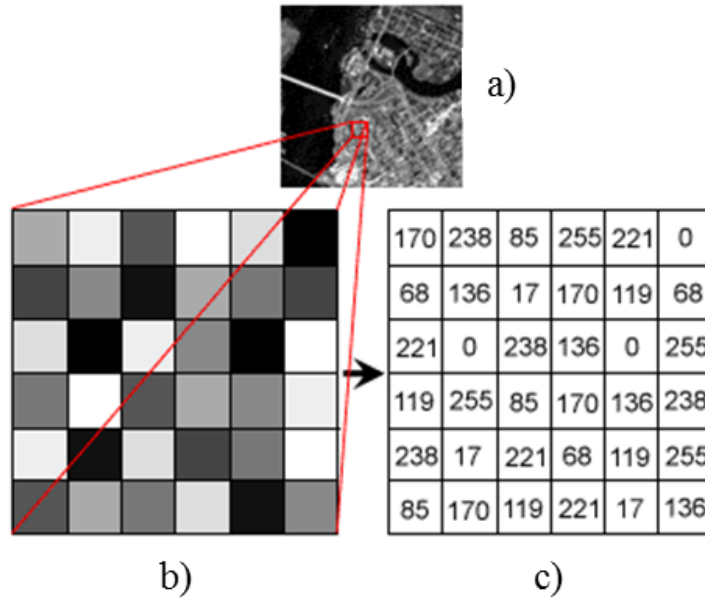


Figura 2.3. a) Imagen original. b) Fracción pequeña de la imagen original en píxeles. c) valores de la intensidad de píxeles.

Una imagen, puede representarse como una matriz de valores de píxeles, como se muestra en la Figura 2.4 y 2.5.

$$f(x,y) = \begin{bmatrix} f(0,0) & f(0,1) & f(0,2) & \dots & f(0,N-1) \\ f(1,0) & f(1,1) & f(1,2) & \dots & f(1,N-1) \\ \cdot & \cdot & \cdot & & \cdot \\ \cdot & \cdot & \cdot & & \cdot \\ \cdot & \cdot & \cdot & & \cdot \\ f(M-1,0) & f(M-1,1) & f(M-1,2) & \dots & f(M-1,N-1) \end{bmatrix}$$

Figura 2.4. Representación de imagen en matriz.

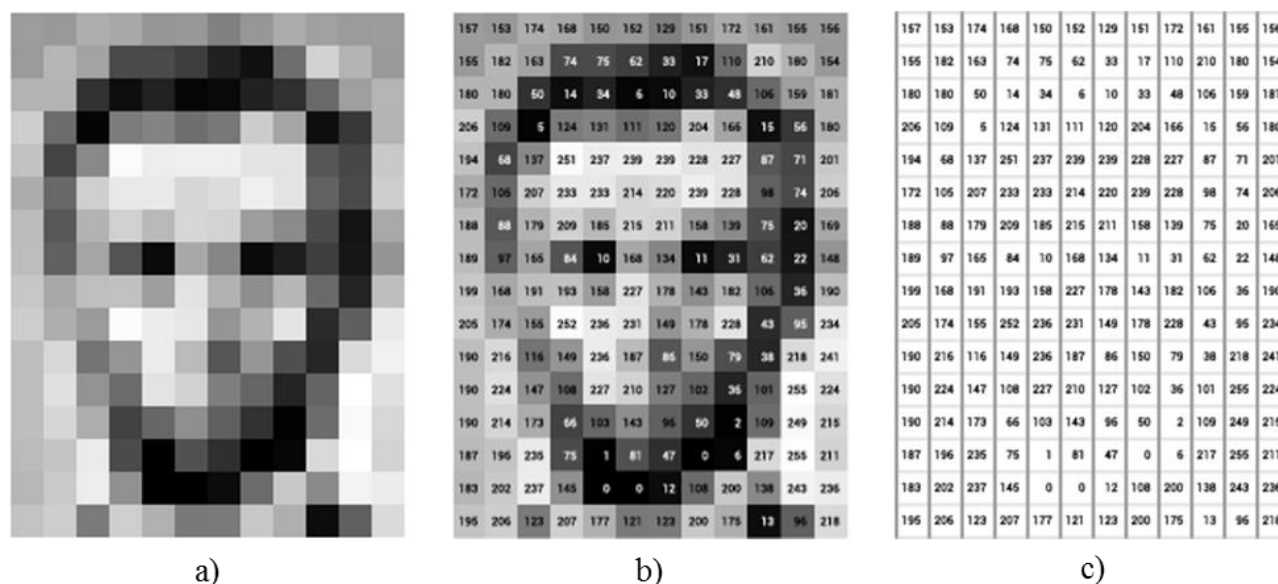


Figura 2.5. Representación de imagen en píxeles. a) Imagen original pixelada. b) Imagen pixelada con valores de los píxeles. c) Valores de intensidad de los píxeles (Yeung & Alahi, 2015).

2.4 Filtrado

Los filtros se utilizan para eliminar ruido en las imágenes, este ruido se origina por varias causas: dispositivos de captura, condiciones ambientales, proceso de digitalización, almacenamiento y transmisión.

Los filtros se dividen principalmente en 2 tipos:

- Los filtros de paso bajo – Eliminando las altas frecuencias, suavizando las imágenes eliminando ruido, pero emborronando los bordes/orillas.
- Los filtros de paso alto – Eliminando las bajas frecuencias, resaltando bordes/orillas en la imagen.

Una rutina de software que cambia el aspecto de una imagen o parte de una imagen al alterar los tonos y colores de los píxeles de alguna manera. Los filtros se utilizan para aumentar el brillo

y el contraste, así como para agregar una amplia variedad de texturas, tonos y efectos especiales a la imagen.

Los filtros se pueden clasificar en dos grandes grupos:

Filtros lineales: si satisface los principios de linealidad y superposición. Es decir, el filtro lineal se caracteriza por el hecho de que el resultado de cada pixel se obtiene como combinación lineal de sus vecinos. Para aplicar un filtro lineal multiplicamos el entorno de cada pixel (submatriz $N \times N$) por una máscara (que contiene los pesos de cada pixel del entorno con que se pondera la media), la media ponderada será el nuevo valor del pixel cuyo entorno operábamos. Esto es lo mismo que convolucionar la imagen original por una máscara (submatriz $N \times N$) que contiene los pesos con que cada vecino del entorno contribuye al resultado final o valor del pixel en la imagen filtrada.

Filtros no-lineales: si no satisface los principios de linealidad y superposición. Su importancia deriva de los siguientes hechos. Por una parte, los filtros lineales muestran un bajo rendimiento cuando el ruido no es aditivo, o cuando este no se distribuyen según la función gaussiana. Por otra parte, al parecer, el procesamiento temprano no satisface la propiedad de la linealidad. Como consecuencia de esto último, el resultado de un filtro no-lineal no se puede obtener mediante la convolución de la señal (imagen original) con la función de respuesta al impulso unidad. Ello da lugar a mayor complejidad del cálculo de los diferentes filtros no-lineales que, hasta la fecha, se han propuesto. Entre los que destacaremos por su mayor simplicidad el filtro de mediana (PSICOLOGÍA DE LA PERCEPCIÓN VISUAL, 2018).

Filtro de paso alto

El borde es un límite entre dos regiones con un nivel de gris relativamente distintas propiedades.

- Los bordes son píxeles en los que la función de brillo cambia bruscamente.

- Los detectores de bordes son una colección de métodos de preprocesamiento de imágenes locales muy importantes que se utilizan para localizar cambios (nítidos) en la función de intensidad.

En la visión por computadora, los bordes son discontinuidades repentinas en una imagen, que pueden surgir de la superficie normal, el color de la superficie, la profundidad, la iluminación u otras discontinuidades. Los bordes son importantes por dos razones principales (Yeung & Alahi, 2015).

- 1) La mayor parte de la información semántica y de formas se puede deducir de ellos, por lo que podemos realizar el reconocimiento de objetos y analizar las perspectivas y la geometría de una imagen.
- 2) Son una representación más compacta que los píxeles. Se puede señalar dónde se producen los bordes desde el perfil de intensidad de una imagen a lo largo de una fila o columna de la imagen. Dondequiera que haya un cambio rápido en la función de intensidad indica un borde.

En la Figura 2.6 se muestra el funcionamiento del filtro de paso alto, el cual se realizó detección de bordes mediante el algoritmo de Canny.

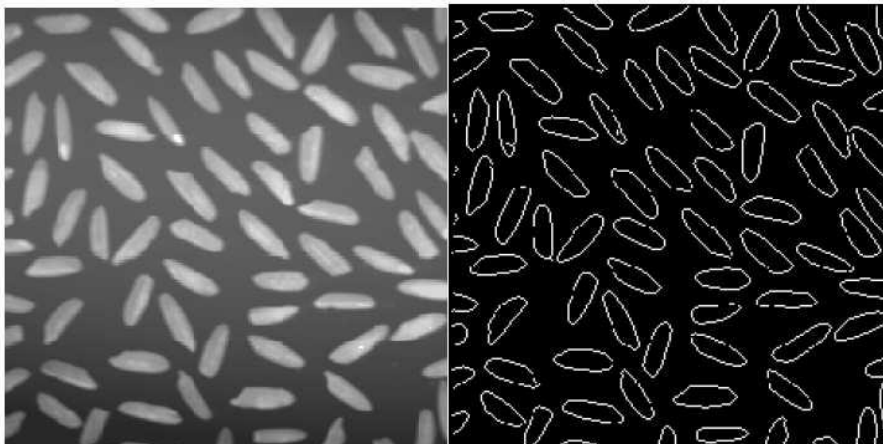


Figura 2.6. Imagen filtrada con algoritmo de Canny.

Filtro de paso bajo

Los filtros de desenfoque suavizan una selección o una imagen, son útiles para retocar y elimina el ruido cuando se producen transiciones de color significativas en una imagen. Suavizan las transiciones promediando los valores de color de los píxeles junto a los bordes duros de las líneas definidas y las áreas sombreadas.

Si se observa con cuidado una imagen borrosa, entonces es normal observar que la imagen es suave, lo que significa que no se observan bordes. Un filtro usado para desenfoque también se llama filtro de paso bajo, porque permite que la baja frecuencia pase y se detenga en la alta frecuencia. Aquí la frecuencia significa el cambio de valor de píxel, alrededor del borde, el valor del píxel cambia rápidamente a medida que la imagen desenfocada es suave, por lo que la alta frecuencia debe ser filtrada.

Para fines de desenfoque se usa un filtro con cada llamada que tiene valor 1 porque para desenfocar una imagen, un valor de píxel debe estar cerca del valor del vecino. En la Figura 2.7 se puede observar el funcionamiento del filtro suavizado.

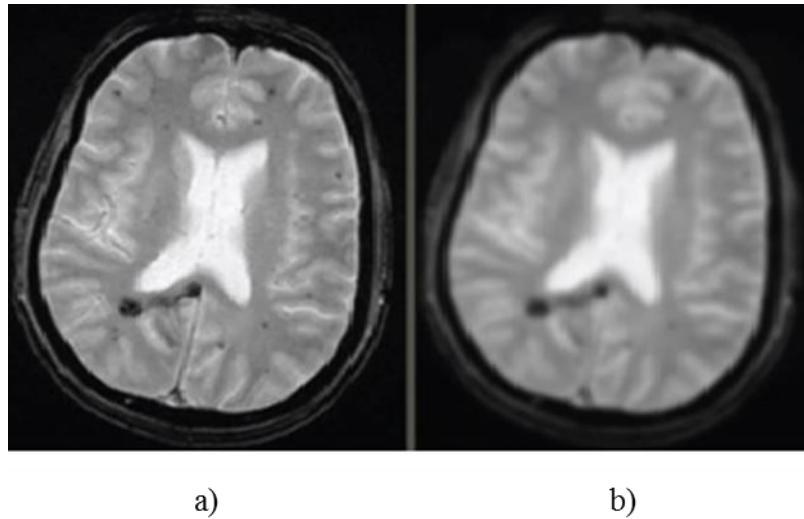


Figura 2.7. Filtro de paso bajo, a) Imagen original con ruido. b) Imagen filtrada con Gaussian filter.

Convolución

La imagen se almacena como matriz dentro de la computadora. Una imagen contiene muchas características como borde, contraste, etc. En el procesamiento de imágenes, las características deben extraerse de la imagen para un estudio adicional de la misma.

La convolución es una operación fundamental en imágenes en las que se aplica una operación matemática a cada píxel para obtener el resultado deseado (mejoramiento de la imagen).

Para este propósito, se usa otra matriz llamada kernel que es más pequeña en tamaño de imagen. Esto también se llama filtro. Este filtro se aplica a cada píxel de la imagen y el nuevo valor obtenido es el valor de ese píxel. La imagen obtenida se llama imagen filtrada. En el **kernel**, cada celda contiene algún valor, ese kernel se mantiene por encima del píxel y los valores correspondientes se multiplican y posteriormente se resume este valor obtenido es nuevo el valor del píxel. La operación que se realiza con el kernel depende del tipo de filtro que se aplica, no en todos se multiplica, en el filtro de la media se suma y posteriormente se divide, en el filtro de la mediana se suma y al final se elige el valor medio.

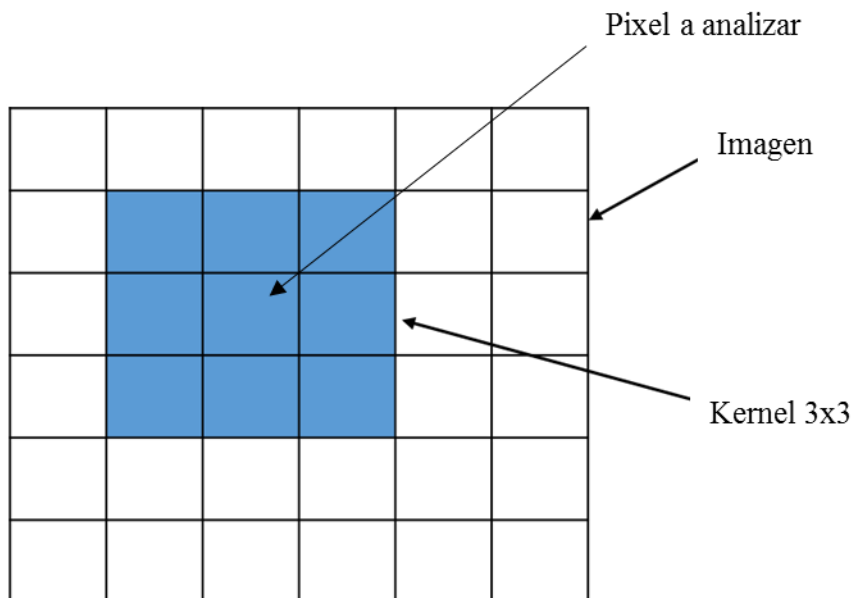


Figura 2.8. Pixel que se procesa en aplicación de filtro.

2.5 Arquitectura Cliente-Servidor

La arquitectura Cliente/Servidor es un modelo informático en el que el servidor hospeda, entrega y administra la mayoría de los recursos y servicios que consumirá el cliente. Este tipo de arquitectura tiene uno o más equipos cliente conectados a un servidor central a través de una red o conexión a Internet. Este sistema comparte recursos informáticos.

La arquitectura Cliente/Servidor también se conoce como un modelo informático de red o red Cliente/Servidor porque todas las solicitudes y servicios se entregan a través de una red.

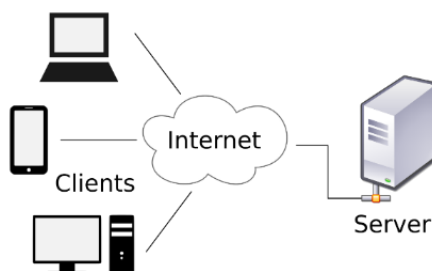


Figura 2.9. Modelo Cliente-Servidor.

La arquitectura cliente / servidor es una arquitectura informática de productor / consumidor en la que el servidor actúa como productor y el cliente como consumidor. El servidor aloja y proporciona servicios de alto nivel e intensivos en computación al cliente a pedido. Estos servicios pueden incluir acceso a la aplicación, almacenamiento, uso compartido de archivos, acceso a la impresora y / o acceso directo a la potencia de cálculo sin formato del servidor.

La arquitectura Cliente/Servidor funciona cuando la computadora cliente envía una solicitud de recursos o procesos al servidor a través de la conexión de red, que luego se procesa y se entrega al cliente. Una computadora servidor puede administrar varios clientes simultáneamente, mientras que un cliente puede conectarse a varios servidores a la vez, cada uno de los cuales proporciona un conjunto diferente de servicios. En su forma más simple, Internet también se basa en la arquitectura Cliente/Servidor donde los servidores web sirven a muchos usuarios simultáneos con datos de sitios web.

2.6 ASP.NET Core MVC

Para el desarrollo de esta herramienta web se está utilizando el marco de trabajo ASP.NET Core esto se debe a que es multiplataforma de alto rendimiento, además es de código abierto. ASP.NET Core es la última evolución del popular marco web ASP.NET de Microsoft, lanzado en junio de 2016 (Roth, Anderson, & Luttin, 2019).

ASP.NET Core se utiliza para desarrollar aplicaciones modernas, basadas en la nube y conectadas a internet. Con ASP.NET Core, puede:

- Crear aplicaciones y servicios web, aplicaciones de IoT y backends móviles.
- Utilizar sus herramientas de desarrollo favoritas en Windows, macOS y Linux.
- Desplegar las aplicaciones en la nube o en las instalaciones.

- Ejecutar en .NET Core o .NET Framework.

ASP.NET Core ofrece los siguientes beneficios:

- Una historia unificada para construir UI web y APIs web.
- Arquitectura para hacer pruebas.
- *Razor Pages* hace que la codificación de escenarios enfocados en páginas sea más fácil y más productiva.
- Capacidad para desarrollar y ejecutar en Windows, macOS y Linux.
- De código abierto y enfocado a la comunidad.
- Integración de marcos de trabajo modernos, del lado del cliente y flujos de trabajo de desarrollo.
- Un sistema de configuración preparado para la nube y basado en el entorno.
- Inyección de dependencia incorporada.
- Un canal de solicitudes HTTP ligero, de alto rendimiento y modular.
- Capacidad para alojar en IIS, Nginx, Apache, Docker o auto-host en su propio proceso.
- Aplicación de versiones lado a lado cuando se dirige a .NET Core.
- Herramientas que simplifican el desarrollo web moderno.

ASP.NET Core MVC es un marco enriquecido para crear aplicaciones web y API utilizando el patrón de diseño Modelo-Vista-Controlador. El patrón arquitectónico Modelo-Vista-Controlador (MVC) separa una aplicación en tres grupos principales de componentes: Modelos, Vistas y Controladores. Este patrón ayuda a lograr la separación de los procesos. Usando este patrón, las solicitudes de los usuarios se enrutan a un controlador que es responsable de trabajar con el Modelo para realizar acciones de los usuarios y/o recuperar los resultados de las consultas.

El Controlador elige la Vista para mostrar al usuario y le proporciona los datos del Modelo que necesita (Smith, 2018).

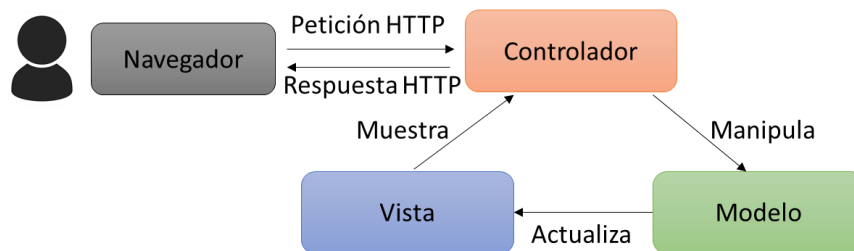


Figura 2.10. Diagrama MVC.

El **modelo** en una aplicación MVC representa el estado de la aplicación y cualquier lógica u operación comercial que deba realizar. La lógica de negocios debe estar encapsulada en el modelo, junto con cualquier lógica de implementación para mantener el estado de la aplicación. Las vistas de tipo fuerte suelen utilizar tipos de *ViewModel* diseñados para contener los datos que se mostrarán en esa vista. El controlador crea y llena estas instancias de *ViewModel* a partir del modelo.

Las **vistas** son responsables de presentar el contenido a través de la interfaz de usuario. Utilizan el motor de vista *Razor* para incrustar código .NET en el marcado HTML. Debe haber una lógica mínima en las vistas, y cualquier lógica debe estar relacionada con la presentación del contenido.

Los **controladores** son los componentes que manejan la interacción del usuario, trabajan con el modelo y finalmente seleccionan una vista para renderizar. En una aplicación MVC, la vista solo muestra información; El controlador maneja y responde a la entrada e interacción del usuario. En el patrón MVC, el controlador es el punto de entrada inicial y es responsable de seleccionar con qué tipos de modelo trabajar y con qué vista (por lo tanto, su nombre controla la forma en que la aplicación responde a una solicitud determinada).

2.6.1 HTML

Para darle una estructura a todo el contenido que se muestra en esta herramienta web se utiliza la versión más actual, HTML5 (Lenguaje de Marcado de Hipertexto). HTML es el bloque de construcción más básico de la Web. Define el significado y la estructura del contenido web (HTML: Hypertext Markup Language, 2019).

Hipertexto se refiere a enlaces que conectan páginas web entre sí, ya sea dentro de un solo sitio web o entre sitios web. Los enlaces son un aspecto fundamental de la web.

HTML usa **Marcado** para anotar texto, imágenes y otro contenido para mostrar en un navegador web. El marcado HTML incluye elementos especiales como: <head>, <title>, <body>, <header>, <footer>, <article>, <section>, <p>, <div>, , , <aside>, <audio>, <canvas>, <datalist>, <details>, <embed>, <nav>, <output>, <progress>, <video> y muchos otros.

Un **elemento** HTML está separado de otro texto en un documento mediante etiquetas, que consisten en el nombre del elemento rodeado por "<" y ">". El nombre de un elemento dentro de una etiqueta no distingue entre mayúsculas y minúsculas. Es decir, se puede escribir en mayúsculas, minúsculas o una mezcla. Por ejemplo, la etiqueta <title> puede escribirse como <Title>, <TITLE>, o de cualquier otra forma.

2.6.2 CSS

Para la parte del estilo y diseño de la herramienta web se utiliza la versión actual de CSS3 (Hojas de estilo en cascada).

CSS es el lenguaje para describir la presentación de las páginas web, incluidos los colores, el diseño y las fuentes. Permite adaptar la presentación a diferentes tipos de dispositivos, como

pantallas grandes, pantallas pequeñas o impresoras. CSS es independiente de HTML y se puede utilizar con cualquier lenguaje de marcado basado en XML. La separación de HTML de CSS hace que sea más fácil mantener los sitios, compartir hojas de estilo en las páginas y adaptar las páginas a diferentes entornos. Esto se conoce como la separación de estructura o contenido de la presentación (Html y Css, 2019).

2.6.3 JavaScript

JavaScript es un lenguaje de programación que surgió con el objetivo inicial de programar ciertos comportamientos sobre las páginas web, respondiendo a la interacción del usuario y la realización de automatismos sencillos. En ese contexto podríamos decir que nació como un "lenguaje de scripting" del lado del cliente. Las necesidades de las aplicaciones web modernas y el HTML5 ha provocado que el uso de JavaScript que encontramos hoy haya llegado a unos niveles de complejidad y prestaciones tan grandes como otros lenguajes de primer nivel (Javascript, 2015).

JavaScript puede funcionar como lenguaje procedimental y como lenguaje orientado a objetos. Los objetos se crean codificándolos, añadiendo métodos y propiedades a lo que de otra forma serían objetos vacíos en tiempo de ejecución, en contraposición a las definiciones sintácticas de clases comunes en los lenguajes compilados como C++ y Java. Una vez que se haya construido un objeto, puede usarse como modelo o prototipo para crear objetos similares.

Para entender bien lo que es JavaScript y en qué situaciones se utiliza se debe conocer los distintos lenguajes que se utilizan en la web. No es necesario que se domine todos los lenguajes de la web para hacer un buen uso de JavaScript, tan solo será necesario que se conozca el HTML y se tenga algunas nociones de CSS.

2.6.4 C#

C# o C Sharp es un lenguaje de programación que está incluido en la Plataforma .NET y corre en el Lenguaje Común en Tiempo de Ejecución (CLR, *Common Language Runtime*). El primer lenguaje en importancia para el CLR es C#, mucho de lo que soporta la Plataforma .NET está escrito en C#. La sintaxis de C# se deriva de C y C++, es moderno, simple y enteramente orientado a objetos, simplifica y moderniza a C++ en las áreas de clases, *namespaces*, sobrecarga de métodos y manejo de excepciones. Se eliminó la complejidad de C++ para hacerlo más fácil de utilizar y menos propenso a errores.

C# es un lenguaje orientado a objetos, pero también incluye compatibilidad para programación orientada a componentes. El diseño de software contemporáneo se basa cada vez más en componentes de software en forma de paquetes independientes y autodescriptivos de funcionalidad. La clave de estos componentes es que presentan un modelo de programación con propiedades, métodos y eventos; tienen atributos que proporcionan información declarativa sobre el componente; e incorporan su propia documentación. C# proporciona construcciones de lenguaje para admitir directamente estos conceptos, por lo que se trata de un lenguaje muy natural en el que crear y usar componentes de software (Un paseo por el lenguaje C#, 2019).

C# tiene un sistema de tipo unificado. Todos los tipos de C#, incluidos los tipos primitivos como `int` y `double`, se heredan de un único tipo *object* raíz. Por lo tanto, todos los tipos comparten un conjunto de operaciones comunes, y los valores de todos los tipos se pueden almacenar, transportar y utilizar de manera coherente. Además, C# admite tipos de valor y tipos de referencia definidos por el usuario, lo que permite la asignación dinámica de objetos, así como almacenamiento en línea de estructuras ligeras.

2.7 Metodología de desarrollo de Software

2.7.1 Kanban

Kanban comenzó a principios de la década de 1940. El primer sistema Kanban fue desarrollado por Taiichi Ohno (ingeniero industrial y empresario) para Toyota Automotive en Japón. Fue creado como un sistema de planificación simple, cuyo objetivo era controlar y administrar el trabajo y el inventario en todas las etapas de la producción de manera óptima.

Una razón clave para el desarrollo de Kanban fue la inadecuada productividad y eficiencia de Toyota en comparación con sus rivales automotrices estadounidenses. Con Kanban, Toyota logró un sistema de control de producción justo a tiempo flexible y eficiente que aumentó la productividad al tiempo que redujo el inventario costoso de materias primas, materiales semiacabados y productos terminados.

Si bien Kanban fue introducido por Taiichi Ohno en la industria manufacturera, fue David J. Anderson quien fue el primero en aplicar el concepto al trabajo de TI, desarrollo de software y conocimiento en general en el año 2004. David se basó en los trabajos de Taiichi Ohno, Eli Goldratt, Edward Demmings, Peter Drucker y otros para definir el Método Kanban, con conceptos como sistemas de extracción, teoría de colas y flujo.

Los cuatro principios fundamentales y las seis prácticas fundamentales de la metodología Kanban se proporcionan a continuación:

- *Comenzar con lo que está haciendo ahora:* El Método Kanban enfatiza fuertemente no hacer ningún cambio en su configuración/proceso existente de inmediato. Kanban debe aplicarse directamente al flujo de trabajo actual. Cualquier cambio necesario puede ocurrir gradualmente durante un período de tiempo a un ritmo con el que el equipo se sienta cómodo.

- *Acordar buscar un cambio evolutivo incremental:* Kanban lo alienta a realizar pequeños cambios incrementales en lugar de hacer cambios radicales que puedan generar resistencia dentro del equipo y la organización.
- *Inicialmente, respetar los roles, responsabilidades y títulos de trabajo actuales:* A diferencia de otros métodos, Kanban no impone ningún cambio organizacional por sí mismo. Por lo tanto, no es necesario realizar cambios en sus roles y funciones existentes que pueden estar funcionando bien. El equipo identificará e implementará en colaboración todos los cambios necesarios. Estos tres principios ayudan a las organizaciones a superar la resistencia emocional típica y el miedo al cambio que generalmente acompañan a cualquier iniciativa de cambio en una organización.
- *Fomentar los actos de liderazgo en todos los niveles:* Kanban fomenta la mejora continua en todos los niveles de la organización y dice que los actos de liderazgo no tienen que originarse únicamente en los altos directivos. Las personas en todos los niveles pueden proporcionar ideas y mostrar liderazgo para implementar cambios para mejorar continuamente la forma en que entregan sus productos y servicios.

Las 6 prácticas fundamentales del método Kanban se muestran a continuación:

- *Visualizar el flujo de trabajo:* Este es el primer paso fundamental para adoptar e implementar el Método Kanban. Debe visualizar, ya sea en una placa física o en una placa Kanban electrónica, los pasos del proceso que utiliza actualmente para entregar el trabajo o los servicios. Dependiendo de la complejidad del proceso y su mezcla de trabajo (los diferentes tipos de elementos de trabajo en los que trabaja y entrega), el tablero Kanban puede ser muy simple o muy elaborado. En la Figura 2.11 se muestra el tablero Kanban con 3 columnas, REQUESTED (por hacer), IN PROGRESS (en proceso) y DONE (hecho).

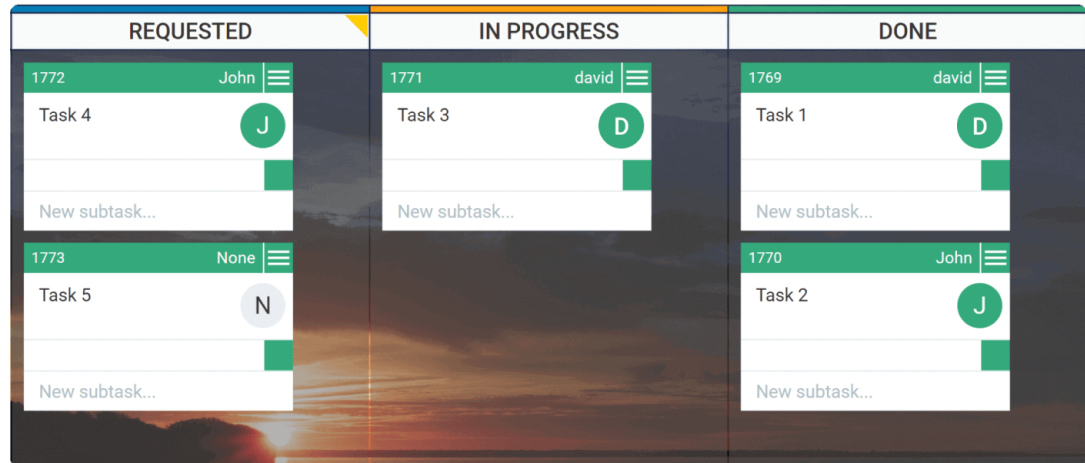


Figura 2.11. Tablero Kanban.

- *Limitar WIP (trabajo en progreso):* Al limitar WIP, Se alienta al equipo a completar el trabajo en progreso primero antes de comenzar un nuevo trabajo. Por lo tanto, el trabajo actualmente en progreso debe completarse y marcarse como realizado. Esto crea capacidad en el sistema, por lo que el equipo puede realizar un nuevo trabajo. Inicialmente, puede no ser fácil decidir cuáles deberían ser los límites de WIP. De hecho, puede comenzar sin límites de WIP.
- *Administrar flujo:* Administrar y mejorar el flujo es el punto más importante del sistema Kanban después de haber implementado las 2 primeras prácticas. Un sistema Kanban ayuda a administrar el flujo al resaltar las diversas etapas del flujo de trabajo y el estado del trabajo en cada etapa. Kanban ayuda al equipo a analizar el sistema y hacer ajustes para mejorar el flujo a fin de reducir el tiempo que lleva completar cada trabajo.
- *Hacer explícitas las políticas de proceso:* Como parte de la visualización del proceso, se debe definir y visualizar explícitamente sus políticas (reglas o pautas de proceso) sobre cómo realizar el trabajo que se está haciendo. Al formular pautas de proceso explícitas,

crea una base común para que todos los participantes comprendan cómo hacer cualquier tipo de trabajo en el sistema.

- *Implementar bucles de retroalimentación:* los bucles de retroalimentación son parte integral de cualquier buen sistema. El Método Kanban alienta y ayuda a implementar bucles de retroalimentación de varios tipos: revisar las etapas en el flujo de trabajo, las métricas y los informes de su tablero Kanban y una variedad de señales visuales que le brindan retroalimentación continua sobre el progreso del trabajo, o la falta de ella, en el sistema.
- *Mejorar en colaboración, evolución experimentalmente:* El método Kanban es un proceso de mejora evolutiva. Ayuda a adoptar pequeños cambios y a mejorar gradualmente a un ritmo y tamaño que su equipo puede manejar fácilmente.

Kanban es un sistema visual para gestionar el trabajo a medida que avanza en un proceso. Kanban visualiza tanto el proceso (el flujo de trabajo) como el trabajo real que pasa por ese proceso. El objetivo de Kanban es identificar posibles cuellos de botella en su proceso y corregirlos para que el trabajo pueda fluir de manera rentable a una velocidad o rendimiento óptimos.

Capítulo 3. Estado del arte

El diagnóstico por imágenes médicas surgió en el año 1895, cuando el físico alemán Wilhelm Conrad Roentgen descubrió los rayos X mientras realizaba experimentos con tubos de vacío y un generador eléctrico. Con el hallazgo de este haz de luz capaz de atravesar la materia, nació el diagnóstico médico por imágenes que mejoró considerablemente el ejercicio de la medicina.



Figura 3.1. Primera radiografía de rayos X.

Las imágenes médicas son el conjunto de técnicas que producen imágenes del interior del cuerpo humano de forma no invasiva con el fin aportar información sobre su estructura y funcionamiento, y ayudar así a detectar posibles anomalías en el mismo (Narváez et al., 2009). Una variedad de aparatos y técnicas pueden crear imágenes de las estructuras y actividades dentro del cuerpo. El tipo de imagen que usará el médico dependerá de los síntomas y de la parte del cuerpo humano que debe examinarse.

Las técnicas de imagen médica están basadas en el desarrollo de sistemas capaces de detectar diferentes señales físicas emitidas por el cuerpo u órgano objeto de estudio, y que son imperceptibles para el ser humano.

Existen diversas herramientas que apoyan a los radiólogos a visualizar y manipular las imágenes médicas en formato DICOM.

3.1 Plataformas existentes para la manipulación de imágenes

El trabajo realizado por Chua (Chua et al., 2015), proponen una herramienta basada en RIS³ y PACS⁴, mencionan que al utilizar estos dos sistemas en conjunción ofrecen establecer una red segura para la transmisión de información de los pacientes. A estos sistemas a menudo se les manejan como sinónimos. Con innumerables formas de llenar y almacenar archivos voluminosos llenos de información del paciente en los exámenes instrumentales, así las imágenes resultantes y documentos puedan ser procesados de manera *offline*.

En el trabajo propuesto por Chua se pretende desarrollar un sistema web, donde diferentes personas puedan acceder a la información médica y a las herramientas disponibles en una plataforma integrada, desde cualquier ubicación geográfica y en cualquier momento.

Chua menciona que los desafíos para los investigadores de visualización consisten principalmente en apoyar con el estándar DICOM y precisión en la evaluación de las imágenes utilizando un conjunto combinado de filtros presentes en una ubicación. En una estación de trabajo independiente, los programas sofisticados de procesamiento y visualización de imágenes están ampliamente disponibles para este tipo de tareas de computación. En respuesta, proponen un sistema de visualización y recuperación de información basado en la web. El principal desafío técnico es hacer que la funcionalidad de los programas dedicados de imágenes y visualización esté

³ RIS (Radiology Information System) - Sistema de Información Radiológica.

⁴ PACS (Picture Archiving and Communication System) - Sistema de Comunicación y Almacenamiento de Imágenes.

disponible públicamente a través de una interfaz basada en la web para facilitar la gestión del flujo de trabajo de las prácticas de radiología.

El sistema de servicios web que proponen los autores de este artículo consta de una serie de capas, la primera es una capa de presentación que está en el lado del cliente, las herramientas actuales basadas en la web requieren una descarga parcial o completa de los algoritmos en la máquina cliente. La ejecución del algoritmo de visualización depende de la capacidad de cálculo y la configuración de la máquina. El cliente puede relegar la tarea de procesar los conjuntos de datos al servidor y recuperar solo las salidas a través de la red, esto reduce la carga en el ancho de banda de la red. Mientras que la segunda capa, la lógica de negocios y la administración de datos se asignan al servidor. La base de datos es creada en SQL Server, las imágenes son almacenadas en formato BLOB, para que la transmisión sea mucho más rápida y el servidor SQL mantenga la consistencia transaccional, la tercera capa le llaman capa de algoritmo, porque contiene el núcleo de la herramienta de visualización.

Los autores realizaron pruebas en diferentes navegadores web, emuladores de dispositivos móviles y dispositivos móviles reales para observar si funcionan correctamente en las diversas plataformas y entornos del sistema operativo que se pueden implementar. Esencialmente, la aplicación se probó en Google Chrome e Internet Explorer para Windows 7, Safari para iOS y Opera para Android; se instalaron complementos antes de usar los emuladores móviles para probar la aplicación. Electric Mobile Studio se usó para probar en la plataforma iOS, y Opera Mobile se usó para la plataforma Android. Al alojar la aplicación web en el servidor de desarrollo ASP.NET y luego usar un programa de terceros llamado SharpProxy que enlaza el puerto interno que el servidor usa para alojar la aplicación en un puerto externo. Para acceder a la aplicación, el dispositivo móvil se conectó en la misma red que el servidor de alojamiento y la dirección IP y el

puerto externo. Los comportamientos del sistema también se monitorearon en la etapa de pruebas manuales en los navegadores web, ya que se realizó una serie de listas de verificación para garantizar que se cumplan los requisitos funcionales para cada unidad de código.

A su vez el trabajo realizado por Marques y otros (Marques et al., 2016) menciona que a lo largo de los años la complejidad del entorno de imágenes médicas ha favorecido la aparición de una multitud de sistemas de información destinados a soportar partes específicas del flujo de trabajo médico. Más concretamente, hay dos clases principales de sistemas de información en los laboratorios de imágenes médicas: el Sistema de Comunicación y Almacenamiento de Imágenes (PACS) y el Sistema de información radiológica (RIS). PACS es la designación común dada a los sistemas que manejan directamente con datos de imágenes médicas y gestionan flujos de trabajo entre dispositivos de adquisición, repositorios y estaciones de trabajo de visor. Están diseñados para permitir el almacenamiento e intercambio de imágenes médicas y metadatos respectivos adquiridos por el equipo, permitiendo que estos datos se muestren en las estaciones de trabajo de los usuarios finales. Además, están diseñados para abordar la necesidad de grandes requisitos de almacenamiento y transmisión de información dentro de las instituciones médicas. Además de la radiología, que fue la primera área en usar estos sistemas, hay otras áreas clínicas que también han adoptado PACS en sus rutinas diarias, especialmente cardiología.

En la investigación Marques y otros proponen un método para desbloquear el potencial de los datos que poseen los sistemas de imágenes médicas, es decir, a través de la integración de distintas fuentes: imágenes, metadatos e informes estructurados. La propuesta fue instanciada en el contexto de la modalidad de ecocardiografía (ultrasonido - US). A pesar de estar cubiertos por el estándar DICOM, los informes médicos aún se almacenan predominantemente en el RIS/HIS, siguiendo distintos modelos de datos y vocabularios terminológicos. El sistema propuesto permite

transcodificar informes a DICOM SR (Reportes estructurados) y su consecuente integración con los elementos restantes (imágenes y metadatos). Este método fue desarrollado como un módulo para el PACS de código abierto de *Dicoogle*⁵ y sigue la arquitectura de software presentada en *Dicoogle*. Permite indexar distintas fuentes de datos, realizar búsquedas multimodales y explorar datos de manera integrada a través de sus herramientas de análisis de datos.

En la arquitectura y metodología del sistema propuesto incluye tres componentes principales. En primer lugar, un archivo PACS capaz de almacenar e indexar distintos tipos de datos de imágenes médicas. En este caso, se utilizó la plataforma *Dicoogle* debido a las ventajas que este marco de trabajo ofrece. El segundo componente es el *Echoogle*, debido a que es capaz de recopilar datos de informes intermedios en varios formatos, ya que admite múltiples sensores diseñados para cada fuente de datos. Por último, el componente más importante en el método propuesto es el generador DICOM US SR. Este generador fue desarrollado como un complemento para *Dicoogle*. Tomaron este enfoque de diseño ya que facilita la integración con el archivo PACS multimodal. Además, el marco *Dicoogle* proporciona muchas de las funcionalidades necesarias para admitir el estándar DICOM, por ejemplo, ensamblar el archivo DICOM SR.

Dicoogle admite distintos tipos de complementos, incluidos complementos de índice. Cada vez que un nuevo objeto DICOM llega a *Dicoogle* a través de una de sus interfaces, como el Servicio de almacenamiento DICOM o los Servicios web, se llama a los complementos de índice activos para indexar el objeto. Se puede considerar que los complementos de índice son como oyentes para los objetos entrantes. El complemento utilizado usa este mecanismo para activar la generación de informes. Para cada uno de los objetos entrantes, verifica si el objeto es una imagen de ultrasonido y extrae su número de acceso, un identificador único para el estudio de imágenes

⁵ *Dicoogle* - Es un archivo de código abierto del Sistema de archivo de imágenes y comunicaciones (PACS).

médicas. Es un campo obligatorio en el estándar DICOM y generalmente lo genera el RIS/HIS a través del servicio de lista de trabajo de modalidad. Durante este proceso, el complemento puede crear una base de datos de imágenes, que puede faltar en su informe médico.

En la segunda etapa, el sistema propuesto genera un informe para cada entrada de la base de datos. Para lograr este objetivo, periódicamente consultará a *Echoogle* por documentos con el mismo número de acceso que la entrada seleccionada. Esto se ejecuta a través de la interfaz de servicio web de Echoogle, que devuelve un objeto JSON con los múltiples campos extraídos del informe médico, en caso de que exista. En este punto, se puede generar el DICOM SR basado en las plantillas estándar.

Por último, en la tercera etapa, el informe generado se almacena de nuevo en el repositorio de Dicoogle. Dado que este método se desarrolló como un complemento, no es necesario recurrir a ningún protocolo de comunicación que no sea la API interna de *Dicoogle*, que tiene un rendimiento mejorado.

El trabajo realizado por Savaris y otros (Savaris et al., 2017) desarrollaron un sistema originalmente concebido para especialidades en radiología, los Sistema de Comunicación y Almacenamiento de Imágenes (PACS), con el paso del tiempo se ha ido relacionando con algunas modalidades de examen como la teledermatología y la cardiología. Esta relación se puede explorar para una eventual integración entre PACS y los sistemas de telemedicina, que pueden beneficiarse de la adquisición, el intercambio y el almacenamiento de imágenes digitales. Para el Sistema Integrado de Telemedicina y Telemedicina (STT) del Estado de Santa Catarina, la integración de un PACS a la infraestructura de telemedicina es obligatoria. Dado que STT proporciona herramientas para la evaluación remota de los exámenes por parte de especialistas que utilizan

interfaces web, la adquisición, transmisión, almacenamiento y disponibilidad de imágenes son condiciones previas para que se realicen estas evaluaciones.

En los resultados, los autores mencionan que la red PACS integrada a STT ha estado activa desde 2005, proporcionando la infraestructura para la comunicación y el almacenamiento de datos de especialidades relacionadas con la radiología. Hoy, los hospitales y las Unidades de Telemedicina Avanzada utilizan la red para el manejo de procedimientos en radiología computarizada, ultrasonido, tomografía computarizada, resonancia magnética y endoscopia.

Este trabajo presenta un caso de estudio para la integración de una red PACS a una infraestructura de telemedicina en todo el estado. El proceso de integración incluye hospitales y unidades avanzadas de telemedicina, lo que permite la adquisición, transmisión, almacenamiento y uso de contenido digital, consolidado para acceder a través de la web.

3.2 Herramientas existentes para el desarrollo de sistemas PACS

En el trabajo desarrollado por Valente y otros (Valente et al., 2015) describen *Dicoogle*, un marco de software que permite a los desarrolladores e investigadores crear rápidamente prototipos e implementar nuevas funciones aprovechando los servicios integrados de Imagen Digital y Comunicaciones en Medicina (DICOM). Lo que hace que *Dicoogle* sea único es su mentalidad hacia la exploración y aplicación de enfoques tecnológicos innovadores al tiempo que proporciona la solidez requerida de un PACS para sus operaciones diarias. El marco de software proporciona un entorno simbiótico donde las preocupaciones múltiples se separan en unidades autónomas orquestadas y administradas por la aplicación central.

Los autores afirman que el elemento central de los sistemas PACS es, típicamente, un servidor de archivo central que almacena imágenes adquiridas por las modalidades, junto con información complementaria sobre pacientes y estudios. Los datos del archivo pueden ser consultados por una multitud de equipos, como sistemas CAD y estaciones de trabajo con pantalla. DICOM es un protocolo orientado a objetos extensible y define varios servicios normalizados que permiten la comunicación entre diferentes componentes PACS. Por ejemplo, permite que las modalidades envíen directamente las imágenes adquiridas al archivo PACS y su posterior recuperación por las estaciones de trabajo del profesional con fines de diagnóstico. El protocolo DICOM es de gran importancia porque define cómo los datos de imágenes médicas y los metadatos respectivos deben almacenarse, recuperarse y transmitirse, permitiendo así la interoperabilidad entre diferentes modalidades y aplicaciones. La especificación de red de DICOM comprende un protocolo de capa de aplicación que utiliza TCP/IP para mover datos a través de la red y un mecanismo de direccionamiento basado en la entidad de aplicación (AE). Otra parte de DICOM es el acceso web a objetos persistentes DICOM (WADO). WADO⁶ es un intento de conectar las tecnologías web y la infraestructura DICOM al permitir el acceso RESTful HTTP a objetos persistentes a nivel de imagen. La falta de mecanismos estándar de consulta y almacenamiento en el WADO original ha llevado a varias extensiones de la arquitectura base, y recientemente, a la adopción por parte de NEMA⁷ del WADO utilizando los Servicios RESTful (WADO-RS), Almacenar en la web (STOW-RS) y Consulta basada en ID para objetos DICOM (QIDO-RS). Estas extensiones permiten, respectivamente, el almacenamiento y la consulta de objetos DICOM a través de HTTP.

⁶ WADO (Web Access to DICOM Objects) - Acceso web a objetos DICOM.

⁷ NEMA (The National Electrical Manufacturers Association) - La Asociación Nacional de Fabricantes Eléctricos.

Los autores mencionan que debido a la experiencia de campo obtenidas del proyecto Dicoogle, los llevaron a separar y racionalizar las funcionalidades proporcionadas en múltiples categorías: almacenamiento, indexación, consulta, servicio y presentación. Cada categoría está asociada a una interfaz particular, cuya implementación se carga en tiempo de ejecución, como un complemento. Depende de la aplicación principal orquestar las operaciones proporcionadas por los diversos complementos.

En la categoría de almacenamiento, una parte crucial de cualquier PACS es la persistencia de los estudios de imagen médica. Una implementación estándar de Dicoogle se basa en el sistema de archivos local como el *backend* de elección para almacenar objetos DICOM persistentes. Los complementos de indexación son los componentes responsables de organizar los datos en un formato que permita un acceso rápido a la información almacenada, mientras que los complementos de consulta median el acceso a esa información. Esta solución armoniza los procesos de extracción, almacenamiento y recuperación de información, permitiendo así la exploración de otros mecanismos de representación y recuperación de información, como la consulta, por ejemplo, y el estudio de la escalabilidad y el rendimiento de distintas bases de datos. Dicoogle es compatible con algunos de los servicios DICOM más importantes, como C-STORE, C-FIND y C-MOVE, y WADO. Las acciones de almacenamiento y recuperación DICOM son totalmente compatibles a través del comando C-STORE (también utilizado en operaciones C-MOVE). Las operaciones DICOM C-STORE generalmente se llaman cuando una modalidad (o cualquier otra entidad) produce una imagen o estudio y la envía al archivo PACS. Dicoogle tiene tres modos de interfaz con el usuario. Puede funcionar como una aplicación de línea de comandos independiente donde se puede crear una secuencia de comandos para su funcionalidad de indexación, análisis y consulta. Para los usuarios finales, ofrecen una interfaz gráfica de usuario

(GUI) basada en la invocación de métodos remotos (RMI) que permite que el servidor PACS y el motor de datos se ejecuten en una máquina y por último una la interfaz gráfica.

3.3 Metodología y arquitectura de sistemas PACS

El trabajo realizado por Lyu y otros (Lyu et al., 2015), mencionan que todo el sistema consta principalmente de dos módulos, que incluyen integración de datos y gestión de datos.

En el módulo de la integración de datos los autores mencionan que la arquitectura de la integración de datos clínicos basada en la plataforma Hadoop⁸ se compone de tres partes: los sistemas de información clínica, el motor de procesamiento de mensajes basado en reglas y la plataforma Hadoop.

Primero, la plataforma Hadoop está construida por el clúster Hadoop, Zookeeper y el clúster HBase. Y en este marco, HDFS⁹ y MapReduce¹⁰ pueden proporcionar la función básica de almacenamiento distribuido y procesamiento. Zookeeper se puede usar para coordinar estas aplicaciones distribuidas y administrar las configuraciones relacionadas. Y HBase es una base de datos NoSQL orientada a columnas, que es una de las bases de datos no relacionales. Además, HBase tiene las enormes ventajas sobre el sistema tradicional de gestión de bases de datos relacionales (RDBMS) en el almacenamiento y operación de datos. Por lo tanto, se adopta HBase para diseñar el modelo de almacenamiento de datos para que pueda satisfacer las demandas, como la alta concurrencia, el almacenamiento masivo y la alta escalabilidad.

⁸ Hadoop - Es un framework open source para almacenar datos y ejecutar aplicaciones en clusters de hardware básicos. Proporciona un almacenamiento masivo para cualquier tipo de datos.

⁹ HDFS (Hadoop Distributed File System) - Sistema de ficheros distribuidos de Hadoop.

¹⁰ MapReduce – Es un framework de procesamiento de ficheros en paralelo en el cluster Hadoop.

Segundo, El motor de procesamiento de mensajes basado en reglas, que incluye la capa de interfaz y la capa de proceso de negocio, está diseñado por el Marco de integración de Apache Camel. Camel es una biblioteca muy útil que ayuda a procesar eventos o mensajes de muchas fuentes diferentes. El diseño de la capa de interfaz se decide por la forma en que Camel interactúa con los sistemas de información clínica y la plataforma Hadoop. Y después, se puede seleccionar el "Componente" correspondiente para conectarse con las aplicaciones externas y entregar el mensaje.

Finalmente, el sistema adopta HL7 para estandarizar los datos producidos por los sistemas de información clínica, como EMR, PACS, LIS, etc. HL7, es un conjunto de estándares internacionales para el intercambio de información médica, puede utilizarse para especificar la transferencia de datos electrónicos sobre la información clínica, financiera y administrativa. Y también, describe la observación y el tratamiento para el paciente utilizando el modelo de clase básico en HL7, de modo que pueda dar una descripción completa de las transacciones médicas.

En el módulo de gestión de datos el *Big Data* tiene algunas características importantes, como el volumen, la variedad, la velocidad, la veracidad. Debido a los límites de la plataforma tradicional, no podía estar a la altura del desafío. Pero a través de la integración de Big Data, esto puede extender la plataforma tradicional a la plataforma Hadoop para que pueda funcionar en un entorno heterogéneo y proporcionar servicios de gestión de datos más eficientes y confiables. Su arquitectura se puede dividir en cinco partes.

- 1) Capa de aplicación: es el principal responsable de recibir las solicitudes enviadas por los usuarios, y luego envía la tarea a los módulos de consulta y la capa de procesamiento para su manejo. Finalmente, presenta los resultados a los usuarios a través del portal de administración.

- 2) Capa de consulta y procesamiento: de acuerdo con la información del tipo de consulta y la distribución de datos, analiza y optimiza principalmente las solicitudes, de modo que pueda generar la programación ejecutiva de MapReduce.
- 3) Capa de control de datos: utiliza la información de metadatos y el índice global para localizar los datos, y luego entrega la programación de MapReduce al motor de ejecución para realizar la consulta.
- 4) Capa de almacenamiento de datos: se encarga del almacenamiento de datos real, el diseño del índice de datos y la gestión del búfer y el archivo de diario. Además, adopta la estructura maestro-esclavo para que pueda mejorar la eficiencia del almacenamiento de datos en cada nodo.
- 5) Servicio y gestión: Proporciona principalmente la gestión de metadatos, el servicio de orquestación, la gestión tolerante a fallos, la supervisión del estado y el servicio de asignación de recursos.

Como resultado, los sistemas de información clínica pueden interactuar con el motor de procesamiento de mensajes a través del mensaje HL7, y luego estos datos estandarizados pueden enviarse a HBase para su almacenamiento. Sobre la base del intercambio de datos, la información podría ser gestionada por el sistema de gestión de datos clínicos. Además, también puede recuperar y consultar los registros médicos sobre cualquier paciente.

El trabajo realizado por Mineiro y Tomé (Mineiro & Tomé, 2015) proponen un modelo de integración de instrumentos RIS, PACS y radiología, como se muestra en la Figura 3.2.



Figura 3.2. Modelo de Integración.

Así mismo realizan un ejemplo de escenario que comienza con la llegada de un paciente a una unidad de radiología con un episodio de emergencia.

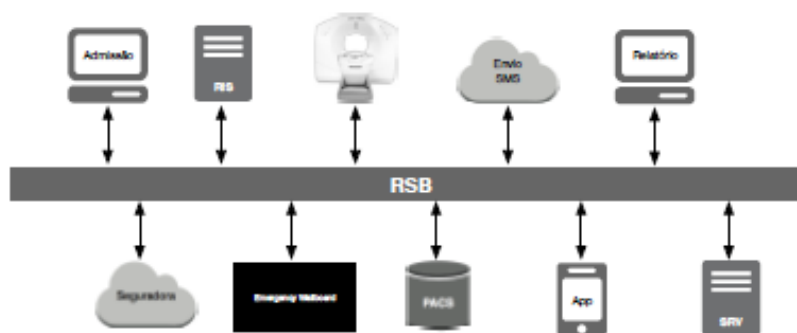


Figura 3.3. Escenario de interconexión e interoperabilidad de RSB.

Los pasos involucrados en el proceso se muestran a continuación. Cada vez que un paso en particular desencadena una acción automática del sistema, estas acciones automáticas se identifican correctamente como subtítulos de ese mismo paso:

1. El asistente de admisión recibe al paciente y localiza el expediente clínico del paciente consultando a la RSB.

2. Después de confirmar las observaciones necesarias, las registrarán. El sistema solicita la validación del registro a través de RSB del asegurador responsable de los aspectos financieros del episodio.
3. La RSB reenvía la solicitud de observación al servicio externo de la aseguradora para validar los datos de la política. Completa este paso devolviendo la confirmación al sistema de admisión con los datos relevantes.
 - a) RSB envía automáticamente el registro a RIS, terminando en la lista de trabajo del departamento responsable del procedimiento.
4. Como se trata de un episodio de emergencia, el RSB envía una notificación al Wallboard del departamento, informando a los radiólogos presentes allí sobre la llegada de un paciente de emergencia.
5. El paciente es derivado rápidamente al departamento donde los técnicos de radiología adquieren las imágenes. El instrumento de adquisición envía las imágenes a RSB que archiva las imágenes en PACS.
 - a) RSB registra en RIS la cantidad de imágenes disponibles
 - b) La RSB enviará a un servicio de SMS externo una notificación dirigida al radiólogo del servicio para alertarlo de que existen imágenes médicas de un episodio de emergencia.
6. El radiólogo recibe la notificación en su Smartphone y, utilizando una aplicación instalada allí, solicita información de episodios de emergencia de RSB.
7. La aplicación web de gestión de informes solicita a la RSB todos los datos de episodios de emergencia: datos demográficos del paciente, informes anteriores y las imágenes médicas capturadas.

8. El radiólogo tiene algunas dudas durante la observación actual y le pide a la RSB imágenes de un episodio anterior que pueden ayudarlo a comprender mejor la situación.
9. El radiólogo se aclara y comienza a dictar su informe enviándolo a la RSB tan pronto como termine.
 - a) RSB envía automáticamente el dictado digital al SRV porque, según los parámetros del sistema, en este momento el departamento de mecanografía no tiene personal presente.
10. El SRV completa el proceso y envía el informe producido automáticamente a RSB.
 - a) El RSB adjunta el informe producido en el registro de observación enviando un mensaje al RIS.
11. La RSB notifica al radiólogo que su dictado ha sido transcrito y puede ser validado.
12. El radiólogo confirma y envía el informe con su firma digital adherida a la RSB.
 - a) RSB envía un mensaje a RIS para informar el episodio como completado.

El modelo propuesto resultó del análisis de los diversos sistemas comerciales existentes, así como de la experiencia laboral de uno de los autores. El modelo propuesto está actualmente implementado y en uso en el dominio.

El trabajo realizado por Wan y otros (Wan et al., 2016) diseña patrones de recuperación de imagen médica basados en el sistema PACS regional y analiza la anotación semántica, la construcción de la biblioteca de características visuales y otras tecnologías clave involucradas en el logro de la recuperación de múltiples patrones, a fin de aprovechar mejor el papel auxiliar inteligente de la imagen médica en la clínica diagnóstico. El sistema PACS es un sistema de gestión de imágenes médicas y más difícil que la base de datos de texto. El extraer la imagen por si solo enfrenta grandes desafíos. En la etapa actual, el sistema PACS se basa en gran medida en la

información de perfil de la estructura DICOM para completar sus categorías simples, y utiliza el nombre del paciente, la identificación, la coincidencia de palabras clave y otros métodos de recuperación. Por lo tanto, los métodos y resultados de recuperación deben cumplir estrictamente las condiciones de recuperación. A veces, sin embargo, este tipo de recuperación de precisión no puede satisfacer las necesidades reales, porque en muchos casos, las condiciones de coincidencia necesarias para la recuperación de precisión no se pueden expresar por sí mismas; o, en algunos casos, lo que se espera obtener o lo que solo se puede obtener son los posibles resultados de recuperación.

Wan y otros autores diseñan un patrón y proceso de recuperación del sistema de recuperación de imágenes médicas. Este sistema de recuperación admite tres patrones de recuperación:

- La recuperación tradicional basada en el sistema PACS utilizando el nombre del paciente, su identificación y otros métodos de concordancia de palabras clave;
- Recuperación basada en una biblioteca de imágenes jerárquicas y clasificadas y según la semántica de categoría marcada;
- Recuperación que combina la biblioteca de imágenes jerárquicas y clasificadas construidas y la biblioteca de características de categoría y la consulta de soporte de imágenes de ejemplo. Este es un patrón de recuperación eficiente que utiliza de manera integral la semántica de categoría y las características visuales.

Para lograr la recuperación por semántica de categoría y la recuperación por consulta de imágenes, los autores se basan en el siguiente cuadro de trabajo:

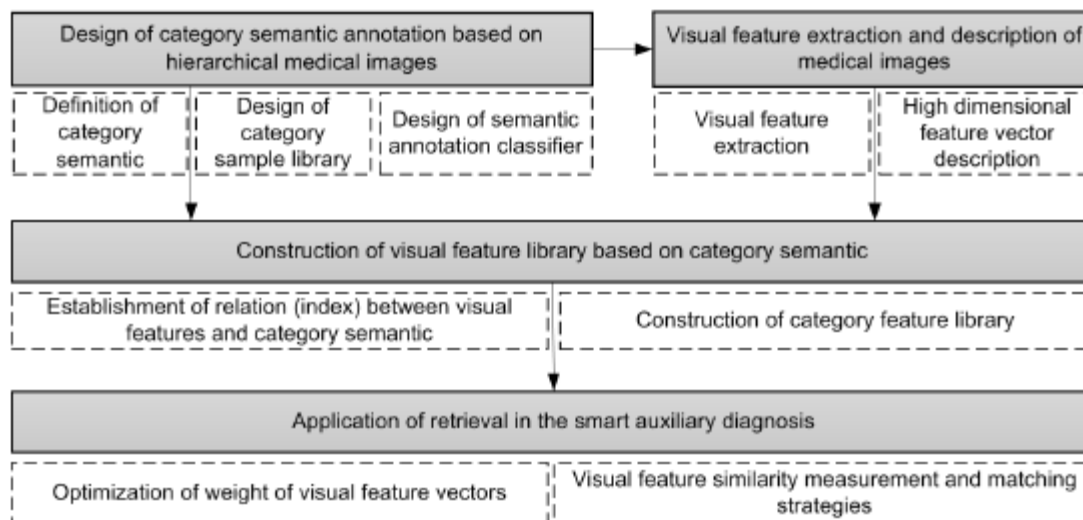


Figura 3.4. Módulos de patrones de recuperación.

La clave para la recuperación de imágenes médicas es determinar las similitudes entre las imágenes que se recuperarán (es decir, una consulta de las imágenes de ejemplo) y las imágenes específicas en la biblioteca de características de la misma categoría. Este es un proceso de medición de similitudes entre los vectores de características correspondientes de las imágenes a recuperar y los de las imágenes objetivo en las bases de datos, pero también un proceso de coincidencia de características visuales.

El trabajo realizado por Jodogne (Jodogne, 2018) describe el ecosistema Orthanc gratuito y de código abierto, cuyo desarrollo fue desencadenado en el 2012. El objetivo a largo plazo del proyecto Orthanc es fomentar la cooperación y el intercambio de conocimientos sobre el flujo de trabajo de imágenes médicas en hospitales (incluso con proveedores de dispositivos de imágenes y empresas de ciencia de datos), para estimular el desarrollo de nuevos algoritmos dedicados a imágenes médicas multimodales, como así como para mejorar la interoperabilidad en el cuidado de la salud al proporcionar un servidor DICOM que se puede implementar fácil y rápidamente en múltiples lugares en los hospitales como complemento del servidor PACS principal.

El componente central de este ecosistema es el servidor Orthanc. El servidor Orthanc es inherentemente un VNA (Archivo Neutral de Proveedores) que puede recibir, almacenar, indexar y transmitir imágenes médicas de acuerdo con el estándar DICOM. La arquitectura interna del servidor Orthanc viene con las siguientes propiedades:

- **Tamaño reducido:** puede ejecutarse en cualquier tipo de plataforma de hardware, desde Raspberry Pi hasta infraestructuras en la nube a gran escala a través de máquinas virtuales o computadoras de escritorio. Incluso es posible ejecutar el servidor Orthanc desde una memoria USB.
- **Multiplataforma:** escrito completamente en C ++ puro, se puede empaquetar fácilmente para muchos sistemas operativos. Los paquetes binarios para Microsoft Windows, Apple OS X, Docker y varias distribuciones GNU/Linux o UNIX (en particular Debian, Fedora y FreeBSD) están disponibles.
- **Independiente:** viene incluido con su propio motor de base de datos (SQLite), que evita la carga de configurar un servidor de base de datos para casos de uso simples. Del mismo modo, no se necesita ningún software o marco externo (como Java o .NET) para ejecutar el servidor Orthanc.
- **Cumple:** la mejor interoperabilidad DICOM se proporciona integrando el conocido kit de herramientas DCMTK. También se enfatiza la calidad y la validación automatizada de su código fuente.

Gracias a esta atención a la simplicidad del empaquetado, la implementación de un servidor Orthanc es casi inmediata: en la mayoría de los casos, es suficiente descargar los archivos binarios y adaptar varios de opciones (como DICOM AET - Título de entidad de aplicación - del servidor) editando el archivo de configuración proporcionado. La interfaz de un servidor Orthanc con

modalidades de imágenes periféricas, incluido el PACS del hospital, equivale a declararlas en el archivo de configuración al proporcionar sus respectivos parámetros DICOM (AET, dirección IP y número de puerto TCP).

Además de su servidor DICOM incorporado, el servidor Orthanc también viene incluido con un servidor HTTP incorporado. Este servidor HTTP expone una interfaz de usuario web incorporada que se conoce como Orthanc Explorer. Esta interfaz de usuario permite a los usuarios de Orthanc navegar e interactuar con el contenido de la tienda DICOM, desde cualquier computadora cliente dentro de la red del hospital mediante el navegador web. El servidor Orthanc se puede configurar para usar HTTPS y asignar contraseñas a los usuarios, con el fin de asegurar el acceso a las imágenes médicas a través de la autenticación y el cifrado adecuados.

Orthanc Explorer permite la administración manual de los servidores Orthanc. Para controlar los flujos DICOM de manera automatizada, el servidor Orthanc también incorpora un motor de secuencias de comandos que se puede utilizar para definir reglas de enrutamiento utilizando el lenguaje Lua. Un script Lua puede monitorear la llegada de instancias DICOM y reaccionar en consecuencia, redirigiéndolas a otra modalidad o modificándolas si se cumple alguna condición. La secuencia de comandos de Lua es muy útil para tareas básicas de enrutamiento automático. Para escenarios más avanzados de automatización DICOM, el servidor Orthanc proporciona un acceso programático completo a todas sus funciones principales, bajo la forma de un REST API que se sirve a través de su servidor HTTP incorporado. Gracias a este REST API, los scripts o aplicaciones externas pueden controlar el servidor Orthanc, independientemente del lenguaje de programación que se use para desarrollarlos (como Python, C #, Java, Matlab).

Esta aplicación se utiliza para investigación, garantía de calidad y fines educativos. Es muy importante tener en cuenta que la REST API del servidor Orthanc se puede utilizar para transmitir

imágenes de manera segura entre dos sitios remotos a través de Internet, a través del protocolo HTTPS encriptado. Además de su motor Lua y su REST API, el núcleo del servidor Orthanc también se puede extender a través de complementos. Dichos complementos están codificados en C o C ++, y pueden usarse para servir nuevas páginas web o para agregar nuevos puntos finales en la REST API.

El motor de complementos de Orthanc también se puede utilizar para reemplazar la base de datos SQLite incorporada (puede almacenar hasta aproximadamente 50,000 instancias DICOM), por una base de datos preparada para la empresa (que puede almacenar una cantidad prácticamente ilimitada de instancias DICOM). El proyecto Orthanc propone un complemento gratuito y de código abierto para PostgreSQL que se ha informado que gestiona con éxito al menos 10 TB de datos de imágenes. El servidor Orthanc presenta capacidades de scripting ricas (a través de su motor Lua y su API REST) y compatibilidad integrada con tecnologías web, lo que lo hace muy versátil.

Capítulo 4. Desarrollo e implementación

Para el desarrollo de esta herramienta web se utilizó la metodología Kanban, donde establece que las actividades se deben de dividir en pequeñas tareas para reducir el tiempo de entrega, y disminuir los costos. En la Figura 4.1 se visualiza las actividades divididas en dos partes *INITIATIVES WORKFLOW* y *CARDS WORKFLOW*. Cada actividad de *INITIATIVES WORKFLOW* son divididas en pequeñas tareas en el apartado de *CARDS WORKFLOW*, a cada tarea se le asigna una fecha de entrega, prioridad (bajo, promedio, alto y critico) y responsable, así como el estado en que se encuentra.

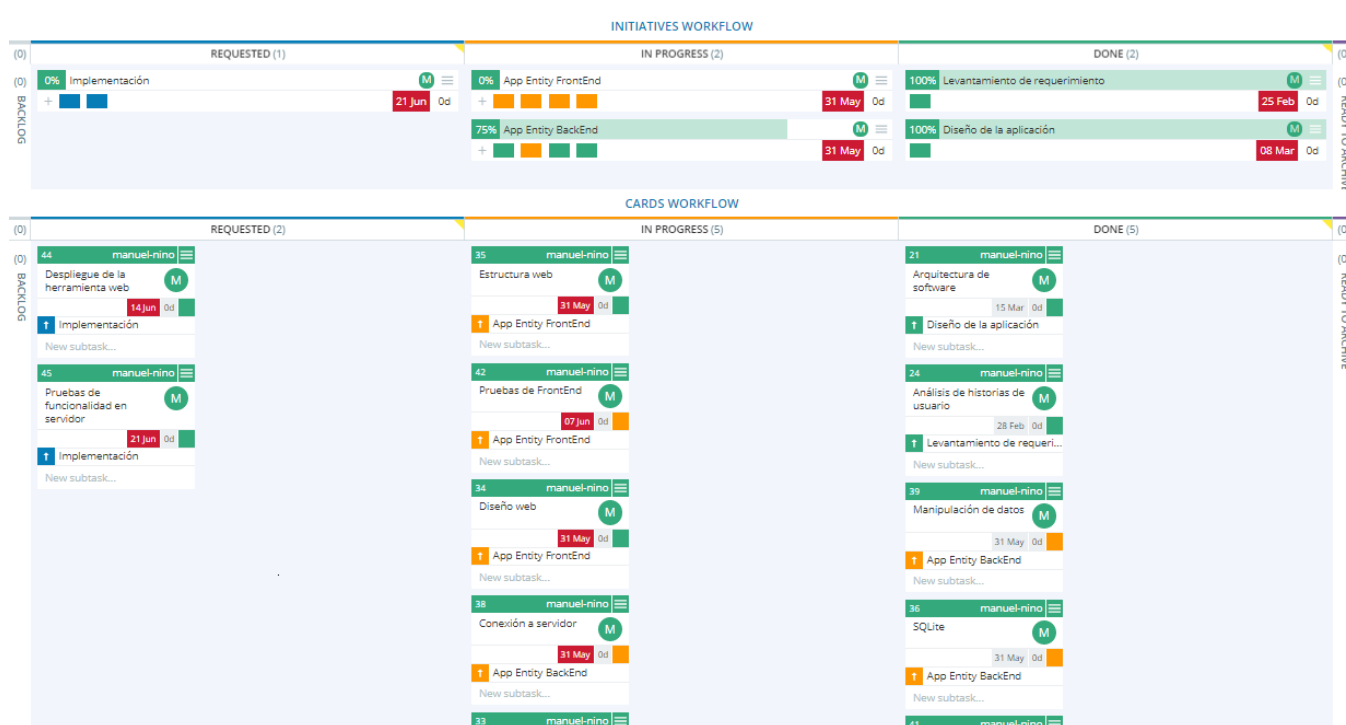


Figura 4.1. Flujo de trabajo Kanban.

La Figura 4.2 muestra el diagrama de bloques aplicado en este trabajo. Las etapas que lo integran, así como una descripción de cada uno se indican a continuación:

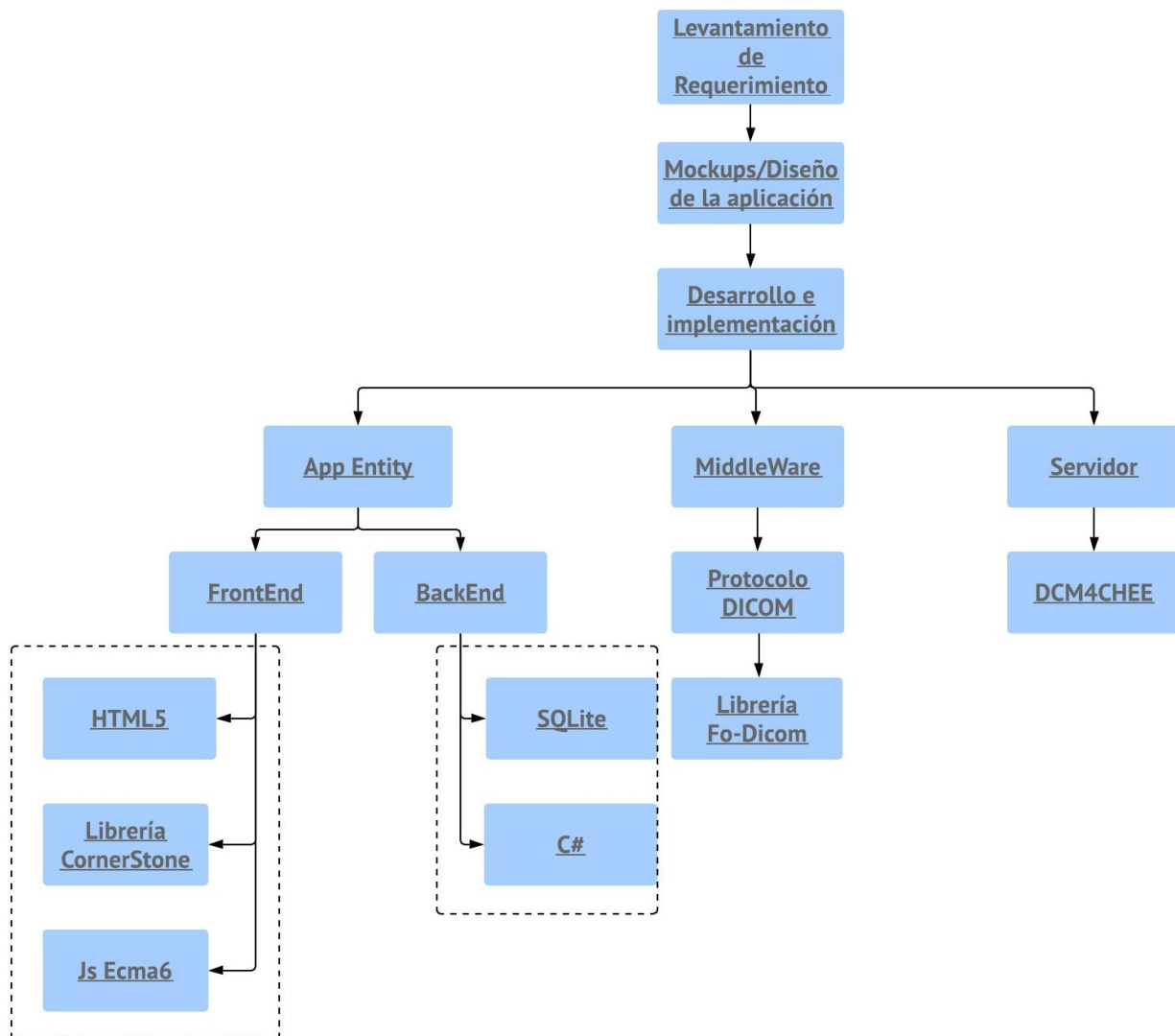


Figura 4.2. Diagrama de bloques.

Para el desarrollo de esta herramienta web es necesario hacer un levantamiento de requerimiento, para saber qué es lo que el cliente necesita, posteriormente se hace un análisis de los requerimientos para poder realizar el diseño y saber que herramientas utilizar, las etapas de App Entity, MiddleWare y Servidor trabajan de manera paralela debido a que se deben hacer pruebas de conexión con la App Entity y el servidor mediante el uso del MiddleWare, esas etapas

de la metodología es de suma importancia ya que las imágenes médicas que se manipularán se obtendrán del servidor DCM4CHEE.

4.1 Levantamiento de requerimientos.

Las tareas que se tienen que considerar para el desarrollo de este software son las siguientes:

- ❖ Establecer una conexión con el servidor PACS DCM4CHEE del IECan.
- ❖ Desarrollar una interfaz que permita la visualización de imágenes médicas en formato DICOM.
- ❖ Desarrollar una interfaz gráfica que permita la manipulación de las imágenes médicas con las siguientes funcionalidades:
 - Enfocar y desenfocar imagen – Permite hacer un acercamiento y alejamiento de la imagen.
 - Ajustar el nivel de brillo y contraste de la imagen – Permite mejorar el contraste y brillo de la imagen.
 - Desplazar la imagen dentro del contenedor – permite desplazar la imagen dentro del contenedor.
 - Realizar anotaciones en la imagen – Permite hacer anotaciones en la imagen.
 - Borrar anotaciones en la imagen – Permite borrar todas las anotaciones que hay en la imagen.
 - Invertir colores de la imagen – Permite invertir los colores de la imagen.
 - Rotar imagen – Permite rotar la imagen de 90° en 90°.
 - Sacar el ángulo en la imagen – Permite sacar el ángulo en partes de la imagen.

- Magnificar la imagen – Permite abrir una lupa y magnificar la parte que este bajo la lupa.
- Permitir la interpolación – También llamado *Nearest neighbor interpolation*, Permite crear nuevos pixeles a partir de los pixeles ya dado.
- Restablecer los valores por default de la imagen – Permite restablecer la imagen en su forma original.
- Marcar ROI en la imagen – Permite marcar en la imagen región de interés, en formas circular o rectangular.
- Realizar procesamiento de imagen – Permite utilizar diferentes tipos de técnicas de detección de bordes y suavizado. Para este procesamiento se utiliza la matriz de convolución, la convolución es la operación matemática de una matriz por otra que se llama "kernel, mascara o filtro" (Convolution Matrix, 2016). Para el procesamiento de imagen generalmente, primero se suaviza la imagen mediante la aplicación de filtros.
 - *Sharpen Filter* – Permite enfatizar la textura de la imagen, para ello se está utilizando el siguiente kernel (Figura 4.3):

$$1 \begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

Figura 4.3. Sharpen filter.

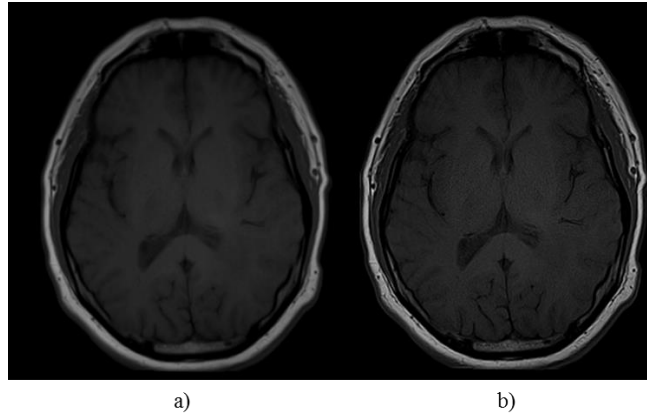


Figura 4.4. Filtrado de imagen. a) Imagen original, b) Resultado del Sharpen Filter.

- *Emboss Filter* – Permite reemplazar cada uno de los pixeles de una imagen por un resalto o una sombra, según sean los límites claros/oscuros de la imagen original. Se está utilizando el siguiente kernel (Figura 4.5):

$$\frac{1}{3} \begin{bmatrix} -2 & 1 & 0 \\ -1 & 1 & 1 \\ 0 & 1 & 2 \end{bmatrix}$$

Figura 4.5. Emboss filter.

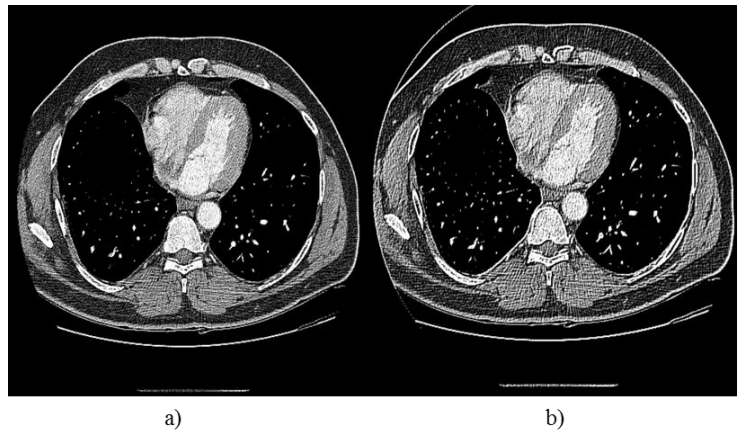


Figura 4.6. Filtrado de imagen. a) Imagen original, b) Resultado del Emboss Filter.

- *Edge Detection Filter* – Permite encontrar los límites o bordes de los objetos dentro de las imágenes. Se está utilizando el siguiente kernel (Figura 4.7):

$$1 \begin{bmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

Figura 4.7. *Edge Detection Filter*.

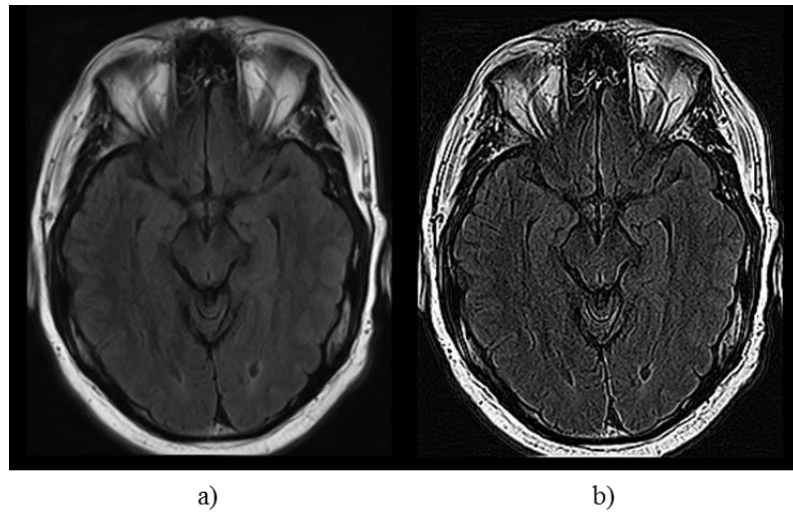


Figura 4.8. Filtrado de imagen. a) Imagen original, b) Resultado del *Edge Detection Filter*.

- *High-Pass Filter*– Permite realzar márgenes, líneas y bordes de dicha imagen. Se está utilizando el siguiente kernel (Figura 4.9):

$$\frac{1}{10} \begin{bmatrix} -1 & -2 & -1 \\ -2 & 22 & -2 \\ -1 & -2 & -1 \end{bmatrix}$$

Figura 4.9. *High-Pass Filter*.

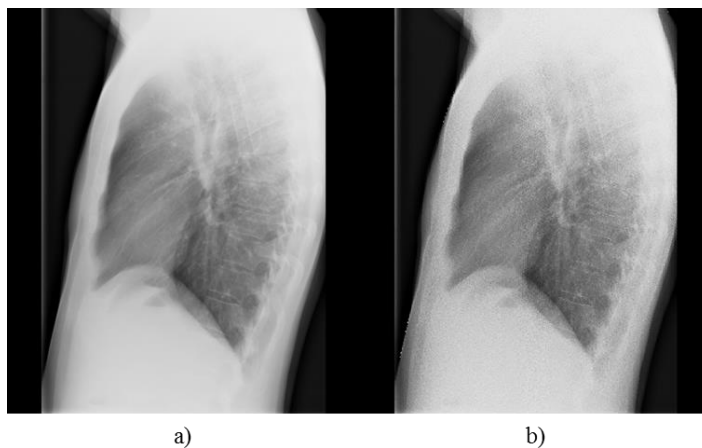


Figura 4.10. Filtrado de imagen. a) Imagen original, b) Resultado del High Pass Filter.

- *Unsharp Filter*– Permite aumentar el contraste de los bordes de los elementos. Se está utilizando el siguiente kernel (Figura 4.11):

$$\frac{-1}{256} \begin{bmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 16 & 24 & 16 & 4 \\ 6 & 24 & -476 & 24 & 6 \\ 4 & 16 & 24 & 16 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{bmatrix}$$

Figura 4.11. Unsharp Filter.

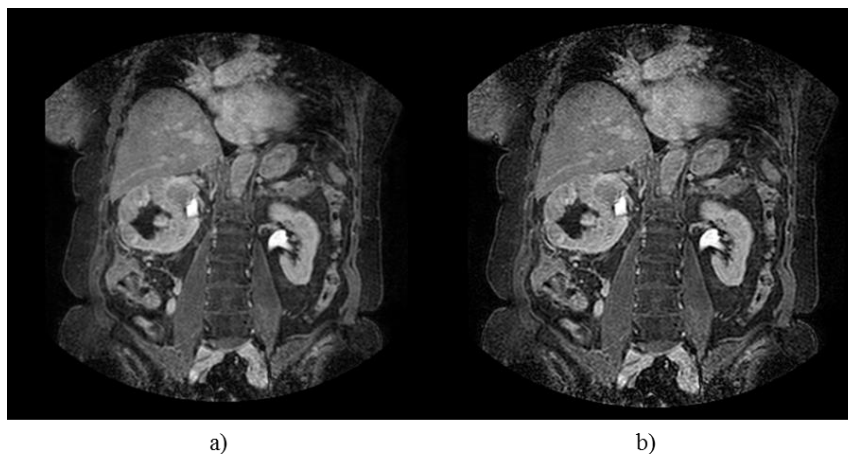


Figura 4.12. Filtrado de imagen. a) Imagen original, b) Resultado del Edge Unsharp Filter.

- *Gaussian Blur* – Permite reducir el ruido y detalles en la imagen. Se está utilizando el siguiente kernel (Figura 4.13):

$$\frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

Figura 4.13. *Gaussian Blur*.

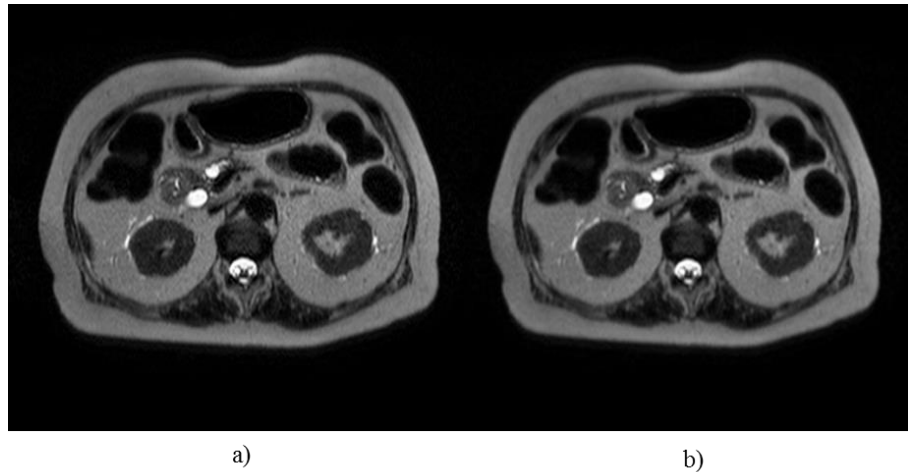


Figura 4.14. Filtrado de imagen. a) Imagen original, b) Resultado del *Gaussian Blur*.

- *Box Blur* – Permite promediar los valores de los píxeles de un vecindario (neighborhood) y así reducir la nitidez de la imagen. Se está utilizando el siguiente kernel (Figura 4.15):

$$\frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Figura 4.15. *Box Blur*.

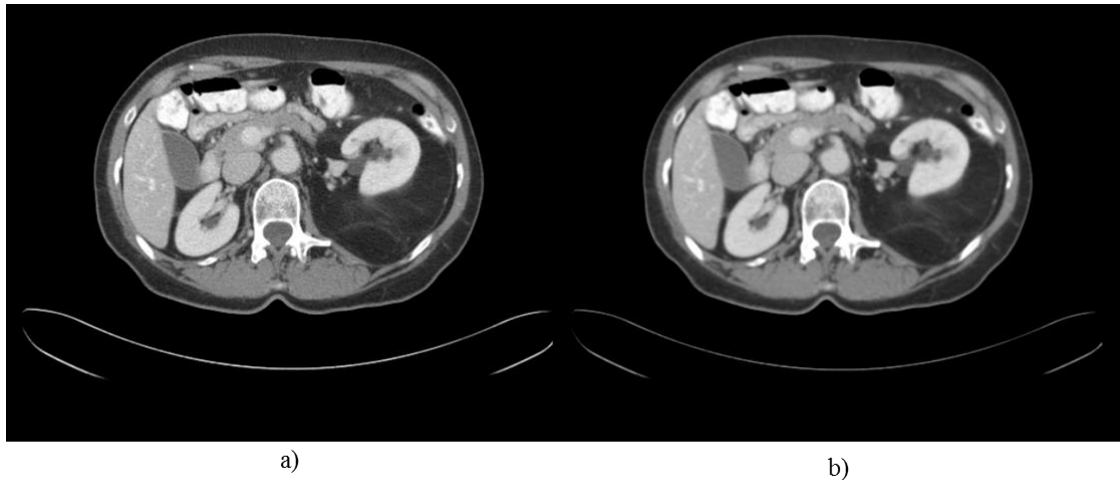


Figura 4.16. Filtrado de imagen. a) Imagen original, b) Resultado del Box Blur.

4.2 MockUp/Diseño de la Aplicación.

Según Pressman (2010), el diseño de software es realmente un proceso de muchos pasos, pero que se clasifican dentro de uno mismo. La actividad del diseño se refiere al establecimiento de las estructuras de datos, la arquitectura general del software, representaciones de interfaz y algoritmos. El proceso de diseño traduce los requerimientos necesarios para resolver un problema en una representación de software. En la Figura 4.17 se muestra el diagrama de despliegue de la herramienta web.

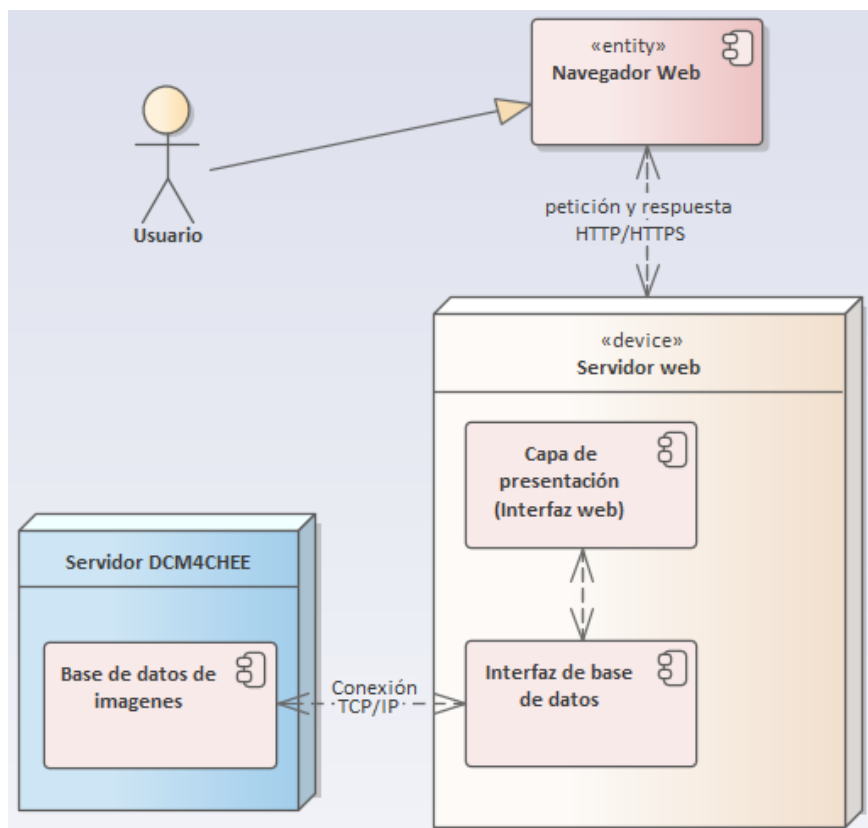


Figura 4.17. Diagrama de despliegue.

4.2.1 Arquitectura de la herramienta Software

La Arquitectura de la herramienta de software está compuesta por los dispositivos de adquisición de imágenes médicas, servidor de imágenes y la herramienta web, esta última es la que va a permitir visualizar y manipular las imágenes médicas. En la Figura 4.18 se muestra como está estructurada la arquitectura mencionada.

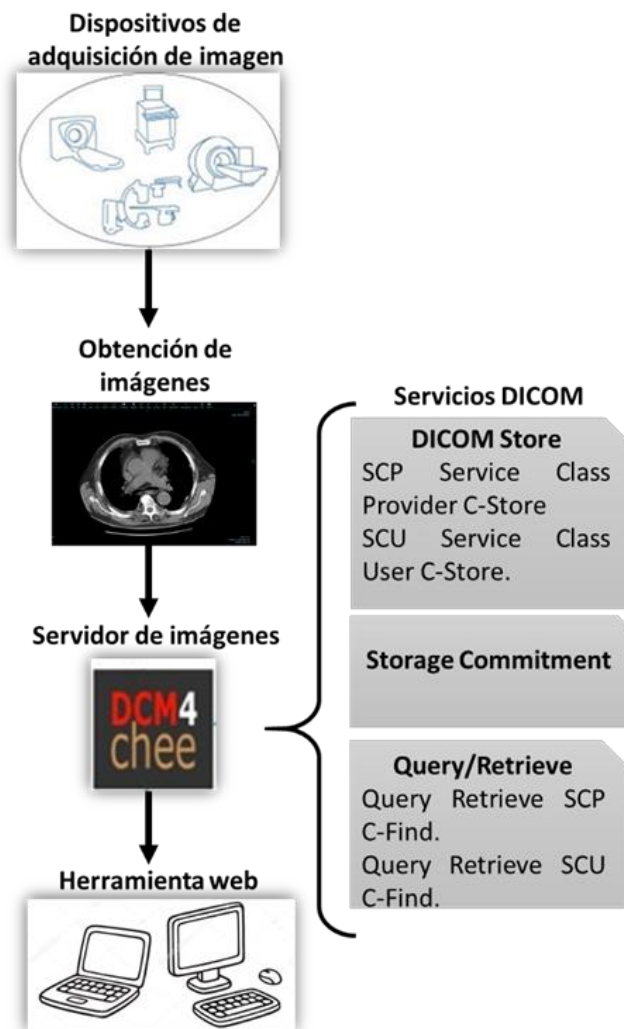


Figura 4.18. Arquitectura de la Herramienta Web.

Mediante los dispositivos de adquisición se obtienen las imágenes médicas, las cuales se crean en formato DICOM, para almacenar estas imágenes primero se tiene que vincular mediante IP el servidor PACS DCM4CHEE con los dispositivos de adquisición de imágenes, después se utiliza el comando C-STORE para almacenar las imágenes en el servidor, el C-STORE es la operación DICOM más básica, también conocida como "DICOM Push" permite que una SCU (Service Class User) envíe una instancia compuesta a un SCP (Service Class Provider). Por ejemplo, se utiliza para enviar imágenes de una modalidad a PACS. Mediante el uso de "Storage

Commitment” permite que un creador de imágenes u otras instancias compuestas compruebe que un servidor (el SCP) las ha almacenado de forma segura antes de eliminarlas de su propia memoria caché.

Para recuperar y visualizar las imágenes del servidor, la herramienta web tiene que hacer un “handshake” con el servidor PACS, donde se deben establecer los siguientes parámetros: dirección IP, puerto, título AE (Application entity) y el comando C-MOVE. El comando C-MOVE es el único protocolo de recuperación utilizado por la mayoría de los servidores PACS.

Por último, para hacer una búsqueda de alguna imagen en específico se realiza mediante el comando C-FIND, inicialmente utilizado como parte del servicio de Consulta/Recuperación, pero posteriormente reutilizado en los servicios de Lista de Trabajo de Modalidad y Lista de Trabajo de Propósito General, esta es una operación muy simple similar a una consulta SQL, por la cual se pasa un conjunto de datos desde la SCU al SCP contiene 2 tipos de atributo:

- Aquellos que necesitan coincidir (equivalentes a la cláusula WHERE de SQL). Estos tienen valores "rellenados"
- Aquellos que se devolverán a la SCU (equivalente a la cláusula SELECT de SQL). Estos se envían un campo en blanco.

El SCP responde enviando una serie de conjuntos de datos coincidentes, seguidos de una respuesta "completa" para indicar que ha finalizado.

4.3 Desarrollo e implementación

El desarrollo de software consiste en la elaboración de un producto basado en las especificaciones del cliente, es decir, el análisis de su actividad o negocio. Es decir, el producto se adapta a las necesidades del usuario. Para ello se requiere una clara identificación de las necesidades planteadas y su traducción en especificaciones de software, la evaluación de este proceso de desarrollo determinará la efectividad de su aplicación, así como otros factores, instalación del software, ciclo de vida del software, etc.

4.3.1 Servidor

Actualmente el Instituto Estatal de Cancerología almacena todas sus imágenes médicas a través de la aplicación DCM4CHEE la cual está alojada en el servidor Ubuntu Server 18.04.3 LTS. El núcleo de esta aplicación es una implementación robusta del estándar DICOM. El DCM4CHEE *DICOM toolkit y library* se usa en muchas aplicaciones de producción en todo el mundo y la versión actual (5.x) ha sido rediseñada para un alto rendimiento y flexibilidad.

DCM4CHEE es un Administrador de imágenes/Archivo de imágenes. La aplicación contiene los servicios e interfaces DICOM, HL7 que se requieren para proporcionar almacenamiento, recuperación y flujo de trabajo a un entorno sanitario. *dcm4chee* está pre-empaquetado y desplegado dentro del servidor de aplicaciones *JBoss*. Al aprovechar muchas características de *JBoss* (Java Message Service, Enterprise Java Beans, Servlet Engine) y asumir el papel de varios actores de IHE (Integrando la Empresa de Salud) para el bien de la interoperabilidad, la aplicación proporciona muchos servicios robustos y escalables:

Tabla 4.1. Tabla de servicios DCM4CHEE

Servicios	Descripción
Interfaz de usuario basado en Web	dcm4chee contiene una interfaz de usuario robusta para administradores que se ejecuta completamente en un navegador web.
Almacenamiento DICOM	Actuando como un archivo, dcm4chee puede almacenar cualquier tipo de objeto DICOM en sistemas de archivos estándar, con compresión si es necesario.
Consultar/Recuperar DICOM	Consultar el archivo en busca de objetos DICOM y recupérellos.
WADO(Acceso Web a Objetos DICOM) y RID(Recuperar Información para Mostrar)	Acceso web al contenido archivado.
Otros servicios DICOM	MPPS(Modalidad realizada en Procedimiento de Paso), GPWL(Lista de trabajo de uso general), MWL(Lista de trabajo de modalidad), Compromiso de almacenamiento, Notificación de disponibilidad de instancia, Notificación de contenido de estudio, Contenido de salida a CD, Protocolos colgantes.
Servidor HL7 (Alto Nivel 7)	Un servidor HL7 integrado que puede actuar sobre los tipos de mensajes ADT(Admitir, Descargar, Transferir), ORM(Orden Farmacia / tratamiento para solicitar algo, pedido de laboratorio) y ORU(Resultado de observación).

Servicios IHE(Integrando la Empresa de Salud)	dcm4chee puede existir felizmente en un entorno compatible con IHE al integrarse con un registro y repositorio XDS(Intercambio de documentos entre empresas)/XDSI(Intercambio de documentos entre empresas para imágenes), actuar como un nodo seguro y proporcionar auditorías compatibles.
--	--

La aplicación DCM4CHEE cuenta con 33 tablas en su base de datos para el almacenamiento de las imágenes DICOM e información, como se muestra en la tabla 4.2.

Tabla 4.2 Tablas de la base de datos DCM4CHEE.

Tablas	Hijos	Padres	Columnas	Filas
<u>code</u>	7		5	0
<u>content item</u>		3	6	0
<u>dicomattrs</u>	7		2	0
<u>diff task</u>	1	1	14	0
<u>diff task attrs</u>		2	2	0
<u>export task</u>		1	13	0
<u>hl7psu task</u>		1	7	0
<u>ian task</u>		1	7	0
<u>id sequence</u>			3	0
<u>instance</u>	3	3	21	0
<u>issuer</u>	5		4	0

<u>location</u>		2	12	0
<u>metadata</u>	1		7	0
<u>mpps</u>	2	4	14	0
<u>mwl item</u>	1	4	16	0
<u>patient</u>	4	5	18	0
<u>patient id</u>	1	1	5	0
<u>person_name</u>	8		16	0
<u>queue_msg</u>	4		18	0
<u>rejected_instance</u>		1	7	0
<u>rel_study_pcode</u>		2	2	0
<u>retrieve_task</u>		1	19	0
<u>series</u>	3	5	46	0
<u>series_query_attr</u>		1	7	0
<u>series_req</u>		3	9	0
<u>soundex_code</u>		1	5	0
<u>sps_station_aet</u>		1	2	0
<u>stgcmnt_result</u>			14	0
<u>stgver_task</u>		1	13	0
<u>study</u>	3	4	29	0
<u>study_query_attrs</u>		1	9	0
<u>uidmap</u>	1		2	0
<u>verify_observer</u>		2	4	0

33 tablas			358	0
------------------	--	--	------------	----------

Como se mencionó anteriormente DCM4CHEE tiene una interfaz web donde se puede visualizar la información que tiene el servidor, más no es posible abrir las imágenes médicas almacenadas, para ello primero se tiene que acceder con un usuario y contraseña.

Posteriormente se muestra la interfaz con toda la información almacenada en el servidor, tal como se ilustra en la Figura 4.19.

The screenshot shows the DCM4CHEE web interface. At the top, there is a navigation bar with tabs like 'Folder', 'Trash', 'Application Entities', 'Modality Worklist', 'Teaching-Files', 'Dashboard', 'Roles', 'Users', and 'Password'. Below this is a search bar with fields for Patient Name, Phonetic, Patient ID, Issuer, Study Date (from/to), and Accession No. There are also options for Modality, Source AET, and search filters. The main content area displays a table of search results with the following columns: Patient Name, Patient ID/Issuer, Birth Date, Sex, Comments, Study Date/Time, Study ID, Accession No, Modality, Description, #S/#I, and Availability. The table contains 18 rows of data, including entries for patients like ABDULKADIRI YUSUF, ABOYI PIUS OGBOLE, ADEYANJU GRACE, ALABI JANET, AYONNAIKE ADEDAPO, BABY NDUKA, and NDUKA BABY.

Study Date/Time	Study ID	Accession No	Modality	Description	#S/#I	Availability
10/3/2017 15:16	2726		CT	UROGRAPHY	5/216	ONLINE
9/24/2017 14:40	2721		CT	CHEST	4/141	ONLINE
9/25/2017 18:58	2722		CT	ABD-PELVIC	7/967	ONLINE
9/13/2017 11:19	2715		CT	BRAIN	3/140	ONLINE
9/19/2017 11:42	2718		CT	ABDOMINOPELVIC	4/132	ONLINE
9/20/2017 11:16	2719		CT	RT THIGH	4/144	ONLINE
9/23/2017 14:39	2720		CT	ABD-PELVIC	4/143	ONLINE
9/18/2017 15:03	2717		CT	ORBITAL	5/141	ONLINE
9/18/2017 14:10	2716		CT	ORBITAL	4/139	ONLINE

Figura 4.19. Interfaz web de DCM4CHEE.

4.3.2 MiddleWare

Para establecer una conexión entre la aplicación Entity y El servidor DCM4CHEE es necesario utilizar un middleware. Básicamente, funciona como una capa de traducción oculta para permitir la comunicación y la administración de datos en aplicaciones distribuidas. A veces, se le denomina “plumbing” (tuberías), porque conecta dos aplicaciones para que se puedan pasar fácilmente datos y bases de datos por una “canalización”. El uso de middleware permite a los usuarios hacer solicitudes como el envío de formularios en un explorador web o permitir que un servidor web devuelva páginas web dinámicas en función del perfil de un usuario (Microsoft Azure, 2019).

Para lograr esta conexión se está utilizando la librería Fo-Dicom. Es una librería *Open Source*, la cual tiene las siguientes características:

- Biblioteca de clases portátil (PCL).
- Está dirigida a .NET 4.5 y superior, .NET Core (.NET Standard 1.3 y superior), Plataforma universal de Windows, Xamarin iOS, Xamarin Android, Mono y Unity.
- Diccionario DICOM versión 2018b.
- API asíncrono/en espera totalmente asíncrona de alto rendimiento.
- Compresión de imágenes JPEG (incluidas sin pérdida), JPEG-LS, JPEG2000 y RLE (limitado en las plataformas .NET Core, Xamarin, Mono y Unity).
- Admite conjuntos de datos muy grandes con carga de contenido bajo demanda.
- Representación de imagen específica de la plataforma.
- Soporte JSON.
- Exportación XML (vista previa).
- Anonimización (vista previa) código de conexión.

Para hacer uso de la librería, se instaló desde el *NuGet Package Manager*, una herramienta de Visual Studio que contienen código reutilizable que otros desarrolladores ponen a disposición para usar en los proyectos.

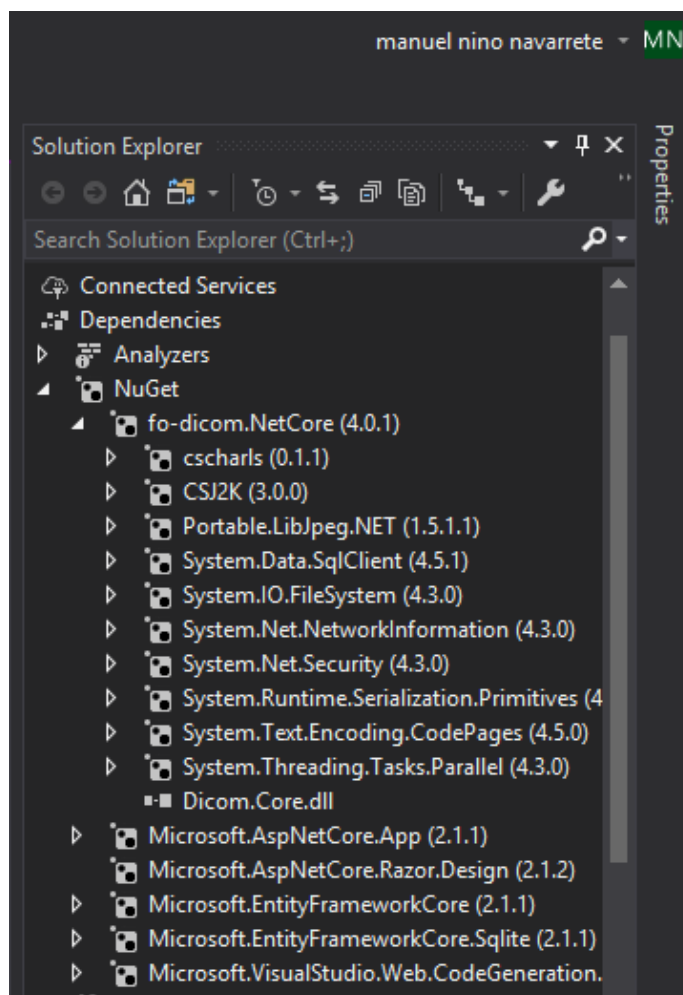


Figura 4.20. Instalación de la librería Fo Dicom.

En la Figura 4.21 se visualiza la configuración de la conexión de la aplicación Entity con el servidor.

```
namespace Herramienta_para_manupular.Infraestructure
{
    public class ConfigPacs
    {
        public static string host = string.Empty;
        public static int puerto = 0;
        public static string applicationEntityTitle =
string.Empty;
        public static string calledAET = string.Empty;
    }
}
```

Figura 4.21. Configuración de la conexión.

Para realizar una conexión entre la aplicación y el servidor es necesario establecer correctamente los parámetros siguientes:

- Host
- Puerto
- Título de la aplicación
- Aplicación llamada

Sin estos parámetros resultaría imposible realizar la conexión.

4.3.3 App Entity

Para la creación de esta herramienta se está utilizando el Entorno de Desarrollo Integrado Visual Studio con ASP.NET.

ASP.NET es un marco de desarrollo web popular para crear aplicaciones web en la plataforma .NET. ASP.NET Core es la versión de código abierto de ASP.NET, que se ejecuta en macOS, Linux y Windows.

4.3.3.1 BackEnd

El BackEnd toma los datos, los procesa y los envía al usuario, además de encargarse de las consultas o peticiones a la Base de Datos, la conexión con el servidor, entre otras tareas que debe realizar en su día a día.

Se Utiliza C# para hacer que todo lo que está detrás de un sitio web funcione correctamente. Además de que es un lenguaje de programación sencillo, moderno, orientado a objetos y con seguridad de tipos. Todo el BackEnd de la Herramienta para Manipular Imágenes Médicas está codificado en C#.

SQLite es una herramienta de software libre, que permite almacenar información en dispositivos empujados de una forma sencilla, eficaz, potente, rápida y en equipos con pocas capacidades de hardware.

Se está utilizando SQLite para almacenar de manera temporal los parámetros que el usuario ingresa al momento de establecer una conexión entre la aplicación y el servidor. El código de la creación de la base de datos se puede visualizar en la Figura 4.22. El código está escrito en el lenguaje C#.

```

using Herramienta_para_manupular.Models;
using Microsoft.EntityFrameworkCore;

namespace Herramienta_para_manupular.DAL
{
    public class IecanDbContext : DbContext
    {
        public IecanDbContext(DbContextOptions<IecanDbContext>
options)
            : base(options)
        {
        }
        public DbSet<PacsSettings> PacsSetting { get; set; }

        protected override void OnModelCreating(ModelBuilder
modelBuilder)
        {
            modelBuilder.Entity<PacsSettings>(entity =>
            {
                entity.HasKey(e => e.Id);
                entity.ToTable("pacs_settings");
                entity.Property(e => e.Id)
                    .HasColumnName("id").ValueGeneratedNever();
                entity.Property(e => e.Host)
                    .HasColumnName("host");
                entity.Property(e => e.Puerto)
                    .HasColumnName("puerto");
                entity.Property(e => e.callingAET)
                    .HasColumnName("calling_aet");

                entity.Property(e => e.calledAET)
                    .HasColumnName("called_aet");
            });

            base.OnModelCreating(modelBuilder);
        }
    }
}

```

Figura 4.22. Código de la creación de la base de datos SQLite.

El modelo de la base de datos SQLite se muestra en la Figura 4.23.


```

using System;
using System.Collections.Generic;
using System.ComponentModel.DataAnnotations;
using System.Linq;
using System.Threading.Tasks;

namespace Herramienta_para_manupular.Models
{
    public class PacsSettings
    {
        public int Id { get; set; }

        [Required(ErrorMessage = "Por favor introduzca la dirección ip o nombre de host")]
        [Display(Name = "Dirección ip o nombre de host")]
        public string Host { get; set; }

        [Required(ErrorMessage = "Por favor introduzca el número de puerto")]
        [Display(Name = "Puerto")]
        public int Puerto { get; set; }

        [Required(ErrorMessage = "Por favor introduzca calling AET")]
        [Display(Name = "Calling AET")]
        public string callingAET { get; set; }

        [Required(ErrorMessage = "Por favor introduzca el called AET")]
        [Display(Name = "Called AET")]
        public string calledAET { get; set; }
    }
}

```

Figura 4.23. Código del modelo de la base de datos SQLite.

4.3.3.2 FrontEnd

El FrontEnd se enfoca en el usuario, en todo con lo que podemos interactuar y lo que vemos mientras navegamos. Así como en una primera cita, nuestra web busca causar una buena impresión y agradar al usuario, para lo cual utiliza HTML, CSS y JAVASCRIPT. Buena experiencia de usuario, inmersión y usabilidad, son algunos de los objetivos que busca un buen frontend y hoy en

día existen una gran variedad de framework, preprocesadores y librerías que nos ayudarán en esta tarea.

Para el desarrollo del FrontEnd se está utilizando la versión más reciente HTML5, para lograr que tenga una funcionalidad correcta con los navegadores actuales.

En las Figuras 4.24 y 4.25 se visualiza el código HTML y la interfaz principal.

```
@{
    ViewData["Title"] = "Inicio";
}

<div id="homeIndex">

    <div class="row">
        <div class="col-md-12">
            <h2 class="text-center"><b>Aplicación desarrollada por el Instituto
Tecnológico de Acapulco en colaboración con el Instituto Estatal de
Cancerlogía</b></h2>
            <h3 class="text-center" style="line-height: 10px;"><i>Dr. Arturo
Beltrán Ortega</i></h3>
            <p></p>
        </div>
    </div>
    <div id="myCarousel" class="carousel slide" data-ride="carousel" data-
interval="6000">
        <ol class="carousel-indicators">
            <li data-target="#myCarousel" data-slide-to="0" class="active"></li>
        </ol>
        <div class="carousel-inner" role="listbox">
            <div class="item active">
                
            </div>
        </div>
    </div>

    <div id="redesSociales" class="row">
</div>
```

Figura 4.24. Código de la interfaz principal.

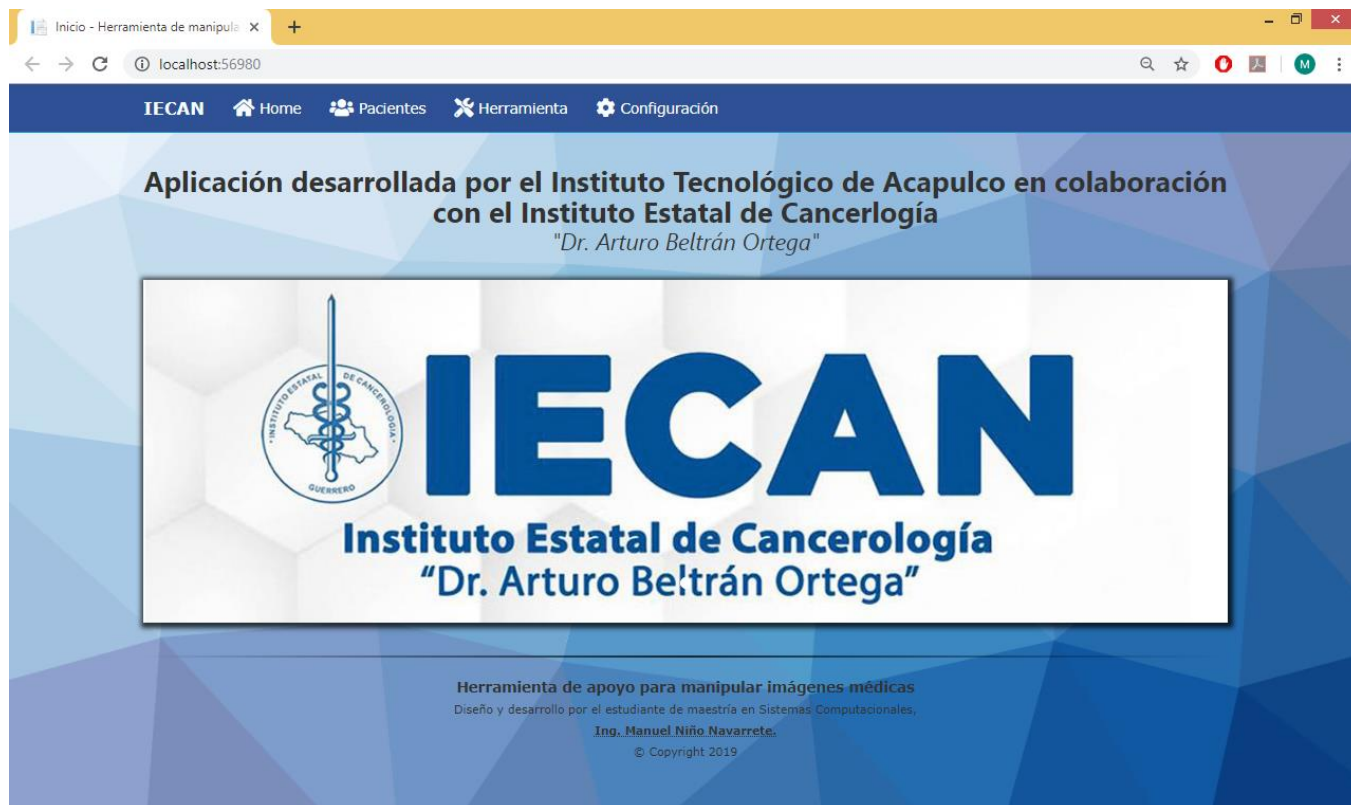


Figura 4.25. Interfaz principal del sistema.

Junto con HTML5 se utiliza la librería Cornerstone.js es una biblioteca ligera de JavaScript para mostrar imágenes médicas en navegadores web modernos que admiten el elemento de lienzo HTML5. Es una librería de *Open Source*, se utiliza esta librería para la parte de manipulación de las imágenes.

En la Figura 4.26, se puede visualizar la interfaz de la herramienta para manipular las imágenes, esta se desarrolló con HTML5 y JavaScript, para que cada botón tenga una funcionalidad se utilizó la librería Cornerstone.js.

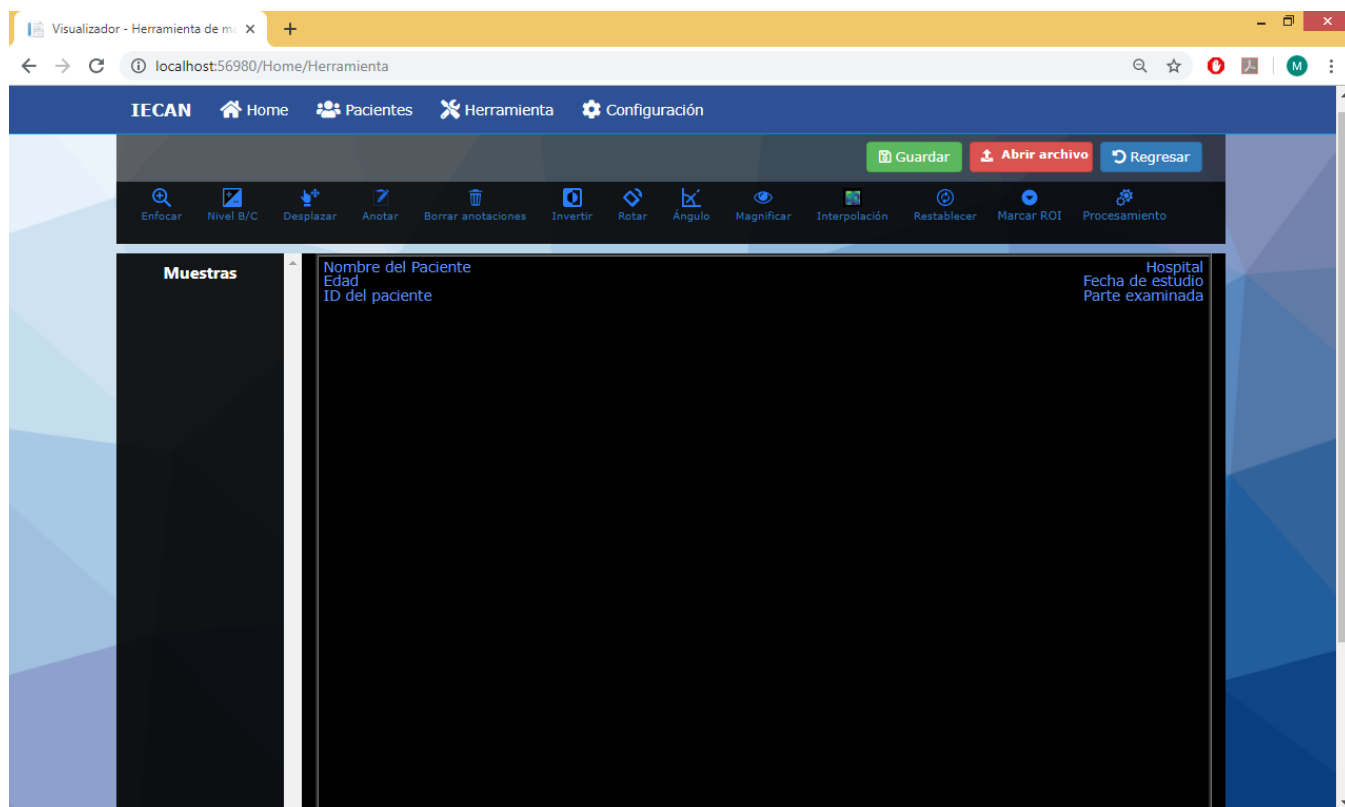


Figura 4.26. Interfaz de la herramienta para manipular imágenes.

4.4 Implementación

La herramienta desarrollada se implementó en el Servidor HP ProLiant ML350p Gen8, en el departamento de Informática del Instituto Estatal de Cancerología. El servidor cuenta con el Windows Server 2008 R2, lugar donde se desplegó la Herramienta para Manipular las Imágenes Médicas.

4.4.1 Configuración del Servidor

En la tabla 4.3 se puede visualizar las características del servidor donde se alojará la aplicación.

Tabla 4.3. Características del servidor HP.

Características	Descripción
Procesador	<p>E5-2600 v2 series Processors</p> <p>Entry processors:</p> <ul style="list-style-type: none"> • Intel Xeon E5-2609 (2.5 GHz/4 cores/10MB/6.4-GT/s QPI/80 W) <p>Base processors:</p> <ul style="list-style-type: none"> • Intel Xeon E5-2640 v2 (2.0 GHz/8-core/20MB/7.2GT/s QPI/95 W, DDR3-1600, HT, Turbo2- 3/3/3/3/3/3/4/5) <p>Performance processors:</p> <ul style="list-style-type: none"> • Intel Xeon E5-2697 v2 (2.7 GHz/12-core/30MB/8.0GT/s QPI/130 W, DDR3-1866, HT, Turbo2- 3/3/3/3/3/3/3/4/5/6/7/8)
Memoria del Cache	<ul style="list-style-type: none"> • 25MB (1 x 25MB) Level 3 (Para diez procesadores core)
Chipset	<ul style="list-style-type: none"> • Intel E5-2600 Processor Family
Capacidad de actualización	<ul style="list-style-type: none"> • Actualizable a 2 procesadores (total de 4, 8, 12 o 16 núcleos). • Hasta 24 ranuras DIMM disponibles para una mayor capacidad de memoria. • Hasta 24 bahías de disco duro SFF; o 18 bahías de unidades LFF. • 8 ranuras de E / S PCIe 3.0 y 1 PCIe 2.0 para expansión adicional de comunicaciones y almacenamiento. • Fuente de alimentación redundante y ventiladores redundantes. • Bahía de unidad óptica (hasta 3).

On-system management chipset	HP iLO (Firmware: HP iLO 4)							
Protección de Memoria	<ul style="list-style-type: none"> • ECC avanzado • Repuesto en línea • Paso de bloqueo 							
Memoria	<table border="1"> <tr> <td>Tipo</td> <td>HP SmartMemory. DDR3 Load Reduced (LRDIMM), Registered.</td> </tr> <tr> <td>Standard (Performance Models)</td> <td>32GB (4 x 8GB) PC3L-10600R (DDR3-1333) Registered DIMMS</td> </tr> </table>		Tipo	HP SmartMemory. DDR3 Load Reduced (LRDIMM), Registered.	Standard (Performance Models)	32GB (4 x 8GB) PC3L-10600R (DDR3-1333) Registered DIMMS		
Tipo	HP SmartMemory. DDR3 Load Reduced (LRDIMM), Registered.							
Standard (Performance Models)	32GB (4 x 8GB) PC3L-10600R (DDR3-1333) Registered DIMMS							
Control de red.	HP 1Gb Ethernet 4-port 331i adapter							
Control de almacenamiento	<table border="1"> <tr> <td>Entry Models</td> <td>HP Smart Array P420i/Zero Memory Controller</td> </tr> <tr> <td>Base Models</td> <td>HP Smart Array P420i/512M FBWC Controller</td> </tr> <tr> <td>Performance Models</td> <td>HP Smart Array P420i/2G FBWC Controller</td> </tr> </table>		Entry Models	HP Smart Array P420i/Zero Memory Controller	Base Models	HP Smart Array P420i/512M FBWC Controller	Performance Models	HP Smart Array P420i/2G FBWC Controller
Entry Models	HP Smart Array P420i/Zero Memory Controller							
Base Models	HP Smart Array P420i/512M FBWC Controller							
Performance Models	HP Smart Array P420i/2G FBWC Controller							
Dispositivos de almacenamiento interno	<ul style="list-style-type: none"> • 600 Gb SAS (RAID 10 4HDD). • 4 TB SATA(RAID 1 2 HDD). 							
Interfaces	<ul style="list-style-type: none"> • Serial: 1 • Video: 1 (rear) • Network RJ-45 (Ethernet): 5 (1 dedicado para HP Integrated Lights-Out) • iLO remote management network port: 1Gb dedicado 							

	<ul style="list-style-type: none"> • SD slot: 1 (internal) • USB:10 (4 front, 4 rear, 2 internal)
Cumplimiento estándar de la industria	<ul style="list-style-type: none"> • ACPI 2.0b Compliant • PCIe 3.0 Compliant • PXE Support • WOL Support • Novell certified • IPMI 2.0, SMASH CLP, DCMI 1.0 compliant • Microsoft logo certifications • USB 2.0 compliant • SMBIOS 2.7.1
Gráficos	<p>Integrated Matrox G200 video standard</p> <ul style="list-style-type: none"> • 1280 x 1024 (32 b/p) • 1920 x 1200 (16 b/p) <p>HP iLO 4 On System Management Memory</p> <ul style="list-style-type: none"> • 16MB Flash • 256MB DDR 3 with ECC (112MB after ECC and video)
Ventiladores del sistema	<ul style="list-style-type: none"> • Cuatro conectadores en caliente, ventiladores redundantes más rejillas.
Form factor	Tower or Rack (5U)

Para iniciar la configuración del servidor donde se alojará la aplicación, primero se debe agregar los roles al servidor, esto se hace desde el apartado de *Personalizar este servidor*.



Figura 4.27. Agregar roles al servidor.

Posteriormente se abrirá una ventana donde se seleccionará el rol que se desea agregar, como se muestra en la Figura 4.28. Se debe seleccionar el rol *Servidor web (IIS)*, y dar clic en *Siguiente*.

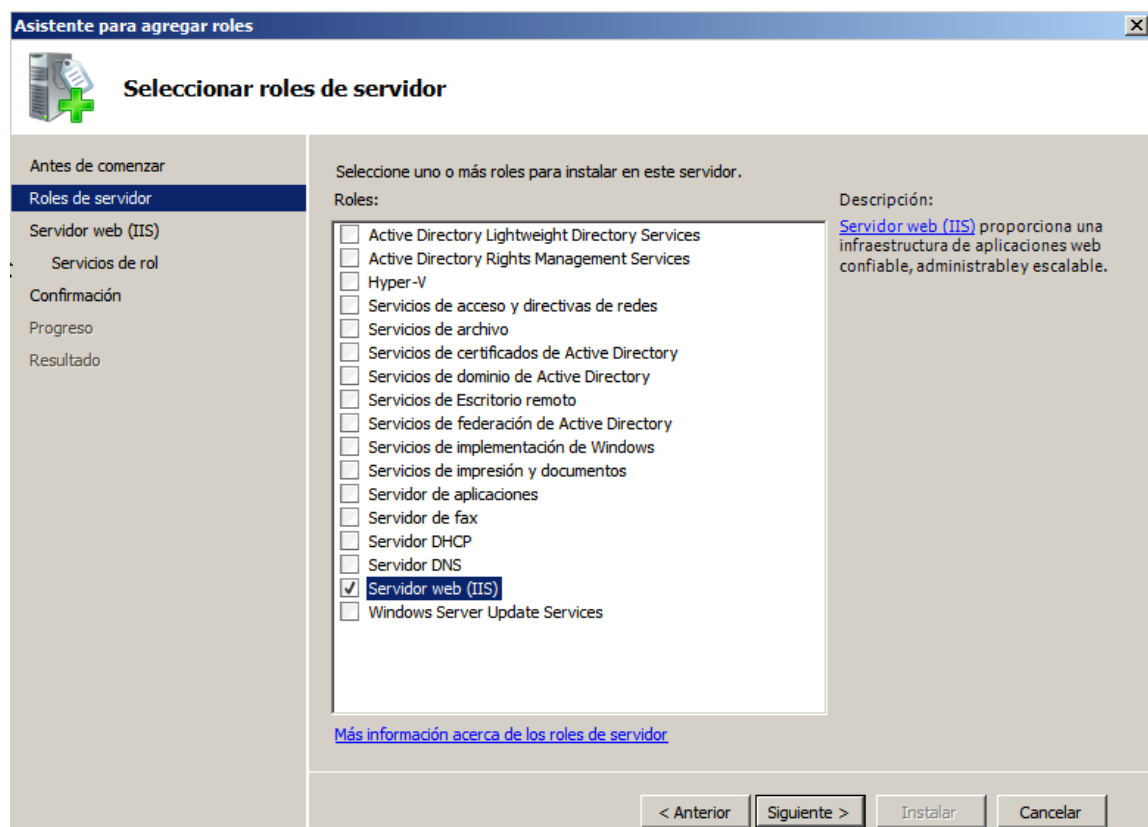


Figura 4.28. Selección del rol Servidor web(IIS).

En la Figura 4.29 se muestran todos los servicios de rol que el *Servidor web(IIS)* ofrece. La herramienta para manipular imágenes médicas fue desarrollada en el *framework* ASP.NET Core de Visual Studio 2016, por lo tanto, se debe seleccionar el servicio de rol *ASP.NET* y dar clic en *Siguiente*.

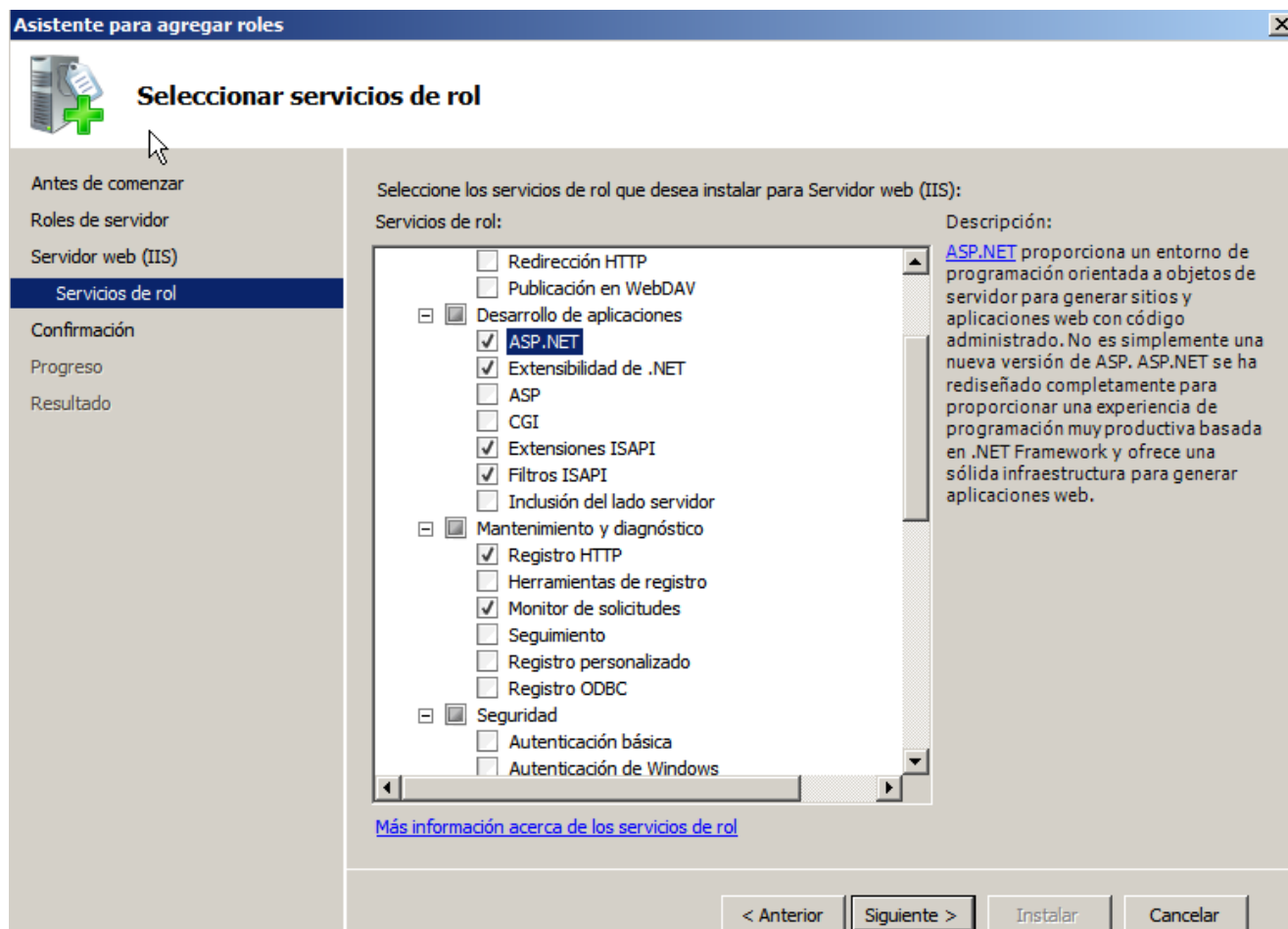


Figura 4.29. Selección del servicio de rol ASP.NET.

Una vez que se haya seleccionado el servicio de rol, se abrirá una nueva ventana, con la cual mostrará una breve información de todos los servicios que se instalarán, tal como se visualiza en la Figura 4.30.

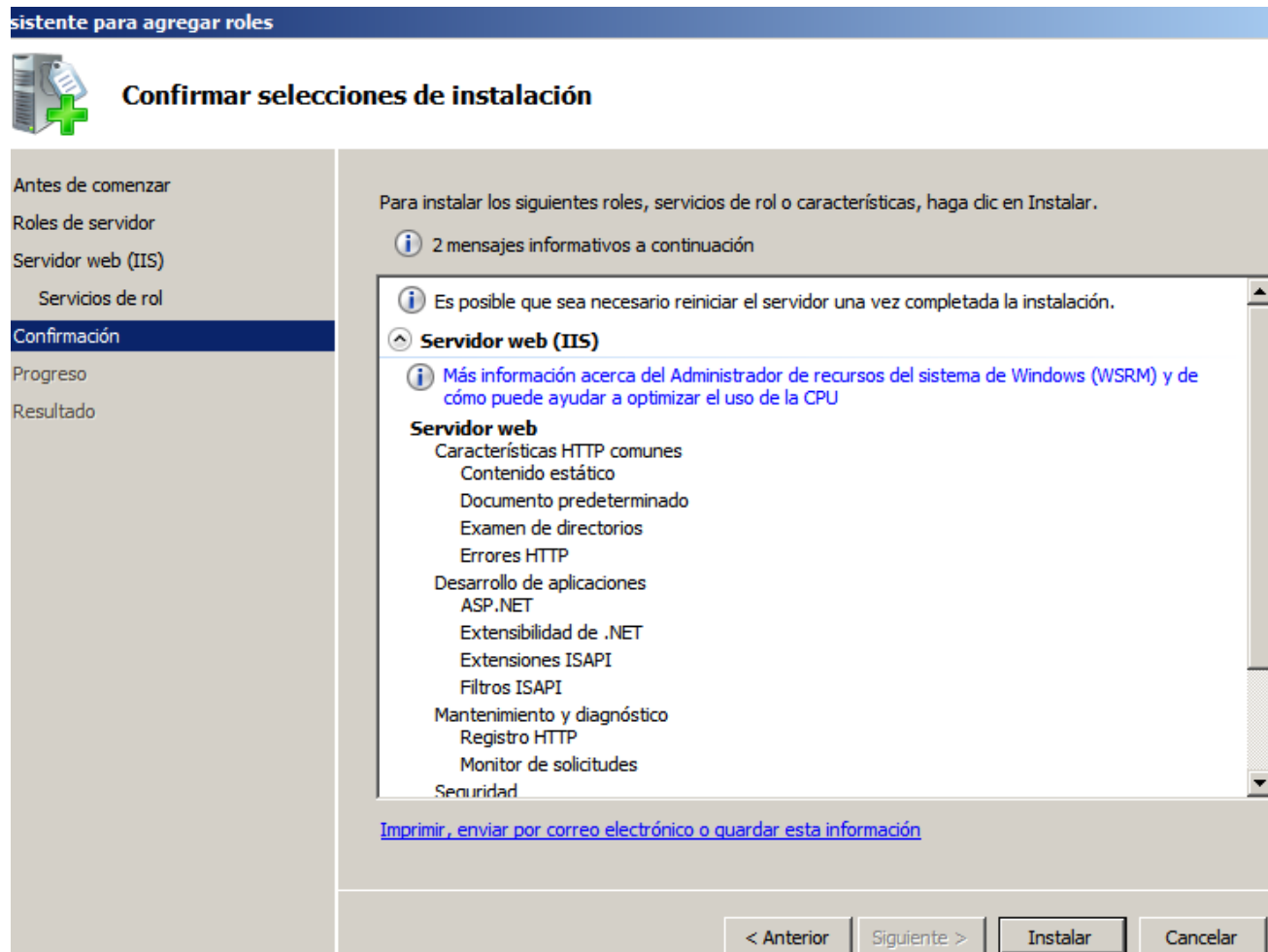


Figura 4.30. Servicios del rol Servidor web(IIS) que se instalarán.

Una vez que se haya instalado el servicio de Servidor web, se debe reiniciar el servidor para evitar que haya problemas, para confirmar que la instalación ha sido correcta, se debe abrir el navegador internet Explorer que se encuentra instalado en el servidor y posteriormente dirigirse a la dirección <http://localhost/>, se visualizará la página de bienvenida de ISS7, tal como se muestra en la Figura 4.31.



Figura 4.31. Confirmación que el Servidor web (IIS) se haya instalado.

Para terminar con la configuración del servidor se debe de instalar los programas *ASP.NET Core/.NET Core: Runtime & Hosting Bundle* y *ASP.NET Core Installer*, para poder ejecutar la aplicación. Para saber la versión que se debe instalar, se hace mediante la ejecución del comando `dotnet --version` desde el cmd, como se ilustra en la Figura 4.32.

```

C:\Windows\System32\cmd.exe

Run 'dotnet [command] --help' for more information on a command.
C:\Herramienta\Herramienta>dotnet --version
2.1.503
C:\Herramienta\Herramienta>

```

Figura 4.32. Versión de ASP.NET Core donde se desarrolló la aplicación.

Después de saber la versión del ASP.NET Core, se descargan los programas desde la página oficial de Microsoft, como se visualiza en la Figura 4.33.

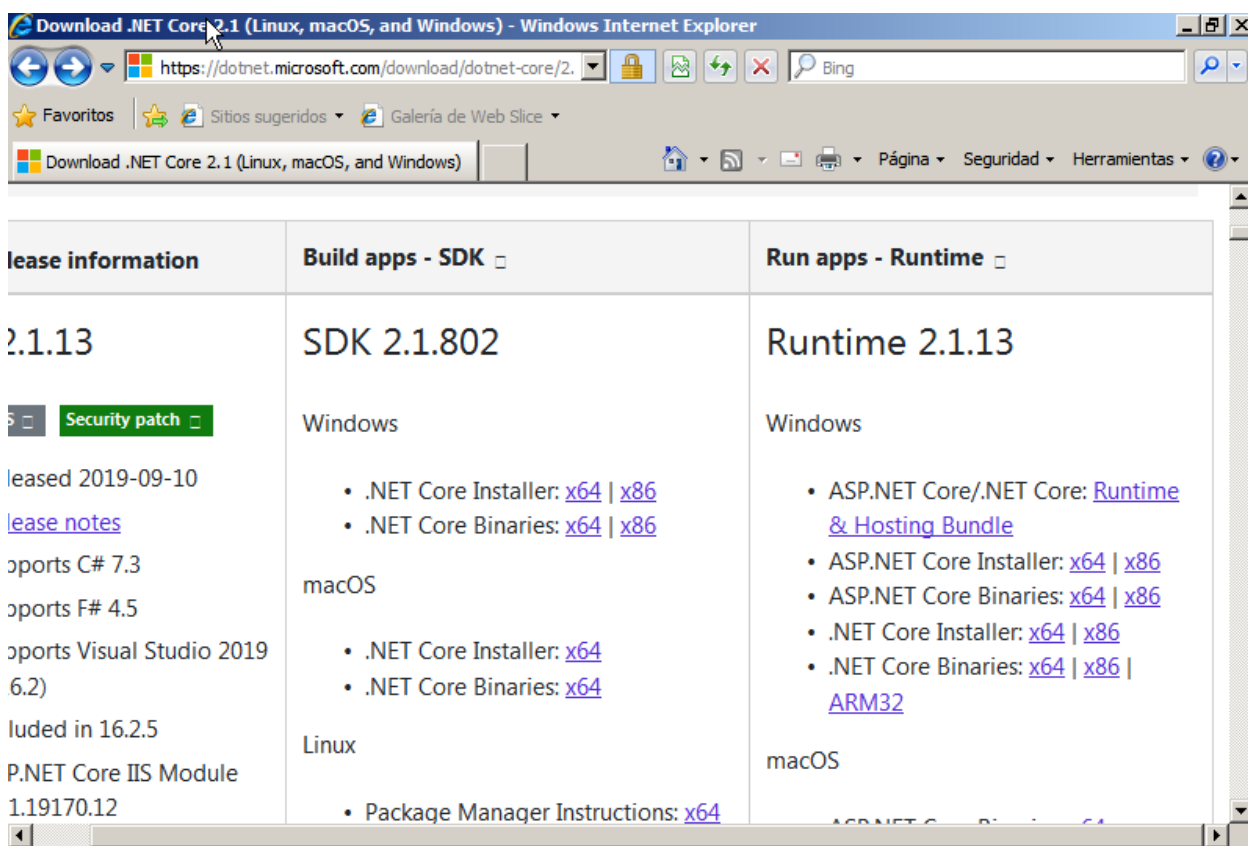
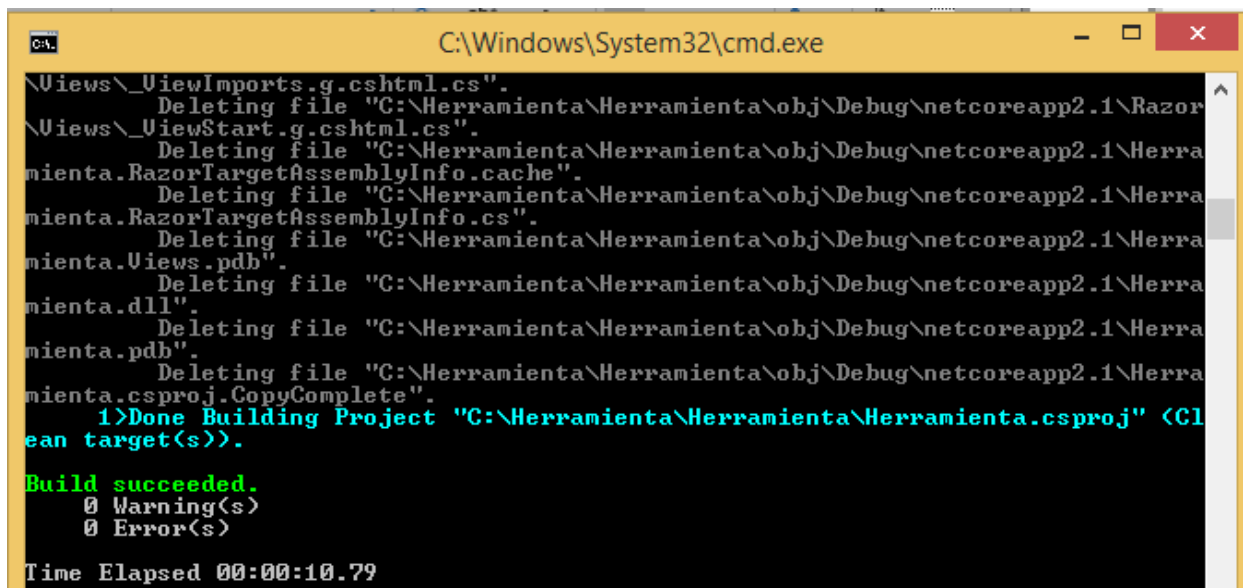


Figura 4.33. Descargar las aplicaciones Runtime para ejecutar la aplicación.

4.4.2 Generación de bibliotecas de enlace dinámico(DLL) para producción

Para la generación de los DLL, primero se debe localizar la carpeta donde está almacenado el proyecto, en este caso el proyecto está en la dirección C:\Herramienta\Herramienta.

Para empezar a generar los archivos DLL se debe ejecutar el comando *dotnet clean* desde el cmd, para limpiar el proyecto de posibles dll que estén en conflictos, es decir dejar limpia la carpeta binaria.



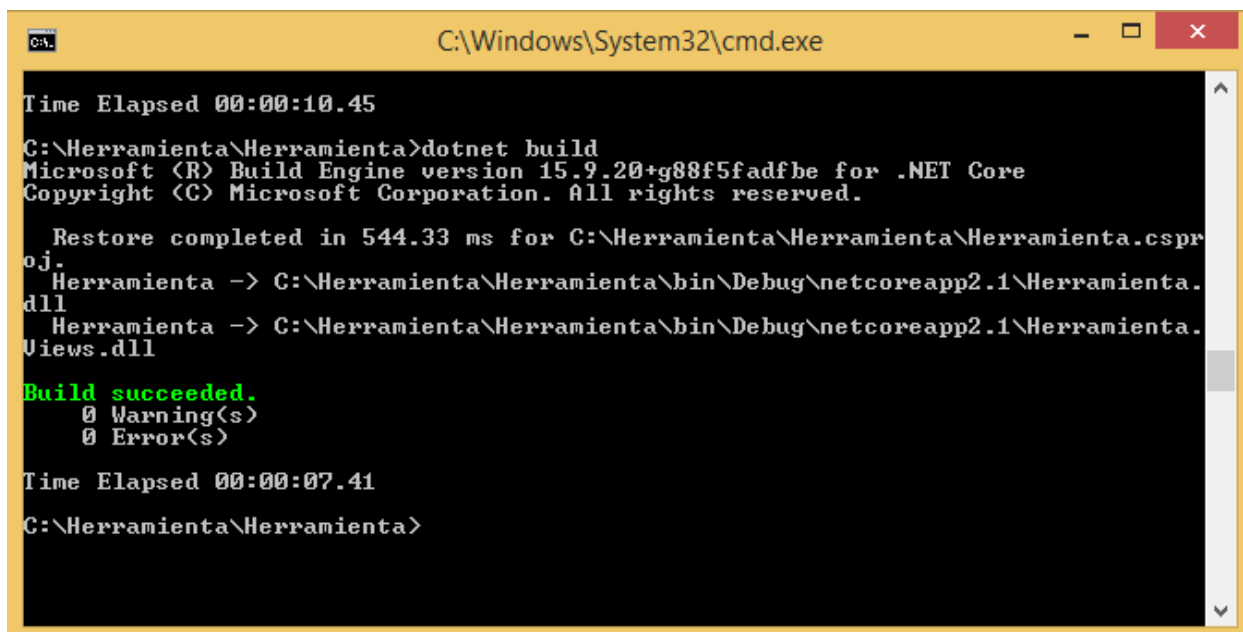
```

C:\Windows\System32\cmd.exe
\Views\_ViewImports.g.cshtml.cs".
  Deleting file "C:\Herramienta\Herramienta\obj\Debug\netcoreapp2.1\Razor
\Views\_ViewStart.g.cshtml.cs".
  Deleting file "C:\Herramienta\Herramienta\obj\Debug\netcoreapp2.1\Herra
mienta.RazorTargetAssemblyInfo.cache".
  Deleting file "C:\Herramienta\Herramienta\obj\Debug\netcoreapp2.1\Herra
mienta.RazorTargetAssemblyInfo.cs".
  Deleting file "C:\Herramienta\Herramienta\obj\Debug\netcoreapp2.1\Herra
mienta.Views.pdb".
  Deleting file "C:\Herramienta\Herramienta\obj\Debug\netcoreapp2.1\Herra
mienta.dll".
  Deleting file "C:\Herramienta\Herramienta\obj\Debug\netcoreapp2.1\Herra
mienta.pdb".
  Deleting file "C:\Herramienta\Herramienta\obj\Debug\netcoreapp2.1\Herra
mienta.csproj.CopyComplete".
1>Done Building Project "C:\Herramienta\Herramienta\Herramienta.csproj" (Cl
ean target(s)).
Build succeeded.
    0 Warning(s)
    0 Error(s)
Time Elapsed 00:00:10.79

```

Figura 4.34. Limpieza del proyecto.

Después de limpiar el proyecto, se debe compilar el proyecto, para realizarlo se hace con el comando *dotnet build*.



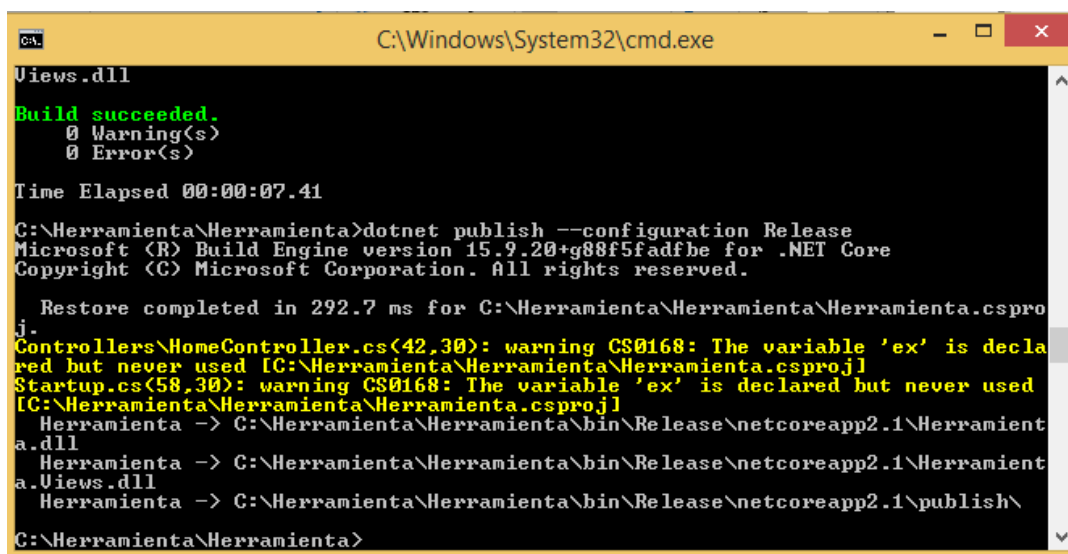
```

C:\Windows\System32\cmd.exe
Time Elapsed 00:00:10.45
C:\Herramienta\Herramienta>dotnet build
Microsoft (R) Build Engine version 15.9.20+g88f5fadbfe for .NET Core
Copyright (C) Microsoft Corporation. All rights reserved.
  Restore completed in 544.33 ms for C:\Herramienta\Herramienta\Herramienta.csproj.
  Herramienta -> C:\Herramienta\Herramienta\bin\Debug\netcoreapp2.1\Herramienta.dll
  Herramienta -> C:\Herramienta\Herramienta\bin\Debug\netcoreapp2.1\Herramienta.Views.dll
Build succeeded.
    0 Warning(s)
    0 Error(s)
Time Elapsed 00:00:07.41
C:\Herramienta\Herramienta>

```

Figura 4.35. Compilación del proyecto.

Para generar los DLL con la configuración de la versión de lanzamiento o de producción, se hace con el comando `dotnet publish - -configuration Release`.



```

C:\Windows\System32\cmd.exe
Views.dll
Build succeeded.
    0 Warning(s)
    0 Error(s)

Time Elapsed 00:00:07.41

C:\Herramienta\Herramienta>dotnet publish --configuration Release
Microsoft (R) Build Engine version 15.9.20+g88f5fadfbc for .NET Core
Copyright (C) Microsoft Corporation. All rights reserved.

Restore completed in 292.7 ms for C:\Herramienta\Herramienta\Herramienta.csproj.
Controllers\HomeController.cs(42,30): warning CS0168: The variable 'ex' is declared but never used [C:\Herramienta\Herramienta\Herramienta.csproj]
Startup.cs(58,30): warning CS0168: The variable 'ex' is declared but never used [C:\Herramienta\Herramienta\Herramienta.csproj]
Herramienta -> C:\Herramienta\Herramienta\bin\Release\netcoreapp2.1\Herramienta.dll
Herramienta -> C:\Herramienta\Herramienta\bin\Release\netcoreapp2.1\Herramienta.Views.dll
Herramienta -> C:\Herramienta\Herramienta\bin\Release\netcoreapp2.1\publish\
C:\Herramienta\Herramienta>

```

Figura 4.36. Generación de DLL.

Los archivos DLL generados se encuentran en la carpeta con dirección `C:\Herramienta\Herramienta\bin\Release\netcoreapp2.1\publish`, todos los archivos que se generaron se deben copiar y pegar en la carpeta `wwwroot` del servidor. Como se puede mostrar en la Figura 4.37 y 4.38.

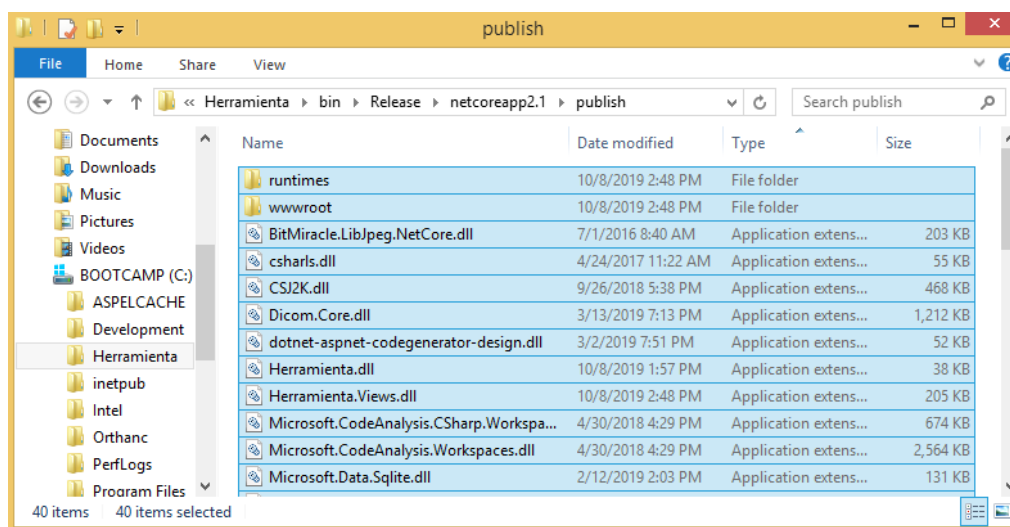


Figura 4.37. Dirección donde se generan los DLL.

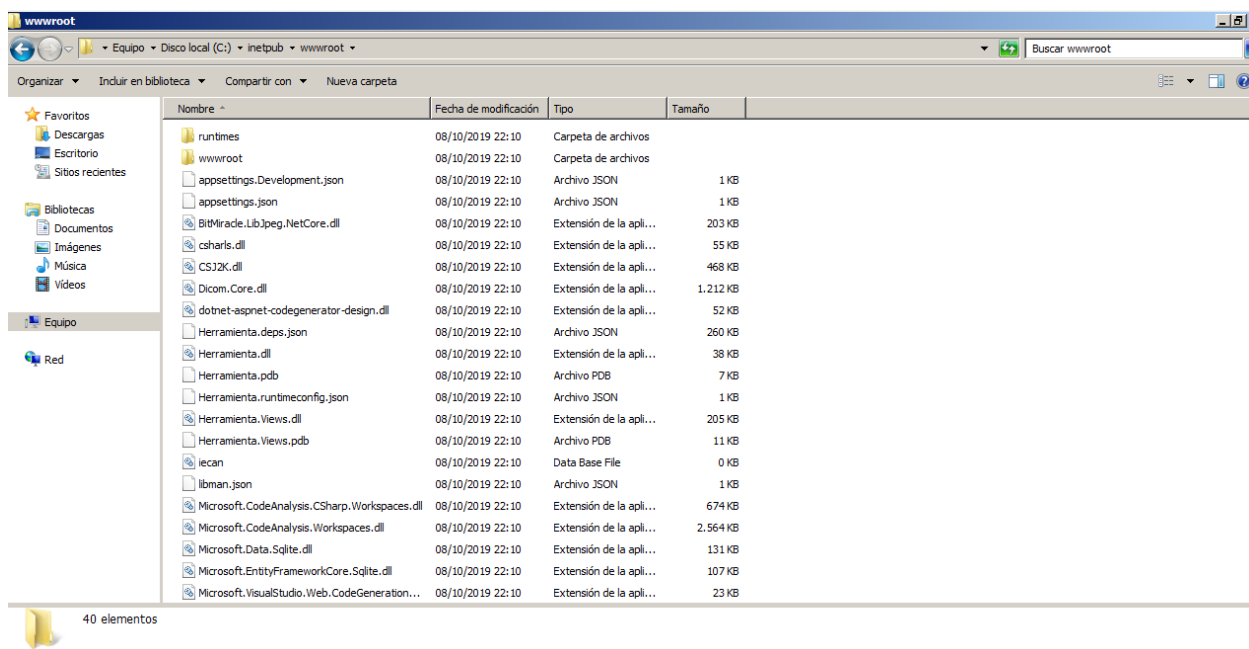


Figura 4.38. Almacenando los archivos DLL en la carpeta wwwroot del servidor.

Por último accedemos a la herramienta web con la siguiente dirección 10.0.2.15, para confirmar que la aplicación se haya alojado correctamente.



Figura 4.39. Herramienta web funcionando.

Capítulo 5. Pruebas y resultados

En este capítulo se muestran los resultados obtenidos al utilizar el software desarrollado con este proyecto. Las pruebas realizadas se hicieron en el Instituto Estatal de Cancerología, en el departamento de Informática, dichas pruebas se llevaron a cabo en los navegadores Google Chrome, Firefox, y Edge Explorer, debido a que son los navegadores más actuales.

5.1 Pruebas de conectividad

Esta prueba consiste en establecer una conexión entre la herramienta web y el servidor de imágenes DCM4CHEE. Para poder realizar la conexión se tiene que tener en cuenta ciertos aspectos:

- dirección IP o nombre del host.
- Puerto en el cual el servidor DCM4CHEE está escuchando.
- Nombre del Servidor al cual estas llamando.
- Nombre del Servidor que ha sido llamado.

Para seleccionar estos aspectos de conectividad se utiliza la integración del estándar DICOM con estándares HL7 (Health Level Seven). El HL7 proporcionan un marco (y estándares relacionados) para el intercambio, integración, intercambio y recuperación de información electrónica de salud. Estos estándares definen cómo se empaqueta y comunica la información de una parte a otra, estableciendo el idioma, la estructura y los tipos de datos necesarios para una integración perfecta entre sistemas (HL7 International, 2019).

Teniendo en cuenta los aspectos mencionado para establecer la conexión se tienen que realizar los siguientes pasos, primero se tiene que ir a la pestaña de Configuración y posteriormente se

tiene que ingresar los parámetros correctos para que la conexión sea exitosa. En la Figura 5.1 se visualiza como se deben ingresar los parámetros.

IECAN Home Pacientes Herramienta Configuración

Configuración de Conexión

Dirección ip o nombre de host
192.168.5.137

Puerto
11112

Calling AET
DCM4CHEE

Called AET
DCM4CHEE

Guardar

Herramienta de apoyo para manipular imágenes médicas
Diseño y desarrollo por el estudiante de maestría en Sistemas Computacionales,
Ing. Manuel Niño Navarrete.
© Copyright 2019

Figura 5.1. Configuración de conexión al servidor.

Los resultados obtenidos de la conexión se visualizan en la pestaña *Pacientes* de la herramienta web, muestra una lista con todas las imágenes de los pacientes almacenados en el servidor, como se observa en la Figura 5.2, también tiene un motor de búsquedas para realizar consultas más específicas sobre algún paciente, estas búsquedas se pueden realizar por medio de id del paciente, nombre y/o apellidos del paciente.

NOMBRE PACIENTE	MODALIDAD ESTUDIO	FECHA ESTUDIO	ACCIONES
EDWIN GAEL^ANGEL VICTORIANO^^^	CR	26/12/2018	Procesar
EDWIN GAEL^ANGEL VICTORIANO^^^	CR	07/03/2019	Procesar
AZUCENA^GALEANA GONZALEZ^^^	CR	26/12/2018	Procesar
IVAN^CELIS ALTAMIRANO^^^	CR	26/12/2018	Procesar
MEZA HEZQUIO^ELENA	CT\SR	28/12/2018	Procesar
YSABEL^DOMINGUEZ VENTURA^^^	CR	03/01/2019	Procesar
VICENTA^VARGAS RAMIREZ^^^	CR	03/01/2019	Procesar
KENIA YAZMIN^BAUTISTA GONZALEZ^^^	CR	03/01/2019	Procesar
KENIA YAZMIN^BAUTISTA GONZALEZ^^^	CR	03/01/2019	Procesar
TAPIA SERNA^MARIO^^^	CR	04/03/2019	Procesar
INES^ROSALES ROMAN^^^	CR	03/01/2019	Procesar
PALACIOS ANTUNEZ^ELVIRA^^^	CR	03/01/2019	Procesar
PALACIOS ANTUNEZ^ELVIRA^^^	CR	03/01/2019	Procesar
PALACIOS ANTUNEZ^ELVIRA^^^	CT	03/01/2019	Procesar
PALACIOS ANTUNEZ^ELVIRA^^^	CR	21/03/2019	Procesar
PALACIOS ANTUNEZ^ELVIRA^^^	CR	12/04/2019	Procesar
PALACIOS ANTUNEZ^ELVIRA^^^	CR	28/05/2019	Procesar

Figura 5.2. Lista de imágenes de pacientes almacenadas en el servidor.

5.2 Pruebas de usabilidad

Para esta prueba se utilizó la Tabla 5.1, para evaluar 15 criterios que debe cumplir la herramienta web, estos criterios se seleccionaron de acuerdo a la norma ISO 9241-11 de la usabilidad de software.

Tabla 5.1. Criterios de usabilidad de la herramienta web.

Características	Definición
previsibilidad	La interfaz de usuario debe producir resultados que estén de acuerdo a los requerimientos especificados.
Capacidad de aprendizaje	La interfaz de usuario debe estar diseñada para que el usuario aprenda fácilmente el uso de la herramienta web.

Estructura	La interfaz de usuario debe estar organizada con un propósito, de manera significativa y útil que junte cosas relacionadas y separe cosas no relacionadas basadas en modelos claros y consistentes que sean aparentes y reconocibles para otros.
Consistencia	La interfaz de usuario debe estar diseñada de manera consistente.
Memorabilidad	La interfaz de usuario debe ser fácil para que los usuarios recuerden cómo usar la herramienta web.
Familiaridad	La interfaz de usuario debe ser familiar para los usuarios.
Esfuerzo	La interfaz de usuario debe estar diseñada para minimizar el esfuerzo del usuario por usar el sistema.
Visibilidad	La interfaz de usuario siempre debe mantener a los usuarios informados sobre lo que está sucediendo, a través de comentarios apropiados dentro de un tiempo razonable.
Sencillez	Simplifica las tareas comunes y sencillas, se comunica en el idioma del usuario.
Eficiencia	La herramienta web debe ser eficiente de usar para que una vez que el usuario haya aprendido el sistema, sea posible un alto nivel de productividad.
Realimentación	La interfaz de usuario debe mantener a los usuarios informados sobre acciones o interpretaciones, cambios de estado o condición utilizando un lenguaje claro, conciso y sin ambigüedades familiar para los usuarios.
Indicación de errores	La representación de errores debe ser clara para los usuarios.
Control de usuario	Los usuarios deben poder controlar la herramienta web por sus propias decisiones.
responsiva	La herramienta web debe ser capaz de adaptarse a diferentes tamaños de pantalla.
Flexibilidad	La interfaz de usuario debe ser flexible para que se adapte a diversos entornos y usuarios.

Para determinar la usabilidad de la herramienta web se realizó una prueba a 12 usuarios del Instituto Estatal de Cancerología, donde los usuarios utilizaron la herramienta web y se evaluó de acuerdo a los criterios de la tabla anterior. Los resultados obtenidos se pueden observar en la Tabla 5.2, los valores van del 0 al 2, donde 2 indica “estoy de acuerdo”, 1 indica “parcialmente de acuerdo” y 0 indica “no estoy de acuerdo”.

Tabla 5.2. Tabla de resultados de la evaluación.

	Personal	Previsibilidad	Capacidad de aprendizaje	Estructura	Consistencia	Memorabilidad	Familiaridad	Esfuerzo	Visibilidad	Sencillez	Eficiencia	realimentación	Indicación de Error	Control de usuario	Responsiva	Flexibilidad
Per1	2	1	1	1	2	2	2	1	2	2	1	1	0	2	1	
per2	2	2	2	2	2	2	1	0	2	2	2	1	1	2	2	
per3	2	1	2	2	1	2	1	2	2	1	1	2	1	2	2	
per4	2	2	2	2	1	2	1	2	2	2	1	2	2	2	2	
per5	2	2	1	1	2	2	2	2	1	1	1	1	2	2	2	
per6	1	2	1	1	2	1	2	2	1	1	2	2	1	2	1	
per7	2	2	1	2	2	1	2	2	1	2	1	2	2	2	1	
per8	2	2	1	1	1	1	1	2	1	1	2	1	1	2	2	
per9	2	2	2	1	1	1	1	1	1	1	1	1	1	2	2	
per10	2	1	1	2	1	1	2	1	1	1	2	2	2	2	2	
per11	1	1	2	1	2	2	2	1	1	2	1	1	2	1	2	
per12	2	1	1	0	2	2	0	1	2	2	1	0	2	2	2	

Los resultados obtenidos de la evaluación de la prueba de usabilidad se traducen en gráficas circulares, tal como se ilustra en la Figura 5.3.

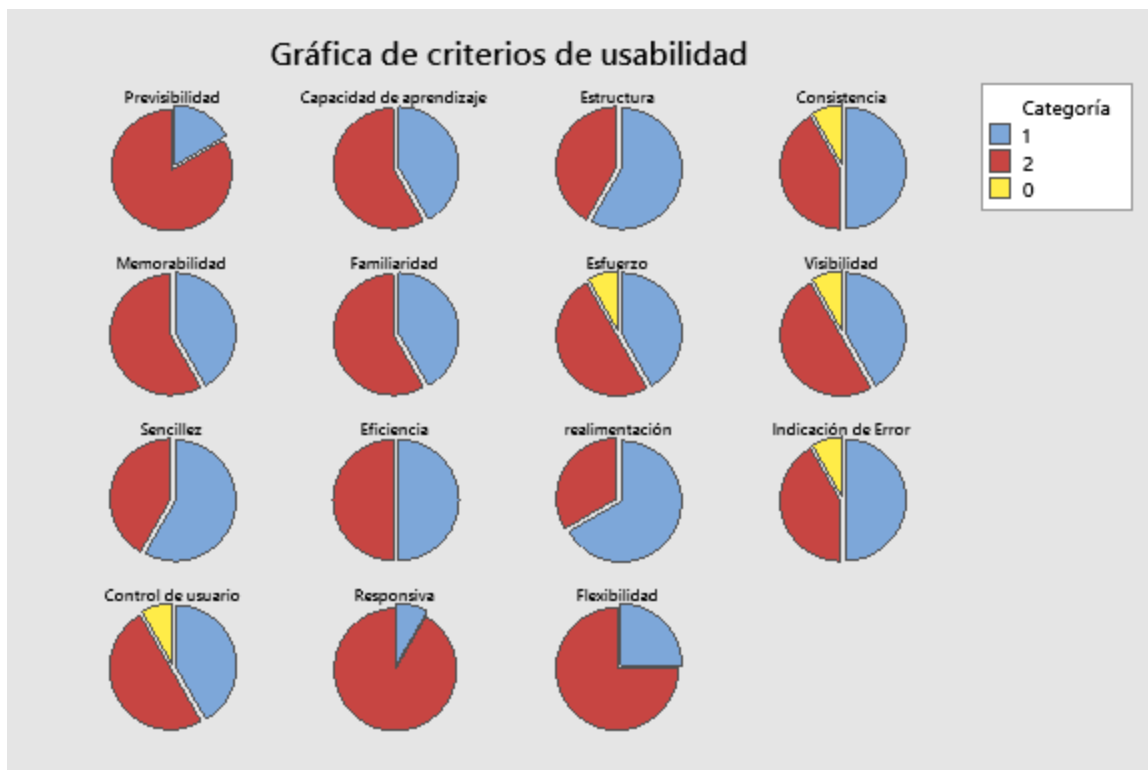


Figura 5.3. Gráficas de resultados de la evaluación.

Los resultados obtenidos de cada criterio de acuerdo a las gráficas fueron los siguientes:

- ***Previsibilidad*** – El 83.3% de los usuarios votaron “estoy de acuerdo” y el 16.7% votaron “parcialmente de acuerdo”.
- ***Capacidad de aprendizaje*** – El 58.3% de los usuarios votaron “estoy de acuerdo” y un 41.7% votaron “parcialmente de acuerdo”.
- ***Estructura*** – El 58.3% de los usuarios votaron “parcialmente de acuerdo” y un 41.7% votaron “estoy de acuerdo”.
- ***Consistencia*** – El 50% de los usuarios votaron “parcialmente de acuerdo”, el 41.7% votaron “estoy de acuerdo” y el 8.3% votaron “no estoy de acuerdo”.
- ***Memorabilidad*** – El 58.3 de los usuarios votaron “estoy de acuerdo” y el 41.7% votaron “parcialmente de acuerdo”.

- **Familiaridad** – El 58.3% de los usuarios votaron “estoy de acuerdo” y solo el 41.7% votaron “parcialmente de acuerdo”.
- **Esfuerzo** – El 50% de los usuarios votaron “estoy de acuerdo”, el 41.7% votaron “parcialmente de acuerdo” y tan solo el 8.3% votaron “no estoy de acuerdo”.
- **Visibilidad** – El 50% de los usuarios votaron “estoy de acuerdo”, el 41.7% votaron “parcialmente de acuerdo” y tan solo el 8.3% votaron “no estoy de acuerdo”.
- **Sencillez** – El 58.3% de los usuarios votaron “parcialmente de acuerdo” y un 41.7% votaron “estoy de acuerdo”.
- **Eficiencia** – El 50% de los usuarios votaron “estoy de acuerdo” y el 50% votaron “parcialmente de acuerdo”.
- **Realimentación** – El 66.7% de los usuarios votaron “parcialmente de acuerdo” y un 33.3% votaron “estoy de acuerdo”.
- **Indicación de error** – El 50% de los usuarios votaron “parcialmente de acuerdo”, el 41.7% votaron “estoy de acuerdo” y tan solo el 8.3% votaron “no estoy de acuerdo”.
- **Control de usuario** – El 50% de los usuarios votaron “estoy de acuerdo”, el 41.7% votaron “parcialmente de acuerdo” y tan solo el 8.3% votaron “no estoy de acuerdo”.
- **Responsiva** – El 91.7% de los usuarios votaron “estoy de acuerdo” y tan solo el 8.3% votaron “parcialmente de acuerdo”.
- **Flexibilidad** – El 75% de los usuarios votaron “estoy de acuerdo” y tan solo el 15% votaron “parcialmente de acuerdo”.

5.3 Pruebas de funcionalidad

Se realizaron pruebas de visualización y manipulación con las diferentes funcionalidades que tiene la herramienta web. Para ello se utilizó el estándar ISO/IEC 25000:2014 SQuaRE (System and Software Quality Requirements and Evaluation), se refiere a la capacidad del producto de software para suministrar un conjunto de funciones que satisfagan las necesidades implícitas o explícitas de los usuarios, al ser utilizado bajo condiciones específicas.

Se clasifica en cinco subcategorías que abarcan los aspectos relevantes en relación con la funcionalidad de un producto software.

- **Apropiabilidad:** Es la capacidad del software para suministrar un conjunto apropiado de funciones que cumplan tareas específicas y objetivos del usuario.
- **Exactitud:** Capacidad del software para proveer los resultados correctos y los efectos pactados, con un adecuado grado de precisión.
- **Interoperabilidad:** Capacidad del software para interactuar con uno o más sistemas específicos. Esto depende, en gran parte, de la necesidad de interacción con los demás sistemas.
- **Seguridad:** Hace referencia a la capacidad del software para proteger los datos y la información, con el fin de que personas no autorizadas nunca puedan ingresar al sistema, leer o modificar los datos.
- **Conformidad** en la funcionalidad: Capacidad del software para ajustarse a los estándares, convenciones y regulaciones relacionadas con la correcta implementación de especificaciones durante todo el proceso de análisis y construcción.

Para hacer pruebas de visualización de imágenes se debe seleccionar las imágenes desde la lista de pacientes, las imágenes se abrirán en la pestaña de *Herramienta*, dentro del contenedor de imagen en la esquina superior izquierda se puede visualizar el nombre del paciente, edad y número de identificación del paciente, en la esquina superior derecha se mostrará el nombre del hospital donde se realizó el estudio, fecha del estudio y la parte examinada del paciente, como se muestra en la Figura 5.4.

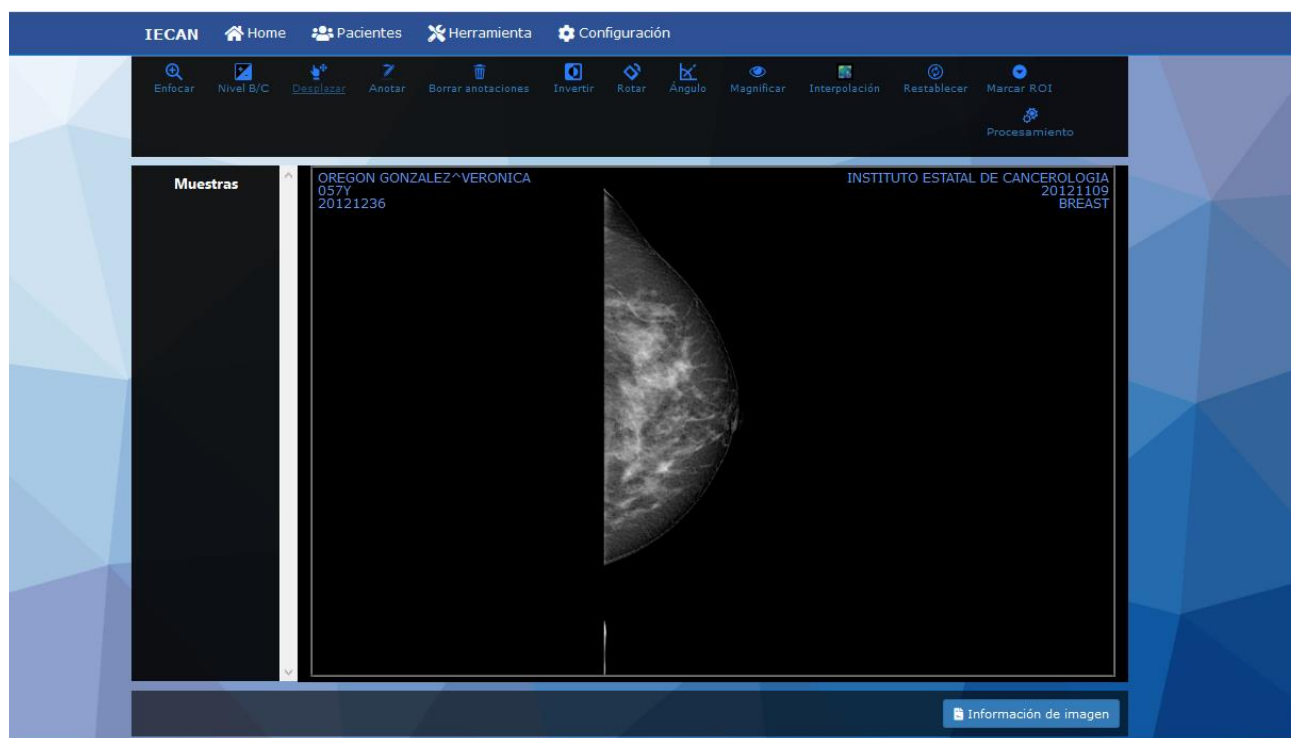


Figura 5.4. Imagen médica visualizada en la herramienta.

En la esquina inferior derecha de la pestaña *Herramienta* hay un botón con nombre *Información de imagen*, este botón muestra información adicional de la imagen.

Características de la imagen		
SOP Instance UID: 2.16.840.1.113669.632.25.1.201130.20121109092224.3		
SOP Class UID: 1.2.840.10008.5.1.4.1.1.1.2 [Digital Mammography X-Ray Image Storage - For Presentation]		
Transfer Syntax: 1.2.840.10008.1.2.1 [Explicit VR Little Endian]		Series Date: 20121109
Acquisition Date: 20121109	Study ID: 421	Study Time: 092058
Series Time: 092219	Modality: MG	Presentation Intent Type: FOR PRESENTATION
Institution Address: Av RUIZ CORTINES 128-A COL. ALTA PROGRESO		Station Name: GIOTTO IMAGE 3DL
Study Description: DIAGNOSTIC	Operators Name: TR. MIRNA	Manufacturer Model Name: GIOTTO IMAGE 3DL
Patient Birth Date: 19550428	Patient Sex: F	Protocol Name: Mammography
Organ Dose: 0.019	Organ Exposed: BREAST	Samples Per Pixel: 1
Photometric Interpretation: MONOCHROME2	Number Of Frames:	Planar Configuration:
Rows: 3584	Columns: 2816	Pixel Spacing:
Bits Allocated: 16	Bits Stored: 14	High Bit: 13
Pixel Representation: 0 (unsigned)	Window Center: 8623\8623 \8590\8706\8623	Window Width: 1652\1652 \1486\1817\1982
Rescale Intercept: 0	Rescale Slope: 1	Basic Offset Table Entries:
Fragments:	Min Stored Pixel Value: 1	Max Stored Pixel Value: 9499
Total Time: 2879ms	Load Time: 261ms	Decode Time: 0ms

Figura 5.5. Información de una imagen médica.

La información mostrada se seleccionó de acuerdo al estándar DICOM, se extrajo la información más relevante y apropiada para el especialista.

La herramienta cuenta con una serie de funcionalidades, estas funcionalidades nos ayudan a manipular las imágenes médicas.

Funcionalidad *Enfocar* (Figura 5.6), permite enfocar y desenfocar la imagen que se está analizando. Para esta prueba se utilizó una imagen de mamografía.

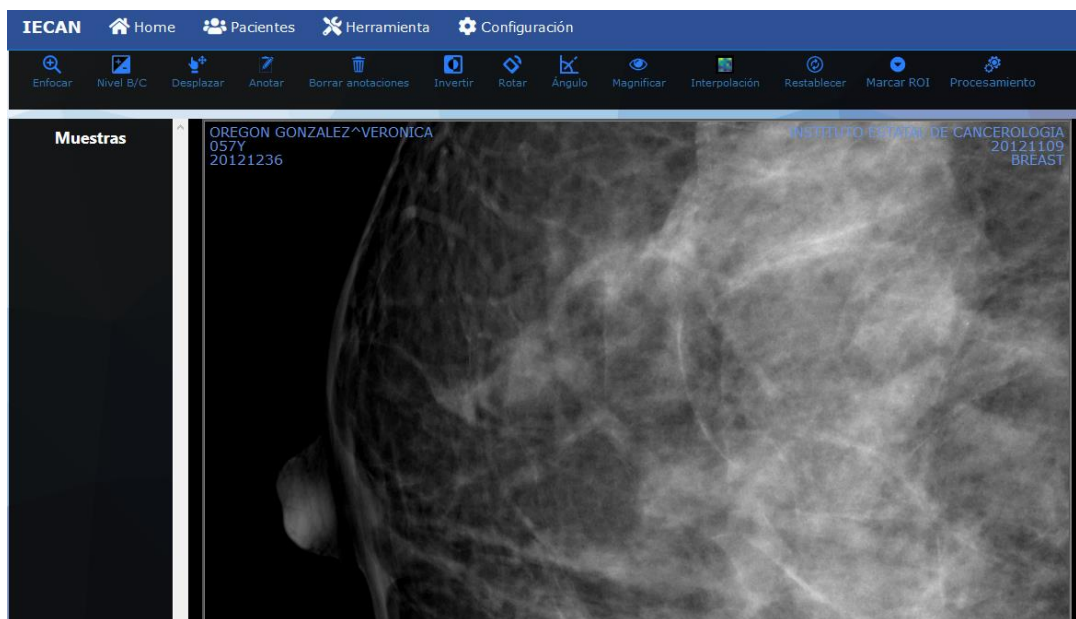


Figura 5.6. Funcionalidad Enfocar de la herramienta.

En los resultados obtenidos la imagen puede incrementar hasta cuatro veces su tamaño original, de la misma manera se puede desenfocar hasta un tamaño menor a la original, esto se hace con el fin de poder tener una mejor visibilidad de la imagen que se está analizando, la reacción de esta funcionalidad en la imagen es al instante, pues no requiere de un alto proceso de la computadora.

Unos de los problemas que se tienen al adquirir las imágenes médicas de los dispositivos de adquisición es que puedan salir con mala iluminación o bajo contraste. La funcionalidad de *Nivel B/C* permite ajustar el brillo y contraste de la imagen, mejorando así la calidad de la misma para una mejor interpretación por parte de los especialistas. Para utilizar esta función se hace a través del mouse, manteniendo oprimido el botón izquierdo y moviéndolo de arriba hacia abajo se ajusta el contraste y moviéndolo de derecha a izquierda se ajusta el brillo. En la Figura 5.7 se muestra una imagen con el nivel del brillo y contraste ajustado.

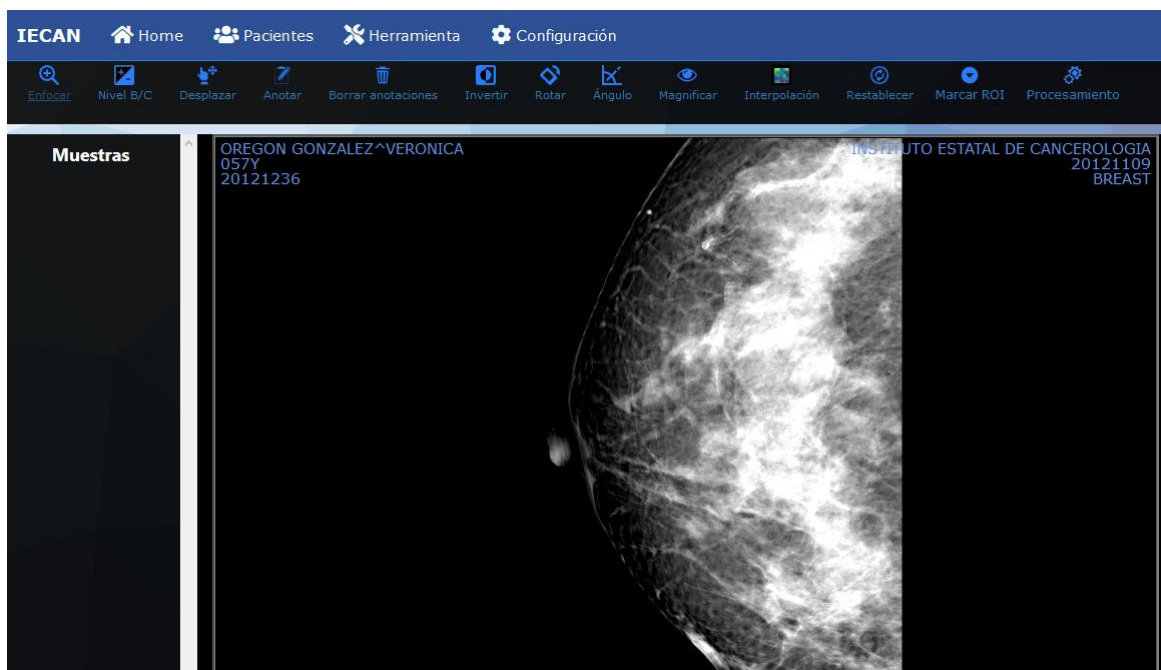


Figura 5.7. Funcionalidad Nivel B/C de la herramienta.

En la figura anterior se visualiza la imagen después de haber sido manipulada con la funcionalidad Nivel B/C, el médico o especialista podrá ajustar el brillo y contraste de la imagen dependiendo de sus necesidades, mejorando así su calidad, el tiempo de respuesta entre el proceso que hay con esta funcionalidad es al instante pues no requiere de muchos cálculos.

Funcionalidad *Desplazar* (Figura 5.8), se utiliza para mover la imagen en diferentes áreas del contenedor donde se encuentra. Esta funcionalidad trabaja en combinación con *Enfocar*, debido a que cuando se enfoca demasiado la imagen ciertas partes pueden quedar ocultas, y el poder desplazar la imagen permite visualizarla y así evitar pérdidas de información.

El usuario podrá desplazar la imagen en tiempo real sin contratiempo, esto es debido a que todo el proceso se realiza en el navegador y no en el servidor.

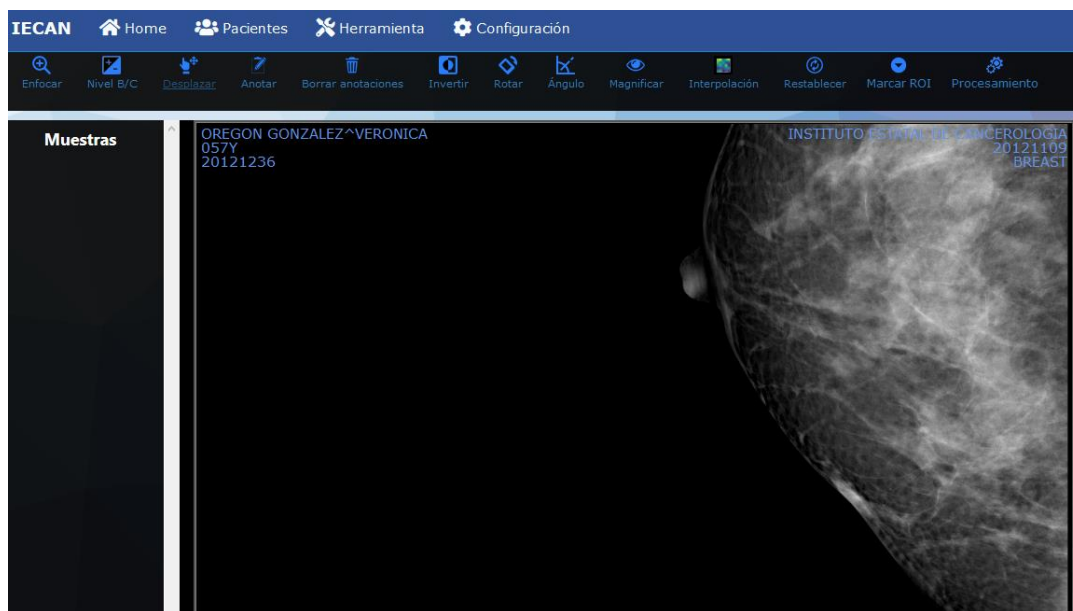


Figura 5.8. Funcionalidad Desplazar de la herramienta.

La funcionalidad *Anotar*, permite hacer anotaciones en las imágenes, tal como se muestra en la Figura 5.9.

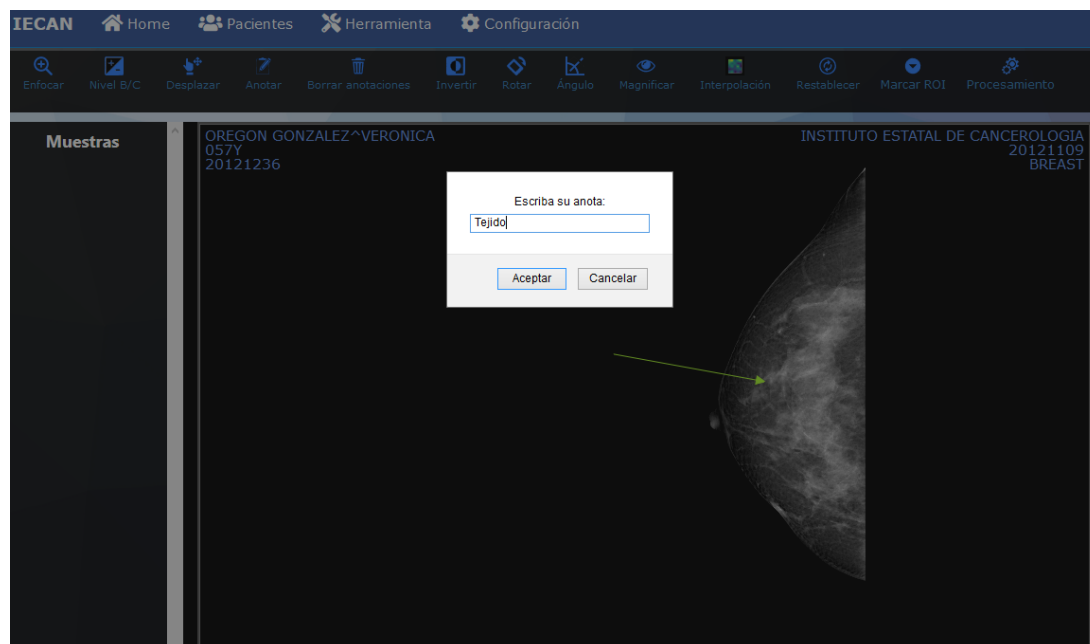


Figura 5.9. Funcionalidad Anotar de la herramienta.

El realizar anotación es de gran utilidad debido a que puede haber varias zonas extrañas en la imagen y el usuario tiene la oportunidad de realizar las anotaciones que sean necesarias.

Funcionalidad *Invertir* (Figura 5.10), invierte los colores de una imagen. Este comando se puede utilizar, por ejemplo, para hacer un negativo de una imagen en positivo en blanco y negro o para hacer un positivo a partir de un negativo en blanco y negro.

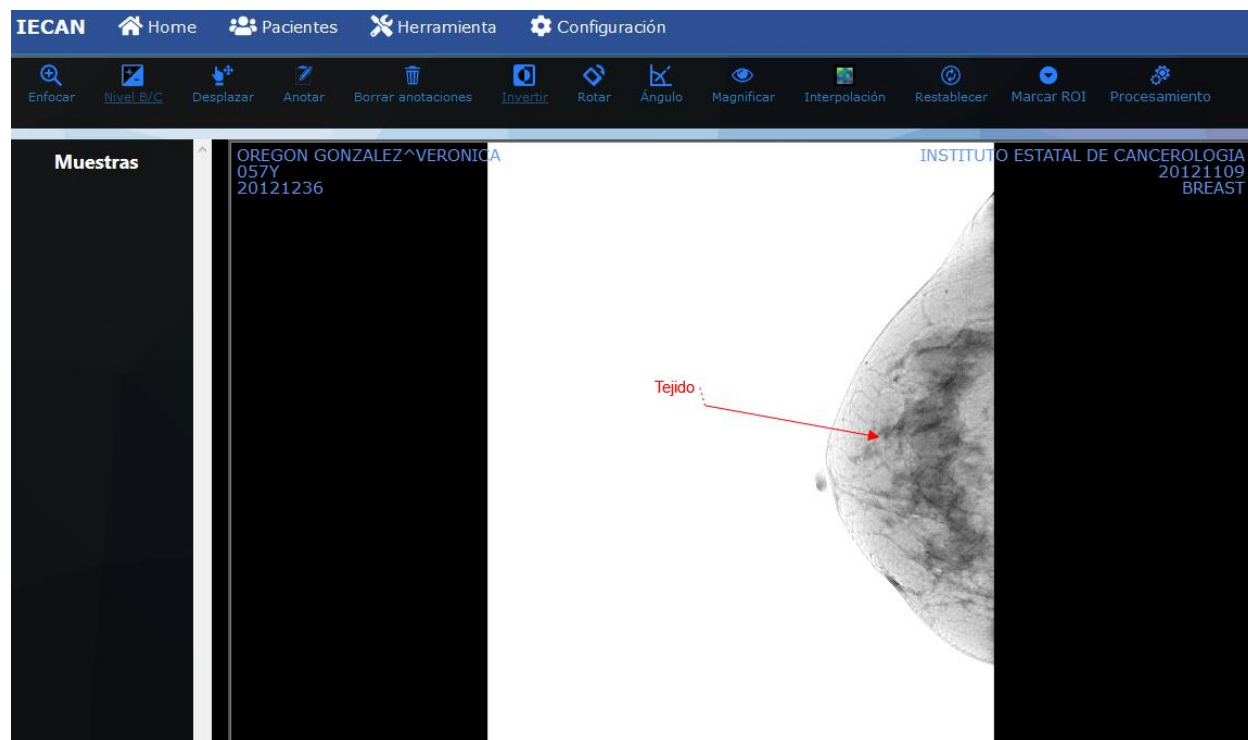


Figura 5.10. Funcionalidad *Invertir* de la herramienta.

El resultado de esta imagen permite visualizar la imagen desde otro ángulo y así tener otro punto de perspectiva.

Funcionalidad *Magnificar*, permite hacer más grande ciertas áreas de la imagen sin tener la necesidad de hacer zoom, tal como se puede visualizar en la Figura 5.11, el resultado es un efecto de lupa.

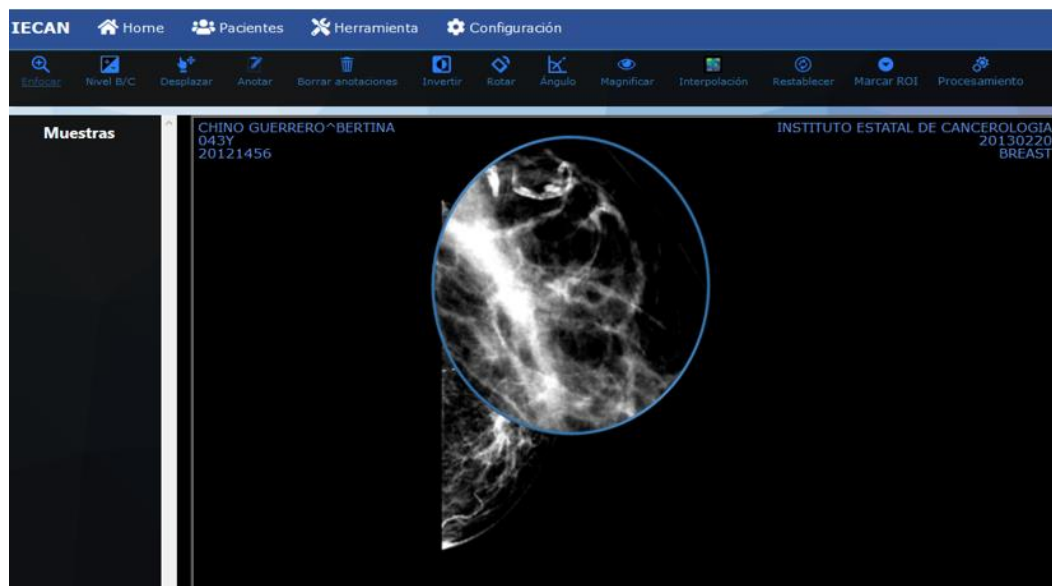


Figura 5.11. Funcionalidad Magnificar de la herramienta.

Funcionalidad *Interpolación* (Figura 5.12), permite crear nuevos píxeles a partir de los píxeles ya dado. Cuando aumenta el tamaño de una imagen se empieza a perder calidad y se empieza a ver borrosa, con el uso de la interpolación se le da más definición a la imagen creando nuevos píxeles.



Figura 5.12. Funcionalidad Interpolación de la herramienta.

La Figura 5.12. muestra una imagen con pixeles marcados, esto hace que al hacer zoom a la imagen no se empiece a ver borrosa. Al aplicar esta funcionalidad los cambios se mostrarán en tiempo real, pues no requiere de gran procesamiento por parte de la computadora.

funcionalidad Marcar ROI (Figura 5.13), permite marcar regiones de interés en forma circular o rectangular.

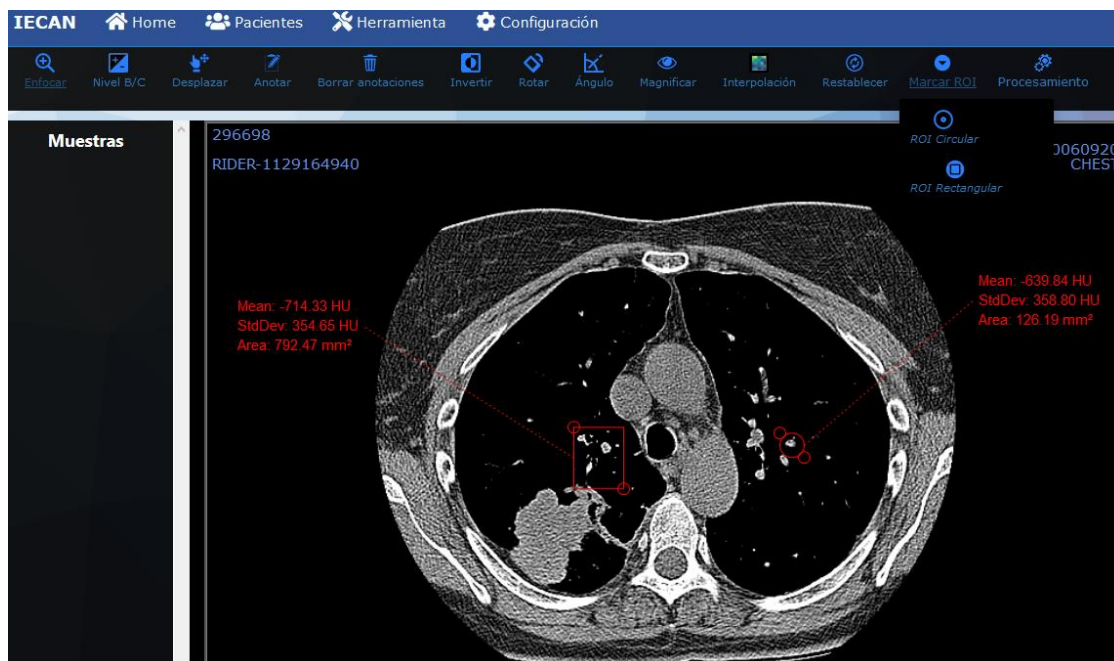


Figura 5.13. Funcionalidad Marcar ROI de la herramienta.

La funcionalidad de *Procesamiento* se divide en dos secciones, detección de bordes y suavizado de imagen, estos dos tipos de procesamiento de imagen trabajan en conjunto, debido a que cuando se aplica una detección de bordes este procesamiento puede llegar a ocasionar ruido en la imagen, para ello se utiliza el procesamiento de suavizado de imagen, para remover ese ruido ocasionado por la detección de bordes. El ruido puede ser causado por 3 razones:

- Equipo empleado - El tamaño de los fotocaptadores, el tipo de filtros que el sensor lleve incorporado, los procesadores de transformación analógico/digital, son elementos que afectan en gran medida a la aparición o no de ruido digital en las imágenes.
- Condiciones de la captura – Una baja iluminación podría ocasionar ruido.
- Procesado de la imagen – En cuanto menos se e hace ajuste a una imagen, menor será el ruido que tenga.

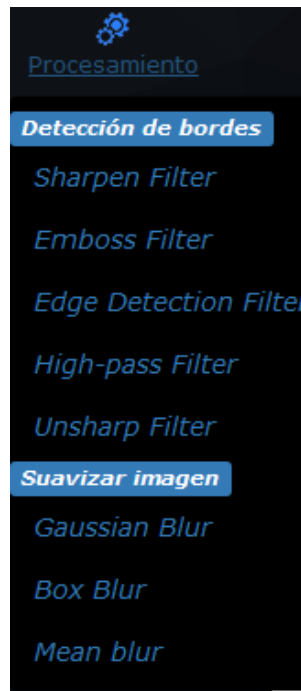


Figura 5.14. Tipos de procesamiento de imagen en la herramienta.

En la Figura 5.15 se muestra una imagen después de haber sido procesada con *Usharper Filter*, este filtro se utiliza para realzar bordes en la imagen.

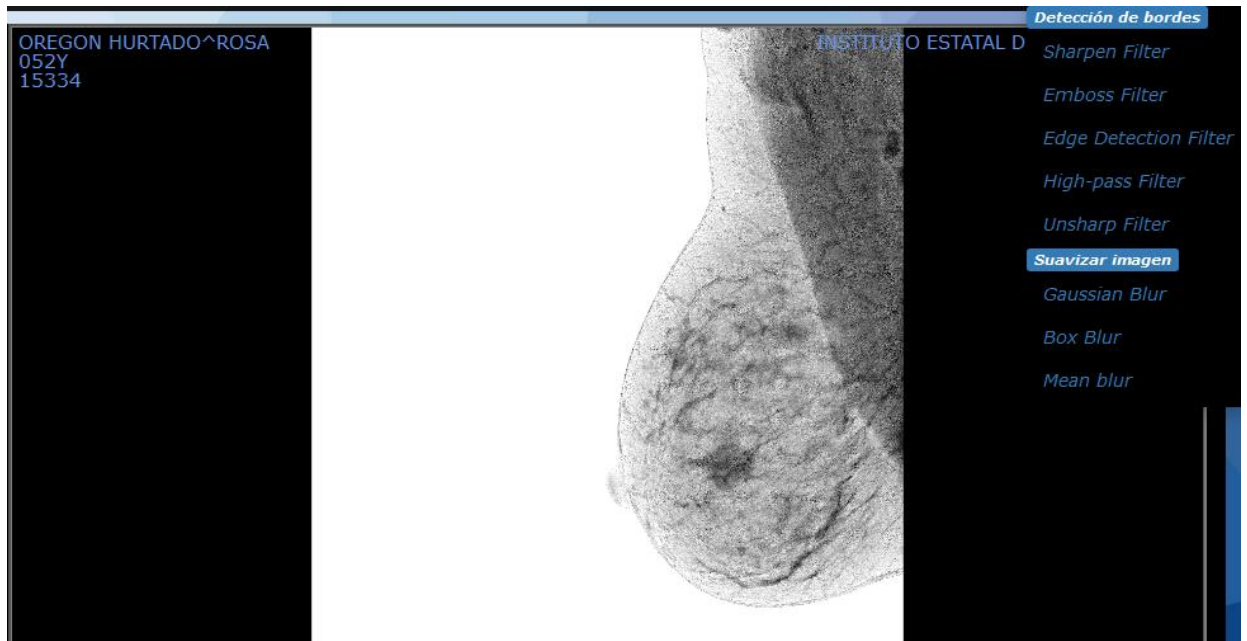


Figura 5.15. Procesamiento de detección de bordes en la imagen.

El resultado obtenido es una imagen con bordes más remarcados, sin embargo, este procesamiento puede causar ruido a la imagen, es por eso que es necesario hacer un procesamiento con un filtro de suavizado

En la Figura 5.16 se visualiza la misma imagen que fue procesada previamente, ahora se le aplicó un filtro de suavizado de imagen *Gaussian Blur*, este filtro se utiliza para eliminar el ruido que la imagen pudiera tener.

El resultado obtenido fue quitar el ruido que había ocasionado el filtro de detección de bordes sin perder los bordes remarcados.

Cabe mencionar que entre más grande sea el kernel mayor es el tiempo que tarde en realizar el procesamiento, debido a que la computadora tiene que hacer un mayor número de cálculos.



Figura 5.16. Procesamiento de suavizado en la imagen.

5.4 Conclusiones

En el Instituto Estatal de Cancerología del Estado se atienden a una gran cantidad de pacientes con casos de cáncer o posibles casos de que padezca algún tipo de cáncer. El IECan se especializa en atender 7 tipos de cánceres, los cuales son:

- Tumores Sólidos Fuera del Sistema Nervioso Central
- Hematopatías Malignas
- Cáncer de Mama
- Cáncer Cérvico Uterino
- Linfoma No Hodgkin
- Cáncer de Testículo
- Cáncer de Próstata

Por cada estudio o examen se obtienen una gran cantidad de imágenes médicas, para poder analizarlas se necesita de una herramienta moderna que auxilie a los especialistas a realizar su trabajo de una forma eficaz. Las imágenes que se manipularon para obtener los resultados de este proyecto van desde el año 2013 hasta el 2019.

De acuerdo a los resultados obtenidos en la evaluación del usuario, se puede concluir que la herramienta web es intuitiva, fácil de manejar y de gran utilidad para minimizar la carga de trabajo de los médicos o especialistas.

Con la implementación de la herramienta web aumentó la productividad de los especialistas, debido a que se evitan pérdidas de tiempo, los especialistas pueden acceder y hacer uso de esta herramienta desde cualquier lugar en que se encuentren, ya sea por medio de computadora de escritorio o laptop. Con las pruebas realizadas se logró mejorar significativamente la calidad y resolución de las imágenes, además de que esta herramienta brinda la capacidad de visualizar imágenes de múltiples modalidades como son: Resonancia Magnética (MR), Tomografías Computarizadas (TC), Rayos X, Mamografías (MG), entre otras.

Bibliografía

- American Cancer Society*. (20 de Enero de 2016). Obtenido de American Cancer Society: <https://www.cancer.org/es/tratamiento/como-compnder-su-diagnostico/pruebas/riesgos-de-la-radiacion-asociados-con-los-estudios-por-imagenes.html>
- Ben, C., Xiuling, L., Bin, D., & Feng, L. (2015). Web-based Information Retrieval and Visualization for Diagnostic Radiology. *19th International Conference on Information Visualisation*, 524-529.
- Bojecho, C., & Kroker, A. (24 de Octubre de 2015). *DICOM and other software standards*. Obtenido de http://199.116.233.101/index.php/DICOM_and_other_software_standards
- Convolution Matrix*. (2016). Obtenido de <https://docs.gimp.org/2.8/en/plugin-convmatrix.html>
- Dao-ming, L., Yu, T., Yu, W., Dan-yang, T., Wei-wei, Y., & Jing-song, L. (2015). Design and implementation of clinical data integration and management system based on Hadoop platform. *7th International Conference on Information Technology in Medicine and Education*, 76-79.
- DICOM Digital Imaging and Communications in Medicine*. (2019). Obtenido de <https://www.dicomstandard.org/>
- Digital Imaging and Communications in Medicine (DICOM)*. (2017). Obtenido de <https://www.dicomstandard.org/News/ftsup/docs/sups/sup23.pdf>
- Europea, C. (2001). *Guía de indicaciones para la correcta solicitud de pruebas de diagnóstico por imagen*. Italy.
- González Pérez, A. D., Zamora Calderón, S., & Núñez Hernández, M. M. (2009). *Factores que influyen en la calidad de la imagen radiográfica*. Obtenido de http://letras-uruguay.espaciolatino.com/aaa/gonzalez_perez_armando/factores_que_influyen.htm

Gutierrez Congo, O. L., & Pulido, C. M. (2009). EL DIAGNÓSTICO POR IMÁGENES: SOLUCIONES FACTIBLES AL DILEMA SALUDENFERMEDAD DESDE LA PERSPECTIVA CTS. *Ciencia y Sociedad*, vol. XXXIV, 32-51.

HL7 International. (2019). Obtenido de Introduction to HL7 Standards: <https://www.hl7.org/implement/standards/index.cfm?ref=nav>

Html y Css. (24 de enero de 2019). Obtenido de <https://japp.es/html-y-css/>

HTML: Hypertext Markup Language. (18 de mayo de 2019). Obtenido de <https://developer.mozilla.org/en-US/docs/Web/HTML>

IMÁGENES MÉDICAS. (2008). Obtenido de <http://bibing.us.es/proyectos/abreproy/11854/fichero/Volumen+1%252FCapitulo+2.pdf>

ImagenYON. (2018). Obtenido de <http://www.fisi-on.com/imagenyon/pruebas-diagnosticas/>

Javascript. (10 de agosto de 2015). Obtenido de <https://sites.google.com/site/istteclpv/calendario-de-trabajo/home/2015-i/temas-a-investigar/framework---javascript/javascript>

Jodogne, S. (2018). The Orthanc Ecosystem for Medical Imaging. *Journal of Digital Imaging*.

Leadtools. (2017). Obtenido de <https://www.leadtools.com/sdk/medical/dicom-spec1>

Marques Godinho, T., Almeida, E., Bastião Silva, L. A., & Costa, C. (2016). Integrating Multiple Data Sources in a Cardiology Imaging Laboratory. *2016 IEEE 18th International Conference on e-Health Networking, Applications and Services (Healthcom)*.

MedlinePlus. (12 de Noviembre de 2018). Obtenido de <https://medlineplus.gov/spanish/diagnosticimaging.html>

MedlinePlus. (2019). Obtenido de Imagenología y radiología: <https://medlineplus.gov/spanish/ency/article/007451.htm>

- Microsoft Azure*. (2019). Obtenido de ¿Qué es middleware?: <https://azure.microsoft.com/es-mx/overview/what-is-middleware/>
- Mineiro, L., & Tomé, P. (2015). The integration of radiology information systems. *10th Iberian Conference on Information Systems and Technologies (CISTI)*.
- Narváez, M., Merello, E., Toribio, C., & Benlloch, J. M. (2009). *Ciencias de la Salud: Diagnóstico por Imagen*. Madrid.
- Pressman, R. S. (2010). *Ingeniería de software, un enfoque práctico*. New York: McGraw-Hill.
- PS3.1: DICOM PS3.1 2019b - Introduction and Overview*. (2019). Retrieved from <https://www.dicomstandard.org/current/>
- PSICOLOGÍA DE LA PERCEPCIÓN VISUAL*. (2018). Obtenido de 3.7 EL FILTRADO DE IMÁGENES: <http://www.ub.edu/pa1/node/filtrado>
- Ramon, R. (12 de Agosto de 2013). *Imagen Radiológica Digital (PACS/RIS) mejora servicios, salva vidas y protege el medioambiente*. Obtenido de <https://www.ramonramon.org/blog/2013/08/12/imagen-radiologica-digital-pacsr-is-y-medioambiente/>
- Roth, D., Anderson, R., & Luttin, S. (6 de Abril de 2019). *Microsoft*. Obtenido de Introduction to ASP.NET Core: <https://docs.microsoft.com/en-us/aspnet/core/?view=aspnetcore-2.2>
- Savaris, A., Gimenes Marques Filho, A. A., Rodrigues Pieres de Mello, R., & Bussolo Colonetti, G. (2017). Integrating a PACS Network to a Statewide Telemedicine System, A case study of the Santa Catarina State Integrated Telemedicine and Telehealth System. *2017 IEEE 30th International Symposium on Computer-Based Medical Systems*, 356-357.
- Smith, S. (1 de Julio de 2018). *Microsoft*. Obtenido de Overview of ASP.NET Core MVC: <https://docs.microsoft.com/en-us/aspnet/core/mvc/overview?view=aspnetcore-2.2>

- Systems, F. (2013). *the complete guide TO BUYING A PAC SYSTEM*. Obtenido de <https://freelandsystems.com/guide-to-buying-pac-system/>
- Ukis, V., Balachandran, B., Tirunellai Rajamani, S., & Friese, T. (2013). Architecture of Cloud-Based Advanced Medical Image Visualization Solution. *2013 IEEE International Conference on Cloud Computing in Emerging Markets (CCEM)*.
- Un paseo por el lenguaje C#*. (4 de abril de 2019). Obtenido de Un paseo por el lenguaje C#: <https://docs.microsoft.com/es-es/dotnet/csharp/tour-of-csharp/>
- Valente, F., Bastião Silva, L. A., Marques Godinho, T., & Costa, C. (2015). Anatomy of an Extensible Open Source PACS. *Journal of Digital Imaging*.
- Wan, Y., Hu, H., Quan, C., Guo, M., & Wang, Y. (2016). Analysis and Design of Multi-Pattern Retrieval Scheme of Regional PACS System Based Medical Images. *8th International Conference on Information Technology in Medicine and Education*, 78-82.
- Yeung, S., & Alahi, A. (2015). *Introduction to Computer Vision*. Obtenido de Image Filtering: <http://ai.stanford.edu/~syyeung/cvweb/tutorial1.html>