

Universidad de las Ciencias Informáticas

Facultad 2



Componente para la impresión convencional de imágenes médicas.

Trabajo de Diploma para optar por el Título de
Ingeniero en Ciencias Informáticas

Autores: Madeivy Sánchez Pérez

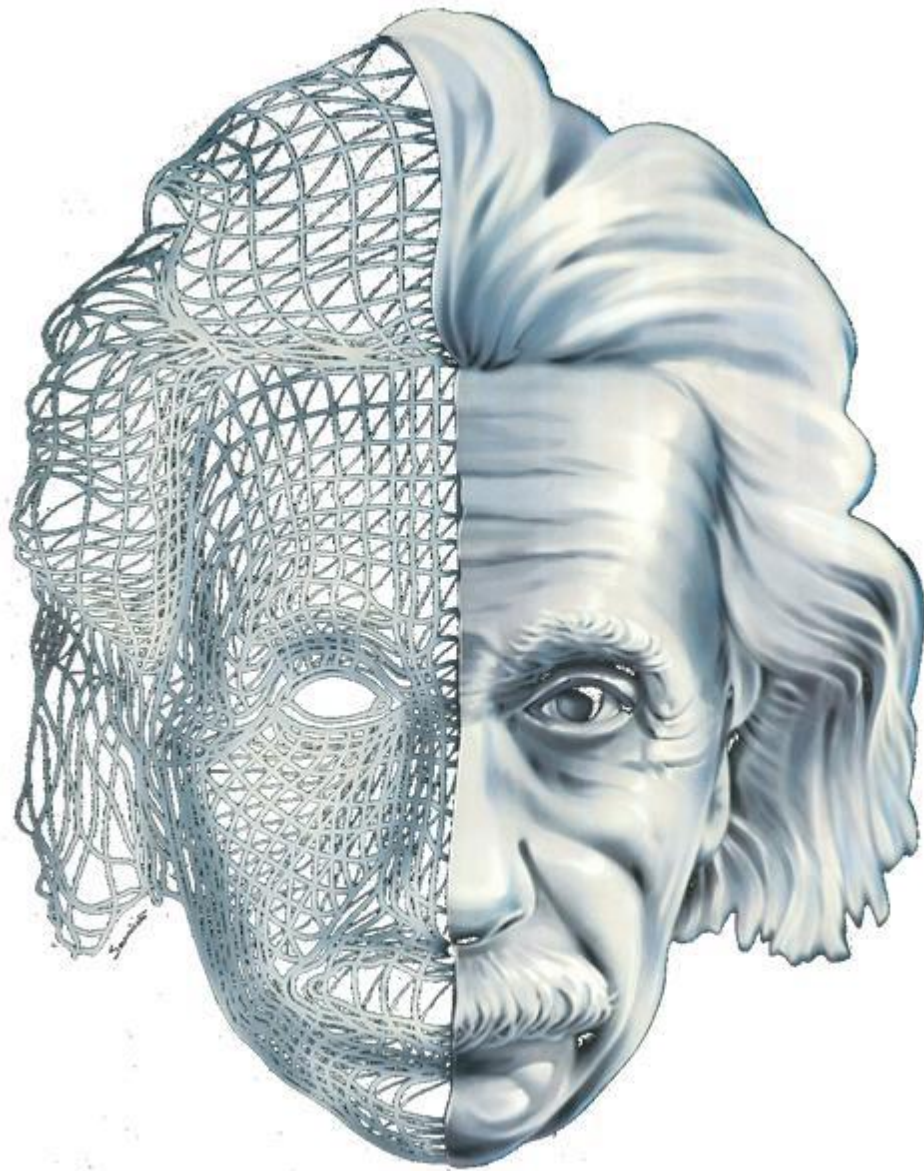
Digmaris Nuñez Gómez

Tutores: Ing. Leudis Hernández Sánchez

Ing. Carlos Luis Castro Márquez

La Habana, 25 de Junio de 2014

“Año 56 de la Revolución”



*"Quien se considera a sí mismo como un juez de la verdad y el conocimiento,
es destruido por la risa de los dioses"*

Albert Einstein

DATOS DE CONTACTO

Tutores:

Tutor principal: *Ing. Leudis Hernández Sánchez:*

Graduado de Ingeniero en Ciencias Informáticas, egresado de la UCI en el 2010. Actualmente se desempeña como jefe de proyecto y programador en el Departamento de Producción de Software Médico Imagenológico, CESIM. Correo electrónico: lhsanchez@uci.cu.

Co tutor: *Ing. Carlos Luis Castro Márquez:*

Graduado de Ingeniero en Ciencias Informáticas, egresado de la UCI en el 2012. Actualmente se encuentra desarrollando su período de adiestramiento para profesor de la Facultad 7 y programador en el Departamento de Producción de Software Médico Imagenológico, CESIM. Correo electrónico: clcastro@uci.cu.

DEDICATORIA

De Digmari:

Quisiera dedicar este trabajo en especial a mi mamá Dignuris y mi papá Rigoberto, porque ellos son el motor de mi vida, mi luz cuando alrededor todo es oscuridad, mi ejemplo a seguir, son las personas por las cuales cada día quiero ser mejor persona y los que me hacen cada mañana despertar con deseos de luchar y seguir adelante. Gracias por acompañarme en todo, confiar en mi cuando a veces ni yo misma lo hago, por su apoyo y comprensión. Los Amo.

A mis abuelos que son mi mayor tesoro, los que siempre me consienten en todo, en especial a mi abu Orfe que sin ella mi vida no tendría sentido. No tengo palabras para decirle todo lo que siento por ella, solo agradecerle por ser mi confidente, mi amiga, por todas las cosas que hace por mí, todo lo que me ha dado incluso más de lo que ha estado su alcance. Gracias mi viejita linda, te adoro y tú lo sabes. Los quiero a todos.

A mis hermanos, por ser mis dos guardianes, mis cómplices en las buenas y malas. Por ustedes cada día me esfuerzo más, para ser un buen ejemplo. Los quiero mucho.

A mi familia en general, por ser lo más valioso y grande de este mundo. Por el apoyo, la confianza y el cariño que me dan día tras día. Me atrevería a decir sin miedo alguno que es la mejor familia que hubiera podido desear cualquier persona. Los quiero mucho a todos, y cada uno de ustedes son una pieza fundamental en el rompecabezas de mi corazón.

A todos mis amigos que han estado a lo largo de mi vida conmigo, los de la UCI, los del barrio y a los otros que sin importar donde estén o donde se encuentren se les quiere igual.

La dedico también a los que una vez comenzaron este sueño pero no pudieron llegar hasta el final de la meta. Los quiero y los recordaré por siempre.

Para todos ustedes va dedicado este trabajo.

De Made:

A mi familia, en especial a mi tía Mayelin.

AGRADECIMIENTOS

De Digma:

Primeramente quiero agradecer a Dios por darme las fuerzas necesarias para seguir adelante.

A la Revolución por darnos la oportunidad de graduarnos hoy aquí, en esta Universidad de excelencia, por darnos una oportunidad sin importar de donde vengamos.

A mis padres por su apoyo, sus consejos, su confianza y su amor infinito. Por ser mis cómplices en esta travesía que hoy está llegando a su fin.

A mis hermanos, abuelos, tíos y primos por estar siempre presente cuando más los necesito y por su apoyo incondicional.

A mis amigos de la vieja escuela por seguir confiando en mí a través de todos estos años y por la amistad tan linda que tenemos y que ha prevalecido a pesar de la distancia hasta la actualidad.

A mi hermanita Lili, gracias por tus regaños, tus consejos, los buenos y malos momentos y por ser además mi impulso y ejemplo a seguir. Sabes que te quiero de corazón.

A mi hermanita negra Zaida, gracias por los consejos, por la amistad y el cariño que me has dado y me que me das.

A Gilbert por ser un amigo, un confidente, mi padrino.

A mi tata Heidi, que en el poco tiempo que hemos compartido se ha ganado mi corazón y mi cariño. Gracias por ser mi otra compañera de tesis.

A Fernandito por ser mi guardián en la Universidad.

A Edelena que es mi gitana favorita y aun en la distancia sigue estando aquí presente a mi lado.

A Dayi (la mía), por ser mi amiga, mi confidente y por abrirme las puertas de su casa y de su corazón.

A mis amigas del alma: Annabellis (Bellota), Yas, Leanni, Yaima y Yicel que siempre han estado ahí para mí.

A todas las familias de mis amigas por aceptarme en su hogar, por brindarme su cariño, por ser una integrante más de la familia, por sus consejos y por su ayuda. Los considero mi familia.

A Odesney por ser como fue conmigo, por soportarme y quererme.

A mi familia de Alquizar, por aceptarme y quererme como soy.

Al piquete de los míos, Landy, Alberto, Ricardo (el perro), Freddy y Pity.

A las niñas del apartamento de enfrente, por aceptar a esta inquilina.

A mis compañeros del 7101, 7305 y actualmente el 2502 por aceptarme como soy, por ayudarme, por su amistad, por su cariño y por demostrarme que nunca estuve sola.

A mis tutores porque más que tutores son mis amigos, y a Leudis en especial que fue un compañero de tesis más.

A todos los profesores por los conocimientos que me proporcionaron y que han hecho de mi lo que soy hoy.

A las personas que me ayudaron en la elaboración de este trabajo, sin ellos no hubiera sido posible.

A mi compañera de tesis por soportar mis regaños y ayudarme a lograr este sueño que hoy convierto en realidad.

A los profesores del departamento, que con su ayuda se mejoró la investigación realizada.

Al tribunal por sus consejos y recomendaciones.

A la UCI, por ser mi casa durante estos años, por la oportunidad de realizar mi sueño y por las nuevas amistades.

A todos los que están aquí conmigo y los que no pudieron estar, de una forma u otra, mil gracias.

Por último, le quiero agradecer también, a los obstáculos que han pasado por mi vida. Ellos son los que me han hecho más fuerte, perseverante y finalmente triunfadora.

A todos gracias.

De Made:

Quiero agradecer a mis padres, mi hermana y mis familiares por darme siempre su amor y cariño cuando más lo necesitaba.

A mi compañera de tesis por su paciencia y a mis amigos por haberme apoyado en todo momento.

A nuestros tutores, por toda su entrega en este trabajo, a los profesores del proyecto que nos ayudaron y en general a todos los que contribuyeron a nuestra formación.

A nuestros compañeros de estos 5 años, gracias por su amistad.

RESUMEN

El ser humano, en su afán de adquirir nuevos conocimientos ha tenido la necesidad de almacenar de alguna manera las investigaciones y estudios que ha realizado. En la actualidad, con el desarrollo de las tecnologías informáticas, existen sectores como el de la salud en el que las impresiones juegan un papel primordial en aras de preservar y emitir diagnósticos. Actualmente en la Universidad de las Ciencias Informáticas se desarrolló un sistema que permite la impresión de las imágenes médicas en placas, este proceso provoca la contaminación del medio ambiente, se utilizan químicos dañinos para la salud del personal que realiza este trabajo y es un proceso engorroso y costoso. Como resultado de un estudio previo se obtuvo un componente que permitirá la integración con la Plataforma de Gestión de Imágenes, de esta manera se proporcionará a los especialistas una herramienta que les permitirá realizar las impresiones de las imágenes médicas de una manera más rápida, permitiéndoles ahorrar tiempo y de esta manera se podrán aumentar los diagnósticos realizados en un día. Además le ahorrará al país considerables montos de dinero, ya que los productos que se necesitan para realizar las placas tienen precios elevados. Para la generación de los artefactos de ingeniería se utilizó la herramienta Enterprise Architect (versión 7.5), haciendo uso UML para el modelado del dominio, análisis y diseño, RUP como base de la metodología de desarrollo y se emplearon las guías y prácticas de CMMI. La aplicación fue desarrollada sobre la plataforma .NET Framework (versión 4.0) con C# como lenguaje de programación, y la utilización de Visual Studio 2010 como IDE de desarrollo.

Palabras claves:

Imágenes médicas, impresora, placas, medio ambiente.

TABLA DE CONTENIDOS

INTRODUCCIÓN	1
CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA-METODOLÓGICA DE LA IMPRESIÓN DE LAS IMÁGENES MÉDICAS.	7
1.1. HISTORIA DE LA IMPRESIÓN.....	7
1.2. SISTEMAS INFORMÁTICOS INTERNACIONALES.	8
1.3. INTERNACIONALES.	8
1.4. NACIONALES.....	9
1.5. ESTÁNDAR DICOM.	10
1.6. TECNOLOGÍAS Y HERRAMIENTAS.	11
1.7. HERRAMIENTAS PARA EL MODELADO.	14
1.8. MODELOS DE CALIDAD.	16
1.9. CONCLUSIONES DEL CAPÍTULO.	18
CAPÍTULO 2. CARACTERÍSTICAS DEL COMPONENTE PARA LA IMPRESIÓN CONVENCIONAL DE IMÁGENES MÉDICAS.	19
2.1. BREVE DESCRIPCIÓN DEL COMPONENTE.	19
2.2. MODELO DE DOMINIO.	20
2.3. ESPECIFICACIÓN DE LOS REQUISITOS DEL SOFTWARE.	22
2.4. DEFINICIÓN DE LOS ACTORES DEL SISTEMA.	26
2.5. DIAGRAMA DE CASOS DE USO DEL SISTEMA.....	26
2.6. DESCRIPCIÓN DE LOS CASOS DE USO DEL SISTEMA.	27
2.7. CONCLUSIONES DEL CAPÍTULO.	29
CAPÍTULO 3. ANÁLISIS Y DISEÑO DEL COMPONENTE PARA LA IMPRESIÓN CONVENCIONAL DE IMÁGENES MÉDICAS.	30
3.1. MODELO ARQUITECTÓNICO.....	30
3.2. DIAGRAMA DE CLASES.....	32
3.3. DISEÑO.....	32
3.4. DIAGRAMAS DE SECUENCIA.	32
3.5. PATRONES DE DISEÑO.....	33

3.6. CONCLUSIONES DEL CAPÍTULO.	34
CAPÍTULO 4. IMPLEMENTACIÓN DEL COMPONENTE PARA LA IMPRESIÓN CONVENCIONAL DE IMÁGENES MÉDICAS.	35
4.1. DIAGRAMA DE COMPONENTES.....	35
4.2. DESCRIPCIÓN DE LOS COMPONENTES.....	36
4.3. DIAGRAMA DE DESPLIEGUE.....	36
4.4. NODOS DEL DESPLIEGUE.	37
4.5. ESTÁNDARES DE CODIFICACIÓN.	37
4.6. CONCLUSIONES DEL CAPÍTULO.	39
CONCLUSIONES.....	40
RECOMENDACIONES.....	41
REFERENCIAS BIBLIOGRÁFICAS.....	42
BIBLIOGRAFÍA.....	47
ANEXOS	52
ANEXO 1. ENTREVISTA REALIZADA AL LICENCIADO EN IMAGENOLOGÍA Y ESPECIALISTA EN LOS TAC, ROBERTO MIRANDA BARRABIA.	52
ANEXO 2. PREFIJOS DE LOS REQUERIMIENTOS NO FUNCIONALES.	52
ANEXO 3. DESCRIPCIÓN DETALLADA DE LOS CASOS DE USO DEL SISTEMA.....	53
ANEXO 4. DIAGRAMAS DE LAS CLASES DEL SISTEMA.	61
ANEXO 5. DIAGRAMAS DE SECUENCIA DE LOS CASOS DE USO.	64
GLOSARIO DE TÉRMINOS.....	66

INTRODUCCIÓN

La creciente evolución de las Tecnologías de la Información y las Comunicaciones (TIC) (1) representa un gran avance en las disímiles esferas de la sociedad tales como: la educación, las comunicaciones, la salud entre otras. Es importante destacar que en el campo de la salud, se han realizado avances científicos (2) (3) muy notorios en las diferentes ramas de la misma. Actualmente, entre los cambios en la atención de la salud, a través del uso de la informática se han podido implementar sistemas capaces de apoyar y mejorar la labor que se realiza en las diversas esferas de la medicina facilitando así el trabajo a todas las personas que interactúen con estos sistemas.

La informática ha tenido un gran auge en la medicina, surgiendo de esta manera el término informática médica, el cual se refiere a la organización de información y servicios de salud utilizando el internet y otras aplicaciones relacionadas a esta. El objetivo fundamental de la misma es brindar una ayuda a los profesionales de la salud para mejorar la calidad de la atención al paciente, ya que propicia el diagnóstico por imagen, la telemedicina, los sistemas de gestión hospitalaria y registro clínico. Esta se aplica en los laboratorios de análisis clínicos, dispositivos electrónicos para hacer mediciones, archivos de imágenes, software de gestión hospitalaria, manejo de turnos, de historias clínicas y bases de datos de pacientes (4).

Como respuesta tecnológica al difícil manejo de la creciente cantidad de información proveniente de los distintos métodos de diagnóstico por imagen antes mencionados, surgen los Sistemas para la visualización, almacenamiento y transmisión de imágenes médicas (PACS por sus siglas en inglés) (5). Un PACS está compuesto por dispositivos para la adquisición de imágenes, unidades de almacenamiento, estaciones de visualización, impresoras DICOM¹ y servidores centrales de bases de datos y de imágenes médicas; todos ellos integrados por las redes digitales y software de aplicación y conectados a través de una red de comunicación. Debe ser capaz de recibir imágenes de cualquier fabricante, saber interpretarlas y extraer la información de forma correcta. Es el estándar DICOM quien permite que el PACS pueda manipular imágenes de diversos tipos de equipos médicos, incluyendo ultrasonido, resonancia magnética, tomografía computarizada, densitometría ósea, entre otros (6).

¹ Estándar para la Transmisión y Manejo de Imágenes Médicas.

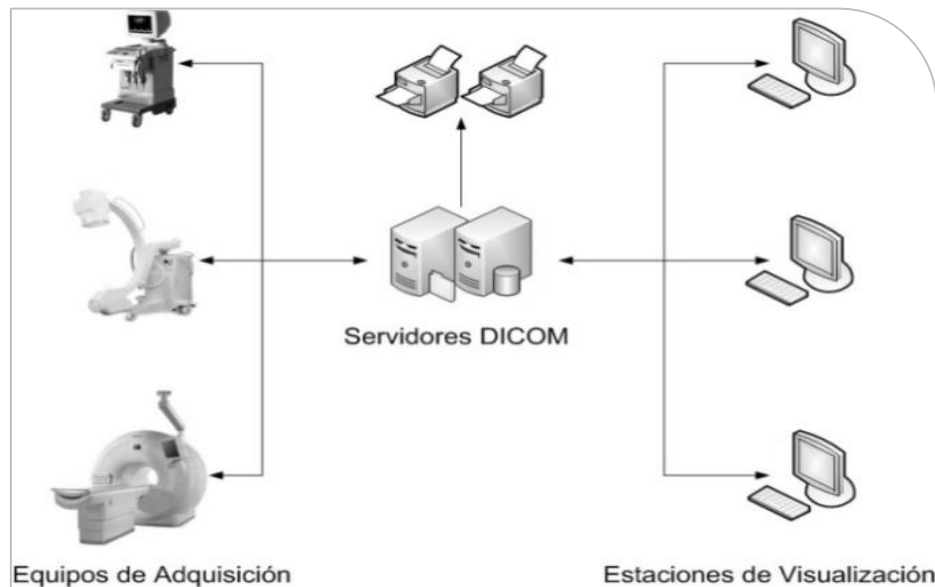


Figura 1. Componentes de un sistema PACS.

En el mundo existen diversas aplicaciones y sistemas sofisticados para realizar diagnósticos a los pacientes, donde algunos de ellos dan la posibilidad al especialista de imprimir los resultados para ser entregados al paciente. Imprimir no es más que dejar una marca, texto escrito, dibujo o figura en un papel u otro soporte por medio de procedimientos mecánicos o digitales (7).

En la Universidad de las Ciencias Informáticas (UCI), existen varios departamentos que se especializan en diferentes niveles y áreas de atención a la rama de la salud, entre los que se destaca por su labor y buen desempeño el departamento Software Médico Imagenológico (SWMI), perteneciente al Centro de Informática Médica (CESIM) e integrado por especialistas, estudiantes y profesores. Uno de los sistemas desarrollados en el departamento anteriormente mencionado es la Plataforma de Gestión de Imágenes (PACS-RIS), la cual posee entre sus componentes a un PACS. Entre sus funcionalidades se encuentran la visualización y procesamiento de las imágenes médicas, con el objetivo de resaltar regiones específicas que aporten un mejor diagnóstico al paciente, y la impresión de dichas imágenes en impresoras DICOM con el objetivo de preservar y trabajar con los diagnósticos realizados.

Las soluciones y proyectos que se realizan en la Universidad antes mencionada, están en consonancia con los lineamientos de la política económica y social que aprobó el VI Congreso del PCC² el 18 de Abril de 2011 para actualizar el modelo económico cubano, que engloba todas las esferas susceptibles de incidir en el desarrollo económico del país. A continuación se presentan algunos lineamientos que sirvieron de base para el desarrollo de esta investigación (8).

Capítulo V Política de Ciencia, Tecnología, Innovación y Medio Ambiente.

129. Diseñar una política integral de ciencia, tecnología, innovación y medio ambiente que tome en consideración la aceleración de sus procesos de cambio y creciente interrelación a fin de responder a las necesidades del desarrollo de la economía y la sociedad a corto, mediano y largo plazo; orientada a elevar la eficiencia económica, ampliar las exportaciones de alto valor agregado, sustituir importaciones, satisfacer las necesidades de la población e incentivar su participación en la construcción socialista, protegiendo el entorno, el patrimonio y la cultura nacionales.

133. Sostener y desarrollar investigaciones integrales para proteger, conservar y rehabilitar el medio ambiente y adecuar la política ambiental a las nuevas proyecciones del entorno económico y social. Priorizar estudios encaminados al enfrentamiento al cambio climático y, en general, a la sostenibilidad del desarrollo del país. Enfatizar la conservación y uso racional de recursos naturales como los suelos, el agua, las playas, la atmósfera, los bosques y la biodiversidad, así como el fomento de la educación ambiental.

135. Definir una política tecnológica que contribuya a reorientar el desarrollo industrial, y que comprenda el control de las tecnologías existentes en el país; a fin de promover su modernización sistemática atendiendo a la eficiencia energética, eficacia productiva e impacto ambiental, y que contribuya a elevar la soberanía tecnológica en ramas estratégicas. Considerar al importar tecnologías, la capacidad del país para asimilarlas y satisfacer los servicios que demanden, incluida la fabricación de piezas de repuesto, el aseguramiento metrológico y la normalización.

² Partido Comunista de Cuba.

Capítulo VI Política Social. Salud

154. Elevar la calidad del servicio que se brinda, lograr la satisfacción de la población, así como el mejoramiento de las condiciones de trabajo y la atención al personal de la salud. Garantizar la utilización eficiente de los recursos, el ahorro y la eliminación de gastos innecesarios.

El empleo de las impresiones DICOM trae consigo algunas desventajas, entre ellas se destaca que las impresoras DICOM donde se imprimen las imágenes pertenecen a proveedores de empresas multinacionales, que venden a un alto costo cada uno de sus componentes y muchas de las instituciones no cuentan con los recursos financieros para su adquisición. Actualmente, uno de los altos costos que tiene el país en el sector de la salud es obtención del rollo de revelación de placas, ya que son sumamente caros y no se pueden volver a utilizar, si se echa a perder una placa por alguna razón, se tiene que volver a repetir la película lo que trae consigo demasiadas radiaciones para el cuerpo humano que las está recibiendo, también implica el gasto innecesario del producto por un error llevado a cabo. Algunos hospitales que no cuentan con estas impresoras realizan el revelado tradicional de la imagen, para este proceso se requiere de un local con ciertos requerimientos específicos (cuarto oscuro), además del uso de productos químicos dañinos para la salud del personal que realiza este trabajo, además se necesita un tiempo de espera adicional para el proceso de revelar las imágenes que generalmente muestran la toma desde una única posición.

Sin importar donde se haga la grabación de la imagen, ya sea en una impresora DICOM especializada o mediante el revelado tradicional, estas placas resultantes tienen un alto costo ya que los materiales con que están creadas tienden a ser productos caros. Este proceso provoca la contaminación del medio ambiente y se hace incontrolable la recolección de las placas que los pacientes adquieren al realizarse este tipo de exámenes y luego llevan a sus hogares. El reciclado de estos materiales se convierte en un proceso complejo, además hay que depurar las aguas usadas para no dañar el medio ambiente.

La situación anteriormente descrita permitió identificar como **problema a resolver**: ¿Cómo viabilizar el proceso de impresión de las imágenes DICOM en papel?

Para el desarrollo de la investigación se define como **objeto de estudio**: El proceso de impresión de imágenes médicas.

Se delimita como **campo de acción**: El proceso de impresión convencional de imágenes médicas.

Componente para la impresión convencional de imágenes médicas. | **Introducción**

Para la solución del problema se plantea como **objetivo general**: Desarrollar un componente informático que permita viabilizar los procesos relacionados con la impresión en impresoras convencionales de las imágenes médicas digitales en formato DICOM.

Para dar cumplimiento al objetivo planteado se proponen las siguientes **tareas investigativas**:

- Realización de un análisis crítico y valorativo de los sistemas informáticos, que permiten la impresión de las imágenes médicas en placa y papel a nivel nacional e internacional para establecer similitudes con la investigación en curso y confeccionar el nuevo componente.
- Asimilar las herramientas, tecnologías y metodología propuestas por el departamento software Médico Imagenológico (SWMI), empleadas para el desarrollo del PACS-RIS.
- Generar los artefactos correspondientes a las fases de desarrollo: “Modelado del Negocio”, “Requisitos”, “Análisis y Diseño” e “Implementación”, propuestos por la metodología de Desarrollo Proceso Unificado de Desarrollo de Software (RUP).
- Implementación de un componente de software, que permita la configuración de las impresoras y la impresión de las imágenes médicas, aplicando las pautas de diseño y siguiendo lo establecido en la Especificación de Requisitos de Software.

El presente documento está estructurado por los **capítulos** que se relacionan a continuación:

Capítulo 1. Fundamentación teórica-metodológica de la impresión de las imágenes médicas: se definen los aspectos relacionados con la investigación realizada para determinar las características del sistema que se va a implementar, así como el análisis de soluciones existentes a nivel internacional y nacional. Además se proponen las herramientas, lenguajes de programación y metodología que se utilizarán para el desarrollo del componente informático.

Capítulo 2. Características del Componente para la impresión convencional de imágenes médicas: Se describen las características fundamentales del componente como parte de la propuesta de solución y se realiza una breve descripción de los requisitos funcionales y no funcionales a los que se debe dar cumplimiento. Se muestran además los casos de uso y una breve descripción de los mismos.

Capítulo 3. Análisis y Diseño del Componente para la impresión convencional de imágenes médicas: Centrado fundamentalmente en el análisis y diseño del componente. Se exponen los diagramas de clases y los diagramas de interacción según los casos de uso definidos, y se explica la arquitectura utilizada. Además se explican los patrones de diseño utilizados en el desarrollo de la aplicación.

Capítulo 4. Implementación del Componente para la impresión convencional de imágenes médicas: Se muestran los componentes resultantes del desarrollo de la aplicación, así como la descripción de los mismos. Además se muestra el diagrama de despliegue con la descripción de sus nodos y los estándares de codificación utilizados en la implementación del componente.

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA-METODOLÓGICA DE LA IMPRESIÓN DE LAS IMÁGENES MÉDICAS.

En este capítulo se recogen los principales conceptos vinculados a la investigación. Se realiza un estudio del estado del arte de los diferentes sistemas informáticos que realizan la impresión de imágenes médicas, existentes a nivel nacional e internacional. Se mencionan características de las tecnologías, patrones y lenguajes de programación utilizados. Se describen las herramientas, metodologías, modelo de calidad y notación de modelado designados en el departamento para el desarrollo de las aplicaciones, argumentando los beneficios de las mismas.

1.1. Historia de la impresión.

Desde tiempos remotos, el ser humano ha necesitado adquirir nuevos conocimientos, y con ellos la necesidad de guardar de alguna manera lo que aprendían o las investigaciones que realizaban, de esta forma surgieron los primeros manuscritos para posteriormente dar paso a los libros. Una vez que una persona escribía un libro o un artículo determinado existía la necesidad de tener varios ejemplares, y hasta el año 1449 y aun en años posteriores, los libros se difundían en copias manuscritas por copistas, muchos de los cuales eran monjes y frailes dedicados exclusivamente al rezo y a la réplica de ejemplares por encargo del propio clero o de reyes y nobles (9).

La imprenta había sido inventada siglos antes por los chinos, pero en la Edad Media se utilizaba en Europa para publicar panfletos publicitarios o políticos, etiquetas, y trabajos de pocas hojas; para ello se trabajaba el texto en hueco sobre una tablilla de madera, incluyendo los dibujos. Una vez confeccionada, se acoplaba a una mesa de trabajo, también de madera, y se impregnaban de tinta negra, roja, o azul (sólo existían esos colores) (9).

Después se aplicaba el papel y con rodillo se fijaba la tinta. El desgaste de la madera era considerable por lo que no se podían hacer muchas copias con el mismo molde. Cada impresor fabricaba su propio papel, estampando una marca de agua a modo de firma de impresor (10). Con el paso de los siglos y con el desarrollo vertiginoso de las tecnologías surge la imprenta digital, y con esta el servicio de impresión a través de la red electrónica desplegándose nuevos horizontes en el mundo de la impresión. El ahorro de tiempo y costos ofrecido por las nuevas técnicas digitales valen también para la salud, que se beneficia de la rapidez y amplias posibilidades que la impresión digital ofrece (10). Algunas de las ventajas que brinda son (11):

- Reproducción fiel del detalle de la imagen.
- Posibilidad de usar una gran variedad de papeles.
- Además, la impresión en offset es mucho más rápida ya que hay menor tiempo de preparación.

1.2. Sistemas Informáticos Internacionales.

Las impresiones de las imágenes médicas realizadas a un paciente luego de realizarse un estudio imagenológico, tienen gran importancia, ya que revelan el lugar donde radica el problema y posteriormente el especialista podrá realizar el análisis de las impresiones y emitir un diagnóstico al paciente. A continuación se describen algunos de los sistemas que realizan este proceso.

1.3. Internacionales.

Después de realizar una investigación a nivel internacional, se encontraron algunos sistemas informáticos que permiten realizar la impresión de las imágenes médicas en placa y en papel de manera eficiente. A continuación se mencionan algunos de estos sistemas.

1.3.1. APC (Advance Print Controller) DICOM:

El software VM MEDICAL WORKSTATION es un visualizador de estudios DICOM que tiene sus orígenes en Estados Unidos. Advance Print Controller es un módulo que forma parte de este sistema y su objetivo principal es realizar la impresión de las imágenes médicas. Permite armar la impresión tanto en película como en papel, según el formato más conveniente, pudiendo modificar la cantidad de filas, columnas, tamaño, zoom, brillo y contraste de cada una de las imágenes que van a ser impresas, permitiendo realizar combinaciones de estudios provenientes de distintos equipos de diagnóstico (12). Los tamaños disponibles en papel son 21x30cm y 30x41cm. Puede ser enviado a impresoras de film DICOM de cualquier marca o modelo (13).

1.3.2. Carestream DRYVIEW Chroma:

Este sistema fue desarrollado por la compañía Carestream Health que tiene sus orígenes en Estados Unidos. Proporciona la impresión de imágenes diagnósticas en películas y copias para remisión al especialista en papel, a todo color o en escala de grises. Tiene la certificación ENERGY STAR (14), otorgado por la Agencia de Protección Ambiental de los EE.UU. Permite combinar la impresión en película y papel en un solo sistema (15). Además necesita de un papel específico que se elabora en la misma compañía para la realizar la impresión (15).

1.3.1. DicomJet:

Este sistema fue desarrollado por la compañía NeoLógica y tiene sus orígenes en Italia. Permite imprimir radiografías y demás imágenes médicas en papel normal con una calidad muy cercana a la lograda por la película y una mejor calidad que la correspondiente al papel térmico. Puede ser usado por los sistemas PACS, permitiendo la impresión de imágenes directamente desde las modalidades médicas o desde Workstation (16). Permite ajustes de brillo, escala de grises y contraste y además recibe imagen desde cualquier modalidad médica y las imprime a través del controlador de las impresoras (17).

Después de realizar una búsqueda a nivel internacional, se puede determinar el alto grado de acabado y prestación de los sistemas que permiten la impresión de imágenes médicas en placas y en papel. No obstante, en su mayoría son sistemas privativos que requieren de la autorización de sus propietarios. Al mismo tiempo, una vez adquiridos, no implica que tengamos acceso a su código fuente, por lo que se hace casi imposible incorporar nuevas funcionalidades al sistema sin contar con la empresa que lo confecciona. Además suponen grandes costos de adquisición y mantenimiento y en su mayoría resultan casi imposibles de comprar por ser Cuba un país bloqueado. Todas estas limitantes dan paso a realizar la presente investigación en aras de resolver estas dificultades.

1.4. Nacionales.

En Cuba, también se han desarrollado algunos sistemas que permiten la impresión de las imágenes médicas en placa. Entre ellos se encuentran:

1.4.1. IMAGIS:

Imagis fue desarrollado en el Centro de Biofísica Médica de la Universidad de Oriente, que radica en Santiago de Cuba. Puede ser utilizado en la telemedicina y la transmisión de imágenes médicas multimodales. Permite la adquisición, procesamiento, almacenamiento, recuperación e impresión de imágenes médicas mediante el uso del protocolo estándar para la comunicación de imágenes digitales en medicina (DICOM 3.0), obtenidas en equipos de diferentes modalidades y fabricantes: Resonancia Magnética, Ultrasonido, Medicina Nuclear, Tomografía Axial Computarizada, Radiología Computarizada, entre otras (18).

1.4.2. Plataforma de Gestión de Imágenes:

Sistema desarrollado en la Universidad de las Ciencias Informáticas por algunos especialistas, estudiantes y profesores del departamento SWMI. Cuenta con un módulo que permite imprimir las imágenes médicas en placas, brindando la posibilidad al usuario de ajustar las impresiones de acuerdo a lo que él necesite. El especialista podrá instalar una nueva impresora, permite seleccionar el tipo de película, los tamaños de la misma, el destino, color del borde, la magnificación, polaridad, si la placa tendrá anotaciones o no, la cantidad de copias que desea realizar el usuario, chequear la cola de impresión, instalar nuevas impresoras, borrar imágenes y seleccionar la cantidad de imágenes que se mostrarán en cada página.

Al terminar la búsqueda a nivel nacional se concluyó que los sistemas existentes carecen de un módulo que permita realizar la impresión de las imágenes médicas en papel. Todas estas limitantes dan paso a realizar esta investigación en aras de resolver los problemas existentes. En los sistemas estudiados se evidencia el empleo del estándar DICOM para facilitar el intercambio de la información digital entre los dispositivos médicos.

1.5. Estándar DICOM.

Desde 1994 el estándar DICOM 3.0 del ACR- NEMA³, da el marco para la comunicación de imágenes entre equipos de Radiología (equipos de exploración, estaciones de trabajo, sistemas de archivo, redes de comunicación). Es el primer estándar instaurado en informática médica, y tiene el soporte de la mayoría de fabricantes (19). El alcance del estándar DICOM está contenido en cuatro áreas de aplicación general: la gestión de imágenes a través de la red, la gestión de impresión a través de la red, procesamiento de imágenes médicas y la gestión del almacenamiento local (20).

1.5.1. Ventajas del uso de DICOM.

- Une la información de la prueba de imagen con la información escrita dentro de un mismo archivo, lo que permite la identificación unívoca del paciente.
- Es posible integrarlo con el sistema de información de documentación clínica.

³ Comité conjunto del Colegio Estadounidense de Radiología y la Asociación Nacional de Fabricantes Eléctricos.

- La imagen se transfiere y almacena directamente desde el equipo de diagnóstico por imagen y el sistema de almacenamiento, lo que da lugar a un mejor aprovechamiento del tiempo de los expertos que pueden consultar la base de datos y brindar los servicios médicos optimizados a distancia.
- Permite implementar una política eficaz de copias de seguridad y registros de todos los estudios de imágenes realizados (21).

1.6. Tecnologías y herramientas.

Para la realización del componente se utilizan un conjunto de tecnologías, lenguajes, patrones y herramientas que permiten realizar un trabajo con mayor calidad y más eficiencia. A continuación se hará un estudio de las mismas:

1.6.1. Plataformas de Desarrollo.

En la actualidad existen disímiles plataformas de desarrollo de aplicaciones informáticas, entre las que se pueden mencionar Java Enterprise Edition (Java EE) (22) , la cual ofrece un conjunto de especificaciones y técnicas que proporcionan soluciones completas, seguras, estables y escalables para el desarrollo, despliegue y gestión de aplicaciones de múltiples niveles de funcionalidad basadas en servidores.

Oracle (23), es otra plataforma que brinda una gama más completa e integrada de herramientas para el desarrollo de aplicaciones. También se puede hacer mención de la plataforma .NET que es un entorno de ejecución y biblioteca de componentes independiente del lenguaje de programación utilizado, que admite la compilación y la ejecución de la siguiente generación de aplicaciones y servicios Web XML. Proporciona un entorno coherente de programación orientada a objetos, posee además un entorno de ejecución de código que reduce al mínimo el esfuerzo de implementación del software y los conflictos de versiones, basa toda la comunicación en estándares del sector para asegurar que el código de la plataforma se pueda integrar con otros tipos de código (24).

La plataforma .NET contiene dos componentes principales: el Entorno Común de Ejecución para Lenguajes (CLR por sus siglas en inglés) y la Biblioteca de Clases. Estos componentes proporcionan un entorno de ejecución administrado, de implementación simplificada y la integración con una amplia variedad de lenguajes de programación (25). .Net brinda otras ventajas como: código administrado, interoperabilidad

multilinguaje, seguridad de acceso al código y un gran despliegue, las cuales aseguran una mejor calidad en el trabajo que se realizará (26).

De todas las plataformas antes mencionadas, se utilizará para el desarrollo del componente la plataforma .Net, ya que facilitará la migración hacia una plataforma libre. Además de ser una tecnología definida por el CESIM, los proyectos desarrollados en el departamento están regidos por la misma y basan su arquitectura sobre la utilización de la misma.

Esta plataforma trabaja con las tecnologías Windows Presentation Foundation (WPF, por sus siglas en inglés) y Managed Extensibility Framework (MEF por sus siglas en inglés). WPF es una tecnología de Microsoft, presentada como parte de Windows Vista. Permite el desarrollo de interfaces de interacción en Windows tomando características de aplicaciones web. Ofrece una amplia infraestructura y potencia gráfica con la que es posible desarrollar aplicaciones visualmente atractivas, con facilidades de interacción que incluyen animación, video, audio y documentos. Además brinda elementos contenedores que implementan diseños de algoritmos de una manera que es completamente independiente del contenido que está sosteniendo. Separa, con el lenguaje declarativo XAML y los lenguajes de programación de .NET, la interfaz de interacción de la lógica del negocio, propiciando una arquitectura Modelo Vista Controlador para el desarrollo de las aplicaciones (27).

MEF es una biblioteca para crear aplicaciones ligeras y extensibles. Permite a los desarrolladores de aplicaciones detectar y utilizar extensiones sin requisitos de configuración. También brinda la opción de encapsular código con facilidad y evitar dependencias lógicas frágiles. MEF no solo permite reutilizar extensiones dentro de las aplicaciones sino también entre ellas (28), es una parte integral del Framework .NET 4.0 y puede utilizarse en aplicaciones cliente, que usen Windows Forms, WPF o cualquier otra tecnología y en aplicaciones servidor que usen ASP.NET.

Los elementos MEF especifican sus capacidades mediante declaración, son reconocibles en tiempo de ejecución, lo que significa que una aplicación puede utilizar elementos sin referencias incluidas en el código o archivos de configuración frágiles. Permite a las aplicaciones detectar y examinar elementos por sus metadatos, sin crear instancias ni cargar sus ensamblados, por lo que no hay necesidad de especificar qué extensiones se deben cargar y cuándo (29).

1.6.1. Lenguajes de Programación.

Existen varios lenguajes de programación, tal es el caso de PHP⁴ (30) que es un lenguaje de código abierto muy popular, adecuado para desarrollo web y que puede ser incrustado en HTML⁵. Entre las características fundamentales de este lenguaje se encuentran la integración con varias bibliotecas externas, permitiendo generar documentos en PDF (documentos de Acrobat Reader) hasta analizar código XML. Además ofrece una solución simple y universal para las paginaciones dinámicas de la web (31).

Además se encuentra Java (32) que presenta la ventaja de que es un lenguaje muy maduro y con una larga trayectoria. Entre sus características sobresalen, la ausencia de los punteros (la memoria es manejada automáticamente) y de las directivas del preprocesador, el soporte para la programación asincrónica y la utilización del enlace dinámico. Posee también un mecanismo robusto de manejo de excepciones.

Otro lenguaje que por ser el último en mencionar no significa que sea menos importante es C# 4.0, última versión del lenguaje, que mejora su interoperabilidad con otros lenguajes y tecnologías. Entre sus características se pueden mencionar la portabilidad del código, orientado a componentes, no admite funciones ni variables globales sino que todo el código y los datos han de especificarse dentro de definiciones de tipos de datos, consta de un editor de código completo, plantillas de proyecto, asistentes para código, eficiente, compatible, así como un depurador muy eficiente y fácil de usar (33).

Después de analizar dichos lenguajes de programación, se decide utilizar C# por ser el más adecuado para realizar el componente, además aunque es posible escribir código para la plataforma .NET en muchos otros lenguajes, C# es el único que ha sido diseñado específicamente para ser utilizado en ella, por lo que programarla usando C# es mucho más sencillo e intuitivo que hacerlo con cualquiera de los otros lenguajes antes mencionados. Es válido aclarar que además es el lenguaje que se definió por el CESIM. Se utilizará Visual Studio 2010, este es un IDE⁶ que soporta diversos grupos de lenguajes de programación tales como

⁴ **Hypertext Preprocessor (PHP)**, es un lenguaje de programación de uso general de código del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico.

⁵ **Hypertext Markup Language (HTML)**, es un estándar que sirve de referencia para la elaboración de páginas web en sus diferentes versiones, define una estructura básica y un código.

⁶ Entorno integrado de desarrollo.

Visual C++, Visual J#, Visual Basic y Visual C#. También cuenta con poderosas herramientas que son de gran efectividad para administrar proyectos y proteger el código fuente. A la vez cuenta con un editor que soporta resaltado de sintaxis y completamiento de código. Permite realizar tareas de modelado, codificación, prueba y depuración, sin salir del entorno, brindando alta eficiencia y productividad (34).

Esta versión es hasta la fecha la más completa y está acompañada por la plataforma .NET 4.0, integra herramientas para el desarrollo de aplicaciones para Windows 7 y utiliza la tecnología WPF. Cuenta con la capacidad de utilizar múltiples monitores, así como la posibilidad de desacoplar las ventanas de su sitio original y acoplarlas en otros sitios de la interfaz de trabajo (34).

Incluye nuevos productos para entornos de pruebas y control de calidad, como son Visual Studio Test Professional 2010: conjunto especializado de herramientas que ayudan a asegurar una calidad profesional en la planificación y ejecución de pruebas. Las funcionalidades para la gestión del ciclo de vida de las aplicaciones (ALM), garantizan crear soluciones de calidad y a la medida, a la vez que se reducen el costo y el tiempo de su desarrollo, independientemente del tamaño del equipo (34).

1.7. Herramientas para el Modelado.

Para llevar a cabo el modelado de las aplicaciones, existen diferentes herramientas que permiten que el trabajo de los analistas sea más rápido y eficiente, entre estas se destacan Visual Paradigm para UML (35), que es una herramienta para desarrollo de aplicaciones utilizando modelado UML, ideal para Ingenieros de Software, Analistas y Arquitectos de Sistemas.

Además se encuentra Enterprise Architect (EA por sus siglas en inglés) (36), es una herramienta gráfica basada en Windows que permite modelar y gestionar información compleja, diseñar y visualizar software, o construir y desplegar diversos sistemas. Además facilita la creación y actualización del código fuente de los elementos del modelo UML en muchos lenguajes de programación, así como el código fuente de importación existente en los elementos del modelo. Abarca por completo el ciclo de vida de desarrollo de software, con herramientas que le proporcionan una infraestructura enormemente competitiva en torno al modelado de negocio, diseño de software, ingeniería de sistemas, arquitectura corporativa, gestión de requerimientos y pruebas. Otras de sus características son:

- Es multi-usuario.
- Cuenta con una producción flexible.
- Documentación de alta calidad.

- Permite la generación de documentos y herramientas de informes.
- Soporta la generación e ingeniería inversa de código fuente.
- Permite visualizar las aplicaciones.

Gracias a todas sus facilidades se decide utilizar para el desarrollo del componente, la herramienta para el modelado EA en su versión 7.5, además de ser otra de las herramientas definidas por el CESIM.

Se usará el lenguaje para el modelamiento unificado (UML por sus siglas en inglés) se utiliza para la especificación, visualización, construcción y documentación de los artefactos de un proceso de sistema intensivo. Este lenguaje unificado cuenta con una notación estándar y semántica, esencial para el modelado de un sistema orientado a objetos. La estandarización de un lenguaje de modelado es invaluable, ya que es la parte principal del proceso de comunicación que requieren todos los agentes involucrados en un proyecto informático. (37).

UML permite:

- Especificar cuáles son las características de un sistema antes de su construcción.
- Construir sistemas diseñados a partir de modelos especificados.
- Visualizar gráficamente un sistema de manera que otros puedan entenderlo.
- Documentar los elementos gráficos del sistema desarrollado para futuras revisiones.
- Verificar y validar el modelo realizado.
- Generar código a partir de los modelos y viceversa.

Como metodología de desarrollo se utilizará el Proceso Unificado de Desarrollo conocido como (RUP por sus siglas en inglés), es una metodología de desarrollo de software que proporciona prácticas recomendadas probadas y una arquitectura configurable. RUP mantiene al equipo de desarrollo enfocado en producir incrementalmente software operativo a tiempo, con las características y calidad requeridas (38).

Es un proceso iterativo e incremental. Organiza el desarrollo para poder hacer planificaciones y definir hitos que posibiliten controlar el progreso del proceso de desarrollo del software. Constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos (38).

Una de las mejores prácticas centrales de RUP es la noción de desarrollar iterativamente. Rational Unified Process organiza los proyectos en términos de disciplinas y fases, consistiendo cada una, en una o más

iteraciones. Con esta aproximación iterativa, el énfasis de cada flujo de trabajo variará a través del ciclo de vida. La aproximación iterativa ayuda a mitigar los riesgos en forma temprana y continua, con un progreso demostrable y frecuentes releases ejecutables (38).

Se utilizó RUP, porque permitió la creación de todos los artefactos y plantillas que se utilizaron para tener un mejor entendimiento del componente que se desea desarrollar.

El Sistema de Control de Versiones (SVN) 1.7 será utilizado por ser un software libre desarrollado bajo una licencia de tipo Apache/BSD, usado para que un equipo de desarrollo pueda trabajar en un mismo proyecto en forma organizada accediendo al repositorio a través de la red.

Se utiliza Tortoise SVN como cliente de Subversion, ya que es un software para la revisión y control de Versiones/software de control de código fuente para Windows. No está integrado a ningún IDE específico como Visual Studio, Eclipse u otros, y se puede utilizar con cualquier herramienta de desarrollo (39). Tortoise SVN provee una interfaz de usuario agradable y fácil para Subversion. Se utiliza TortoiseSVN como cliente de Subversion por ser un software libre bajo la licencia GNU GPL⁷ lo que indica que es completamente gratis, incluyendo su código fuente. (40)

1.8. Modelos de Calidad.

Entre los modelos de calidad existentes a nivel mundial se pueden mencionar ISO/IEC 15504 (ISO 2004) (41) que proporciona un marco de trabajo para la evaluación de los procesos. Otro es Capability Maturity Model Integration (CMMI por sus siglas en inglés), es un modelo para la mejora o evaluación de los procesos de desarrollo y mantenimiento de sistemas y productos de software. Desarrollado por el SEI⁸ y publicado en su primera versión en enero de 2002. Es empleado para guiar las mejoras de procesos durante el desarrollo de un proyecto, un departamento o hasta una organización. CMMI se desarrolló para facilitar y simplificar la adopción de varios modelos de forma simultánea. Mide la madurez del desarrollo de software en una escala del 1 al 5. Integra disciplinas como sistemas y software en un solo marco de trabajo. Describe formas efectivas y probadas de hacer las cosas, no es un enfoque radica. (42).

⁷ Licencia Pública General.

⁸ Instituto de Ingeniería del Software de la Universidad Carnegie Mellon.

1.8.1. Proceso de Mejoras.

El CESIM certificó el nivel 2 correspondiente a la representación escalonada de CMMI. Una organización CMMI Nivel 2 asegura que sus procesos son planeados, documentados, realizados, monitoreados y controlados a nivel de proyectos. Los requerimientos, estándares y objetivos para los procesos, sus productos de trabajo y sus servicios son definidos y documentados. El estado de los productos es visible para la administración en puntos de control preestablecidos (43).

Está compuesto por 7 Áreas de Proceso: Gestión de Requisitos (REQM), Planificación de Proyectos (PP), Seguimiento y Control de Proyectos (PMC), Acuerdos con Proveedores (SAM), Medición y Análisis (MA), Aseguramiento de la Calidad de Procesos y Productos (PPQA) y Gestión de la Configuración (CM) (43).

El propósito del área de proceso REQM es gestionar los requisitos de los productos del proyecto y componentes del producto e identificar inconsistencias entre dichos requisitos y la planificación del proyecto y los productos de trabajo (43).

El proceso gestiona todos los requisitos recibidos o generados por el proyecto, incluyendo tanto los requisitos técnicos como los no técnicos, así como aquellos requisitos añadidos al proyecto por la organización (43).

Posee un objetivo específico (SG por sus siglas en inglés) y cinco prácticas específicas (SP por sus siglas en inglés): SG1 Administrar los requisitos, SP 1.1 Obtener el entendimiento de los requisitos, SP 1.2 Obtener el compromiso a los requisitos, SP 1.3 Administrar los cambios a los requisitos, SP 1.4 Mantener la trazabilidad bidireccional de los requisitos y SP 1.5 Identificar inconsistencias entre el trabajo del proyecto y los requisitos (43).

1.9. Conclusiones del capítulo.

Se realizó un análisis del estado del arte y de las tendencias actuales en los sistemas informáticos que posibilitan la impresión de las imágenes médicas, donde se concluye que los mismos no representan soluciones al problema a resolver, evidenciándose la necesidad de realizar un componente que permita la impresión de estas imágenes en papel. Además el estudio previo de las tecnologías permitió escoger la metodología RUP para guiar el proceso de desarrollo del nuevo componente. Se hará uso de la herramienta Enterprise Architect 7.5 para modelado y como lenguaje de modelado se empleará UML. El framework seleccionado fue .NET Framework 4.0 para la creación del sistema en cuestión. El IDE de desarrollo que se utilizará es Visual Studio 2010.

CAPÍTULO 2. CARACTERÍSTICAS DEL COMPONENTE PARA LA IMPRESIÓN CONVENCIONAL DE IMÁGENES MÉDICAS.

En este capítulo se presenta la propuesta de solución para la situación problemática. Se muestran los procesos del negocio mediante el modelo de dominio, las características y funcionalidades que tendrá el mismo, basado en los requisitos funcionales y no funcionales. Se realizará el diagrama de casos de uso del sistema, las especificaciones de los casos de uso asociados al componente y la trazabilidad de los casos de uso con sus correspondientes requisitos funcionales.

2.1. Breve descripción del componente.

El sistema Visor de Imágenes Médicas cuenta con un módulo que permite la impresión de dichas imágenes en placas. Actualmente, en Cuba se realiza la impresión de las imágenes médicas solamente en placas, este proceso resulta muy lento, costoso y además los productos que se emplean para la realización de las mismas son dañinos para la salud del personal médico que realiza este trabajo y provocan la contaminación del medio ambiente, por lo que se propone el desarrollo de un componente que permita seleccionar la opción de realizar la impresión de las imágenes médicas en papel.

El componente de la impresión en papel, permitirá cargar una o varias imágenes médicas, seleccionar el tipo de papel que utilizará para la impresión, instalar una nueva impresora, la cantidad de copias que desea realizar, elegir la cantidad de imágenes que desea mostrar por cada página, revisar la cola de impresión, si desea que la impresión tenga o no anotaciones y por último imprimir. Permitirá a los médicos poder conferenciar con otros miembros del hospital los resultados obtenidos y añadir al papel nuevas observaciones. También realizará el proceso de obtención de la imagen en papel de manera más ágil y menos engorrosa, ya que se economizará en tiempo y dinero.

A continuación se establece una comparación de la impresión en placa y la impresión en papel de las imágenes médicas atendiendo a los parámetros precio, dinero y tiempo invertido y se recogieron datos a partir de una entrevista realizada en el Hospital Frank País al Licenciado en Imagenología y Especialista en los TAC⁹, Roberto Miranda Barrabia, ver el [Anexo 1](#), el cual indicó que se realizaban 200 placas diarias

⁹ Tomografía Axial Computarizada.

como promedio y el tiempo de impresión de una placa es 15 min aproximadamente. El costo de una placa es \$28.07 por tanto se gastarían diariamente \$5 614, y se invertiría un tiempo 33h. Conociendo que el tiempo que demora la impresión de una imagen médica en papel es como máximo de 1 min. Al realizar 200 impresiones, con un precio de 9¢ la hoja de papel, se invierte en dinero \$1 800 y un tiempo de 3h. Por lo que se obtiene una diferencia de \$3 814 de dinero y 30h invertidas en este proceso. Por tanto es más factible el uso de la impresión en papel, siempre y cuando la calidad de la imagen sea la indicada para emitir un diagnóstico.

Opción	Precio	Dinero invertido	Tiempo invertido
Placa	\$28.07	\$5 614	33h
Papel	9 ¢	\$1 800	3h
Diferencia		\$3 814	30h

Tabla 1. Comparación de las impresiones.

2.2. Modelo de Dominio.

En la primera fase de desarrollo de RUP, este define la realización del modelo de negocio, cuyo objetivo está en comprender el entorno del cliente y detectar las mejoras en los procesos de la organización. El modelo de dominio (44) es un subconjunto del modelo de negocio, utilizado cuando no es posible identificar claramente los procesos del negocio, en este se capturan los tipos de objetos más importantes del sistema. El modelo de dominio se describe utilizando el lenguaje de modelado UML mediante diagrama de clases.

Teniendo en cuenta que en la aplicación que se desea desarrollar no se encuentran bien definidos los procesos del negocio se propone construir un modelo de dominio, el cual proporciona una vista del sistema donde se representan los conceptos fundamentales implicados en el proceso de impresión de las imágenes.

Un Modelo de Dominio es una representación visual de clases conceptuales o de objetos reales en un dominio de interés. Consiste en un conjunto de diagramas, sin definición de operaciones. El propósito principal de este modelo es generar una terminología común y sentar las bases del entendimiento del desarrollo y no para definir el sistema completo (44). Para ver la descripción de los componentes que conforman el modelo de dominio ver la Tabla 2 y la figura 2.

Clases	Descripción
Especialista	Es la persona que interactúa con el sistema, puede ser un médico o cualquier otra persona especializada en el tema. Es el encargado de seleccionar las imágenes que desea imprimir y además podrá realizar cambios en ellas.
Imagen	Es el objeto que se visualiza en la pantalla y se desea imprimir.
Visualizador	Es la herramienta o programa donde se visualizarán las imágenes que se desean imprimir. En él se podrán realizar cambios a estas imágenes que serán impresas.
Impresora	Representa el hardware en el que se realizarán las actividades de impresión.

Tabla 2. Descripción del Modelo de Dominio.

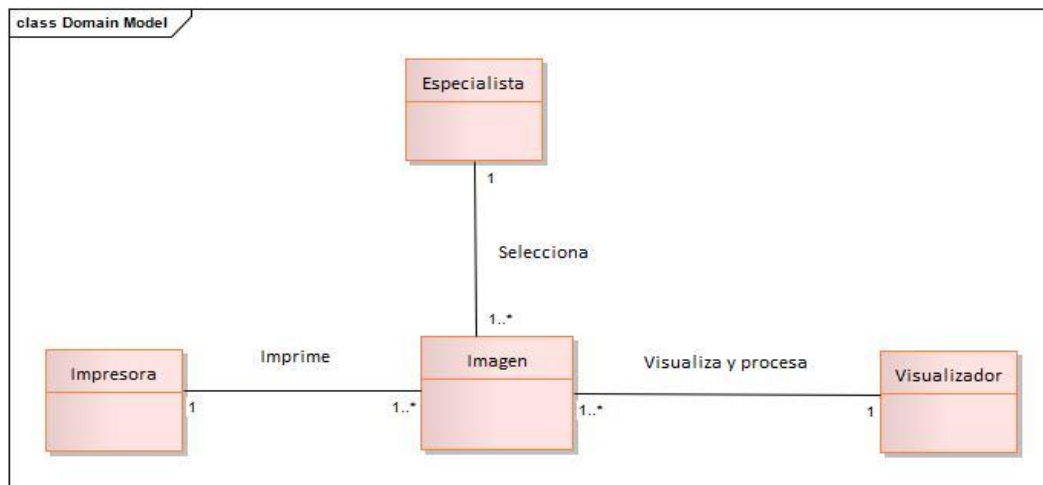


Figura 2. Diagrama de Modelo de Dominio.

2.3. Especificación de los Requisitos del Software.

2.3.1. Requisitos Funcionales.

Los requisitos funcionales definen los servicios que el sistema debe proporcionar, cómo debe reaccionar a una entrada particular y cómo se debe comportar el sistema ante situaciones particulares. Deben de ser sencillos para que puedan ser comprendidos fácilmente por los clientes y desarrolladores (45). En la tabla 3 y la figura 3 están representados los requisitos funcionales.

Requerimiento	Descripción
RF 1. Adicionar impresora.	Permite adicionar una nueva impresora a la lista de impresoras. Para ello se debe seleccionar el método de instalación y especificar la dirección de la impresora.
RF 2. Modificar impresora.	Permite modificar los datos de la impresora seleccionada.
RF 3. Eliminar impresora.	Permite eliminar la impresora seleccionada de la lista de impresoras.
RF 4. Seleccionar impresora.	Permite seleccionar el tipo de impresora que se desea utilizar. Estas pueden ser impresoras físicas (convencionales) o impresoras virtuales (OneNote, PDF, XPS y Fax).
RF 5. Impresoras.	Permite la configuración de una nueva impresora, añadiéndola a la lista de impresoras. Para ello se debe seleccionar el método de instalación y especificar la dirección de la impresora.
RF 6. Mostrar cola de impresión.	Permite acceder a la lista de las imágenes pendientes de la impresión.
RF 7. Seleccionar opciones de papel.	Permite seleccionar los parámetros de presentación que son comunes para todas las imágenes en el papel: Tamaño y Orientación. Se puede seleccionar además el número de mosaicos en los que se presentarán las imágenes a imprimir y también si la impresión tendrá algunas anotaciones.
RF 8. Especificar número de copias.	Permite seleccionar la cantidad de copias a realizar por la impresora.
RF 9. Seleccionar imágenes a imprimir.	Permite seleccionar las imágenes que se desean imprimir en el papel.

RF 10. Borrar imágenes seleccionadas.	Permite borrar imágenes de la selección a imprimir.
RF 11. Imprimir.	Permite la impresión de las imágenes médicas.
RF 12. Adicionar imagen.	Permite adicionar una imagen a la lista de imágenes a imprimir.
RF 13. Adicionar serie.	Permite añadir las imágenes de una serie a la lista de imágenes a imprimir.

Tabla 3. Requisitos Funcionales.

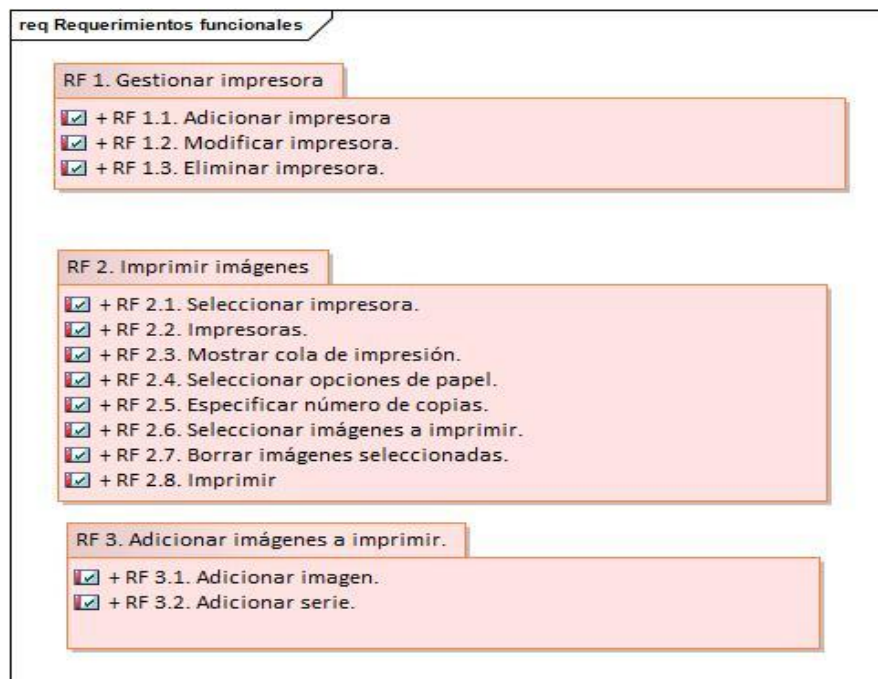


Figura 3. Diagrama de Requisitos Funcionales.

2.3.2. Requisitos No Funcionales.

Los requisitos no funcionales son restricciones que afectan a los servicios o funciones del sistema, tales como restricciones de tiempo, sobre el proceso de desarrollo, estándares, entre otros. Definen propiedades emergentes del sistema, tales como el tiempo de respuesta, las necesidades de almacenamiento, la fiabilidad y otras (45).

En la tabla 4 y la figura 4 se muestra la descripción de los requisitos no funcionales de la aplicación. Además en el [Anexo 2](#) se muestran los prefijos de los requisitos no funcionales.

Requerimiento	Descripción
RNU 1. Facilidad de uso.	El sistema debe ser sencillo para facilitar el uso de los usuarios sin experiencia.
RNU 2. El sistema debe ser configurable.	El sistema debe permitir al usuario hacer la configuración que desee.
RNU 3. La configuración debe poderse preservar de una impresión a otra.	El sistema debe mantener la configuración realizada por el usuario de una impresión a otra.
RNU 4. Iconografía presente en la interfaz de la aplicación.	Los íconos del sistema deben tener relación con lo que hace este para que facilite el trabajo con el mismo.
RNDI 1. Lenguaje de programación C# 4.0.	Se usa el lenguaje de programación C#, en su versión 4.0.
RNDI 2. Uso de Tecnología WPF.	Se usa la Tecnología WPF (Windows Presentation Foundation).
RNDI 3. Uso de Framework.Net 4.0.	Se usa la plataforma de desarrollo .Net, con Framework 4.0.
RNDI 4. Uso de Enterprise Architect 7.5.	Se usa Enterprise Architect como herramienta para el modelado.
RNDI 5. Uso de Microsoft Visual Studio 2010.	Se utilizará Microsoft Visual Studio 2010 como entorno de desarrollo integrado.
RNDI 6. Uso de Lenguaje de Modelado Unificado.	Se utilizará el Lenguaje de Modelado Unificado o UML en su versión 2.0.
RNDI 7. Uso de tecnología MEF.	Se utilizará la tecnología Managed Extensibility Framework o MEF para cargar los plugins en tiempo de ejecución.
RNDI 8. Uso del patrón arquitectónico MVVM.	Se usa MVVM como patrón de diseño.
RNDI 9. Uso de RUP.	Se utilizará RUP como metodología de desarrollo.
RNFO 1. Memoria RAM mínimo 1GB.	Se necesita una estación de trabajo con una memoria RAM

	mayor o igual a 1GB.
RNFO 2. Debe ser una impresora convencional y estar conectada a la red.	Se necesita al menos una impresora convencional en la red para poder realizar la impresión.
RNFO 3. Sistema Operativo Windows XP Service Pack 3 o superior.	Se utilizará el Sistema operativo Windows XP Service Pack 3 o superior.
RNFO 4. Impresora de alta resolución.	La impresora debe ser de alta resolución para que las imágenes tengan una mejor calidad.
RNEA 1. Se utilizará el estándar DICOM 3.0.	Este estándar se utilizará para la comunicación de las imágenes.
RNE 1. El tiempo de respuesta debe coincidir con las funcionalidades ejecutadas siendo este el mínimo.	El sistema debe ejecutar las acciones en el menor tiempo posible.
RNIU 1. Deben prevalecer colores agradables a la vista.	El sistema debe tener colores agradables y amigables a la vista del usuario.
RNIU 2. Interfaz gráfica familiar a la de la plataforma de gestión de imágenes.	La interfaz gráfica debe ser familiar a la de la plataforma de gestión de imágenes ya que este componente será un producto que formará parte del mismo.

Tabla 4. Requisitos no funcionales.

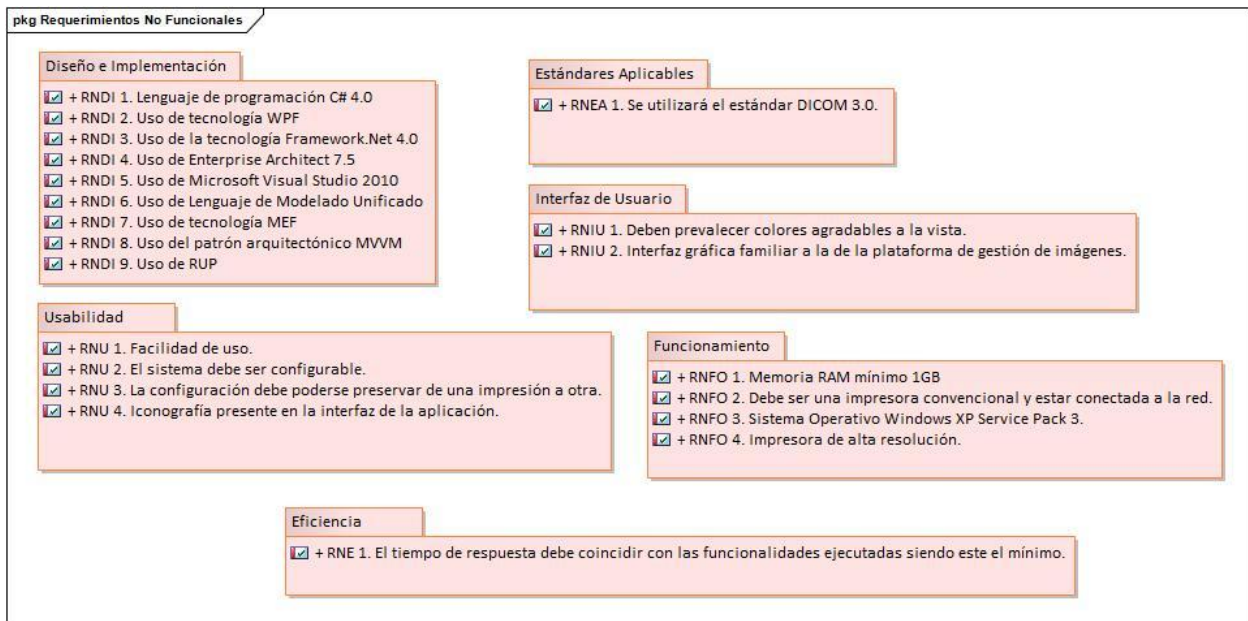


Figura 4. Diagrama de Requisitos No Funcionales.

2.4. Definición de los actores del sistema.

Los actores del sistema son aquellas personas que interactúan o hacen uso del sistema. La siguiente tabla muestra el actor del sistema y su descripción.


Actor	Objetivo
	Persona encargada de la configuración de la impresora y de realizar el proceso de configuración para que imprima lo que se desea.

Tabla 5. Actor del sistema.

2.5. Diagrama de casos de uso del sistema.

Los diagramas de casos de uso documentan el comportamiento de un sistema desde el punto de vista del usuario. Determinan los requisitos funcionales del sistema, es decir, representan las funciones que un sistema puede ejecutar. Su ventaja principal es la facilidad para interpretarlos, lo que hace que sean especialmente útiles en la comunicación con el cliente (46). En la siguiente figura se muestran los casos de uso del sistema.

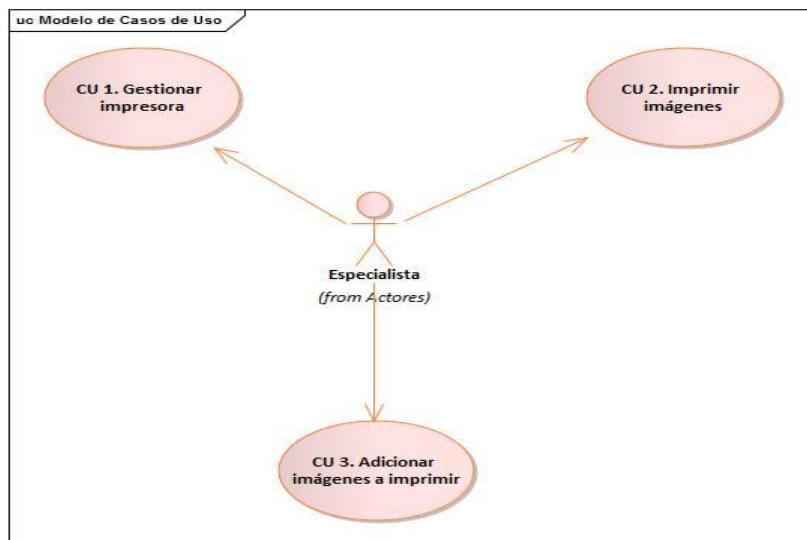


Figura 5. Diagrama de Casos de Usos.

2.6. Descripción de los casos de uso del sistema.

Las siguientes tablas muestran una breve descripción de los casos de uso del sistema. La descripción ampliada de los mismos se puede observar en el [Anexo 3](#).

CU # 1. Gestionar impresora	
Objetivo	Gestionar las impresoras a las que podrá conectarse el sistema. De esta forma se podrá adicionar, editar y eliminar impresoras.
Actores	Especialista (Inicia)
Resumen	El caso de uso inicia cuando el Especialista selecciona la opción Configuración / Impresoras NO DICOM; el sistema muestra las impresoras registradas y las opciones para adicionar, modificar o eliminar alguna impresoras.
Complejidad	Media
Prioridad	Alta
Referencias	RF 1.1, 1.2, 1.3.
Precondiciones	No aplica
Postcondiciones	Se agregaron los nuevos datos de la impresora. Se listaron las impresoras registradas. Se modificaron los valores de las impresoras. Se eliminó la impresora seleccionada.

Tabla 6. Resumen del Caso de Uso “Gestionar Impresora”.

CU # 2. Imprimir imágenes.	
Objetivo	Seleccionar el tipo de impresora a utilizar, verificar la disponibilidad de la misma y acceder a la lista de los estudios pendientes de impresión. Seleccionar las diferentes opciones de papel, la cantidad de copias a realizar, las imágenes que se desean imprimir en el papel y de permitir borrar imágenes de la selección a imprimir.
Actores	Especialista (Inicia)
Resumen	El caso de uso inicia cuando el actor selecciona la opción Impresión en papel ubicada en la pestaña Archivo, el sistema muestra los atributos que se deben seleccionar para imprimir las imágenes.
Complejidad	Baja
Prioridad	Alta

Referencias	RF 2.1, 2.2, 2.3, 2.4, 2.5, 2.6, 2.7, 2.8.
Precondiciones	Deben existir impresoras registradas en la aplicación.
Postcondiciones	Se seleccionó la impresora deseada. Se verificó la conexión de la impresora. Se mostró la cola de impresión. Se seleccionaron las opciones de papel. Se especificó el número de copias a imprimir. Se seleccionaron las imágenes a imprimir. Se borraron las imágenes deseadas.

Tabla 7. Resumen del Caso de Uso “Imprimir imágenes”.

CU # 3. Adicionar imágenes a imprimir.	
Objetivo	Adicionar una imagen o una serie de imágenes que serán enviadas a la vista de la impresión.
Actores	Especialista (Inicia)
Resumen	El caso de uso inicia cuando el especialista selecciona la pestaña Principal y la opción Imprimir en papel, el sistema muestra las opciones que permiten adicionar la serie o imagen.
Complejidad	Baja
Prioridad	Alta
Referencias	RF 3.1, 3.2.
Precondiciones	Para adicionar imágenes o series a imprimir deben encontrarse las mismas cargadas en el visor.
Postcondiciones	Se adicionó una serie. Se adicionó una imagen.

Tabla 8. Resumen del Caso de Uso “Adicionar imágenes a imprimir”.

2.7. Conclusiones del capítulo.

A partir del análisis del modelo de dominio, teniendo en cuenta las características y actividades que se llevan a cabo en el contexto que se desea automatizar, se determinaron las mejoras que se necesitan para el desarrollo del software. Fueron descritos los casos de usos vinculados a los requisitos funcionales identificados y se definió la distribución de los mismos por ciclos de desarrollo, logrando así una mejor distribución y organización de la implementación. La definición de los requerimientos no funcionales brindó un mejor enfoque de las características físicas y técnicas que debe cumplir la aplicación para lograr su buen funcionamiento.

CAPÍTULO 3. ANÁLISIS Y DISEÑO DEL COMPONENTE PARA LA IMPRESIÓN CONVENCIONAL DE IMÁGENES MÉDICAS.

En este capítulo se muestra la arquitectura del sistema, los diagramas de clases del diseño y los diagramas de secuencia según los casos de uso del sistema definidos en el capítulo anterior, así como una breve descripción de los casos de uso arquitectónicamente significativos y los patrones de diseño utilizados.

3.1. Modelo arquitectónico.

La arquitectura de software consiste en un conjunto de patrones y abstracciones coherentes que proporcionan el marco de referencia necesario para guiar la construcción del software para un sistema de información. Tiene como misión principal la identificación de requerimientos críticos en la estructura del sistema y la reducción de riesgos asociados a la construcción del mismo. Debe orientarse a las necesidades del cliente y ser flexible en cuanto a futuras modificaciones tanto de software como de hardware evitando que efectos inesperados puedan tener repercusión en el sistema. Constituye un modelo relativamente pequeño y asequible por la vía intelectual sobre cómo está estructurado el sistema y la forma en que sus componentes trabajan juntos (47).

El sistema propuesto utilizará como patrón arquitectónico principal Model View ViewModel para conectar la interfaz de usuario principal, con las clases que representan el modelo de datos y además posee las funcionalidades necesarias para su manipulación. El mismo define la interfaz de usuario de forma declarativa y usa el marcado de enlace de datos para vincularla a otras capas que contengan datos y comandos de usuario. La infraestructura de enlace de datos proporciona un acoplamiento débil que mantiene sincronizados la interfaz de usuario y los datos vinculados, y que enruta todas las entradas de usuario a los comandos apropiados. Además organiza el código de tal forma que es posible cambiar partes individuales sin que los cambios afecten a las demás partes. Esto presenta numerosas ventajas, como las siguientes (48):

- Permite un estilo de codificación exploratorio e iterativo.
- Simplifica las pruebas unitarias.
- Permite aprovechar mejor herramientas de diseño.
- Admite la colaboración en equipo.

Cuando se usa el patrón MVVM, una aplicación se divide en las siguientes capas (48):

- La capa de **modelo** incluye todo el código que implementa la lógica principal de la aplicación y define los tipos requeridos para modelar el dominio de la aplicación. Esta capa es completamente independiente de las capas de vista y modelo de vista.
- La capa de **vista** define la interfaz de usuario que utiliza marcado declarativo. El marcado de enlace de datos define la conexión entre componentes específicos de la interfaz de usuario y diversos miembros de modelo de vista (y, en ocasiones, de modelo).
- La capa de **modelo de vista** proporciona destinos de enlace de datos para la vista. En muchos casos, el modelo de vista expone el modelo directamente o proporciona miembros que encapsulan miembros de modelo específicos. El modelo de vista también puede definir miembros para realizar un seguimiento de los datos que son relevantes para la interfaz de usuario pero no para el modelo, como el orden de visualización de una lista de elementos.

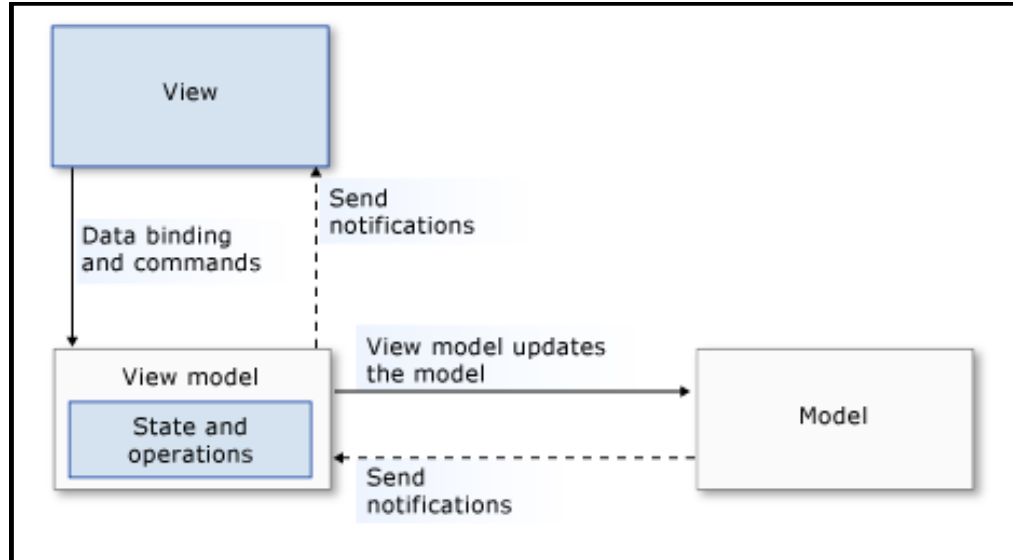


Figura 7. Funcionamiento del patrón MVVM.

3.2. Diagrama de clases.

Las clases son el componente más importante en el paradigma de la programación orientada a objetos, ya que permiten crear tipos personalizados propios mediante la agrupación de variables de otros tipos, métodos y eventos. Una clase es como un plano, define los datos y el comportamiento de un tipo. La descripción de las clases brinda información sobre la interacción de las clases involucradas en la realización de los casos de uso arquitectónicamente significativos (49). Las mismas se especifican en el [Anexo 4](#) junto a la descripción de las clases principales.

3.3. Diseño.

En esta etapa se transforman los requerimientos en un modelo de diseño del sistema a implementar, además se define la arquitectura que debe tener el sistema y se adapta este a un entorno de implementación para que sea eficiente y seguro. Es un modelo físico y concreto. Se centra en cómo los requisitos funcionales y no funcionales tienen impacto en el sistema a desarrollar. Dentro de sus propósitos están: crear una entrada apropiada y un punto de partida para la implementación, descomponer los trabajos de implementación en partes más manejables que puedan ser llevadas a cabo por diferentes equipos de desarrollo. Tiene como objetivo principal la elaboración de los diagramas de clases de diseño, que muestran las clases participantes en la realización de un caso de uso con todos sus atributos (50).

3.4. Diagramas de secuencia.

Los diagramas de clases son vistas estáticas del modelo. Los diagramas de interacción son dinámicos y describen cómo los objetos colaboran entre sí.

Un diagrama de secuencia (51) es un diagrama de interacción que detalla cómo las operaciones se llevan a cabo, los mensajes que se envían y cuándo. Los diagramas de secuencia se organizan en función del tiempo. El tiempo avanza a medida que avanza el diagrama. Los objetos involucrados en la operación son mostrados de izquierda a derecha acorde a cuando toman parte en la secuencia de mensajes.

El actor es quien inicia el ciclo interactuando inicialmente con la interfaz de usuario; luego se inician todos los objetos que intervienen en el funcionamiento de la aplicación. En el [Anexo 5](#) se pueden observar los diagramas de secuencias por casos de usos del sistema.

3.5. Patrones de Diseño.

Los patrones de diseño (design patterns) (52) son la base para la búsqueda de soluciones a problemas comunes en el desarrollo de software y otros ámbitos referentes al diseño de interacción o interfaces.

Un patrón de diseño es una solución a un problema de diseño. Para que una solución sea considerada un patrón debe poseer ciertas características. Una de ellas es que debe haber comprobado su efectividad resolviendo problemas similares en ocasiones anteriores. Otra es que debe ser reusable, lo que significa que es aplicable a diferentes problemas de diseño en distintas circunstancias.

Según su enfoque los patrones de diseño se agrupan en:

- Patrones de creación (o creacionales).
- Patrones estructurales.
- Patrones de comportamiento.
- Patrones de sistema.

Para el diseño del componente propuesto se hace uso de algunos de los patrones existentes. A continuación se describen los patrones que serán utilizados en el diseño del sistema.

Observador (Observer): Es un patrón de comportamiento. Tiene un efecto lateral muy frecuente en aquellos sistemas que se fragmentan en un conjunto de clases que cooperan es la necesidad de mantener la consistencia entre los distintos objetos interrelacionados. Para no recurrir a soluciones fuertemente acopladas (que reducen la posibilidad de reutilización), este patrón define una dependencia “uno-a-muchos” entre objetos, para que, cuando uno de ellos cambie su estado, todos los que dependan de él sean avisados y puedan actualizarse convenientemente.

Solitario (Singleton): Es un patrón de creación. Es uno de los patrones de diseño más sencillo, es útil para limitar el máximo número de instancias de una clase en exactamente solo una. En este caso, si más de un objeto necesita utilizar una instancia de la clase Singleton, esos objetos comparten la misma instancia de la clase Singleton. En un uso más avanzado, este patrón puede ser utilizado también para administrar n instancias de una clase.

Su objetivo es garantizar que la clase NDPrinter sólo tenga una única instancia, proporcionando un punto de acceso global a la misma. De esta forma todos los objetos que hagan uso de esta clase utilizarán la misma instancia.

Orden (Command): Es un patrón de comportamiento. Tiene como objetivo encapsular una operación en un objeto, permitiendo ejecutar dicha operación sin necesidad de conocer el contenido de la misma. La clase PrintVM implementa en comando y lo expone. El control PrintNonDicomControl invoca el comando sin saber detalles de su implementación como respuesta a la ocurrencia de un evento.

3.6. Conclusiones del capítulo.

A través de la realización de este capítulo se concluye que la descripción de las clases más significativas brindó una mayor comprensión de sus atributos y métodos. Los diagramas de clases del diseño mostraron las relaciones entre las distintas clases, logrando así una estructura adecuada para el desarrollo del software. Con el empleo del patrón arquitectónico Modelo Vista- Vista Modelo, se logró una mayor flexibilidad y robustez del componente.

CAPÍTULO 4. IMPLEMENTACIÓN DEL COMPONENTE PARA LA IMPRESIÓN CONVENCIONAL DE IMÁGENES MÉDICAS.

En este capítulo se procede a la realización del flujo de implementación, con el objetivo de especificar las principales características que contendrá el componente propuesto. En este capítulo se realiza la elaboración del diagrama de componentes, así como el diagrama de despliegue con el objetivo de conocer la distribución física de la solución implementada, todo ello como parte del modelo de implementación.

4.1. Diagrama de componentes.

Un Diagrama de Componente es, como su nombre lo indica, un esquema o diagrama que muestra las interacciones y relaciones de los componentes de un modelo. Entendiéndose como componente a una clase de uso específico, que puede ser implementada desde un entorno de desarrollo, ya sea de código binario, fuente o ejecutable; dichos componentes poseen tipo, que indican si pueden ser útiles en tiempo de compilación, enlace y ejecución. Este tipo de diagrama se representa mediante componentes unidos mediante relaciones de dependencia (generalmente de compilación) (53).

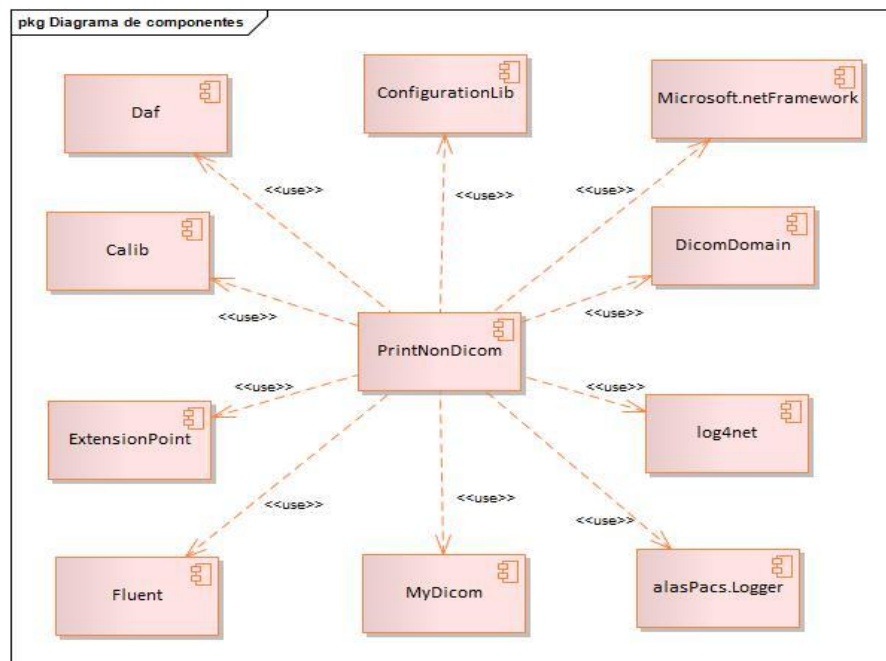


Figura 7. Diagrama de Componentes.

4.2. Descripción de los componentes.

- **ExtensionPoint:** Contiene los contratos que definen la interacción entre el sistema y las partes extensibles del mismo. Se empleó para subscribirse a los cambios de idiomas que el sistema sufra, de tal forma que los textos que se muestran en las descripciones y títulos de las mediciones cambien en correspondencia con el lenguaje del sistema.
- **Daf:** Librería de herramientas para el desarrollo de aplicaciones utilizando WPF, manipulación de eventos y referencias débiles.
- **Calib:** Librería para la lectura y procesamiento de las imágenes DICOM.
- **DicomDomain:** Librería que gestiona la información representada por el control de visualización.
- **log4net:** Librería que se emplea para registrar los sucesos que ocurren en el sistema.
- **MyDICOM:** Librería que se utiliza para realizar el almacenamiento, la lectura y escritura de las imágenes.
- **Fluent:** Librería que permite mejorar la interfaz gráfica del sistema.
- **Configurationlib:** Librería de enlace Dinámico. Esta librería puede ser cargada y ejecutada en cualquier proceso en ejecución.
- **alasPACS.Logger:** Un log es un mensaje que emite el sistema para su funcionamiento. Esta librería permite gestionar los log.
- **Microsoft.NetFramework:** Provee un extenso conjunto de soluciones predefinidas para necesidades generales de la programación de aplicaciones, y administra la ejecución de los programas escritos específicamente con la plataforma.

4.3. Diagrama de despliegue.

Los Diagramas de Despliegue muestran las relaciones físicas de los distintos nodos que componen un sistema y el reparto de los componentes sobre dichos nodos. La vista de despliegue representa la disposición de las instancias de componentes de ejecución en instancias de nodos conectados por enlaces de comunicación. Un nodo es un recurso de ejecución tal como un computador, un dispositivo o memoria. Se encarga de modelar la topología del hardware donde será ejecutado el sistema y los aspectos físicos del mismo (54).

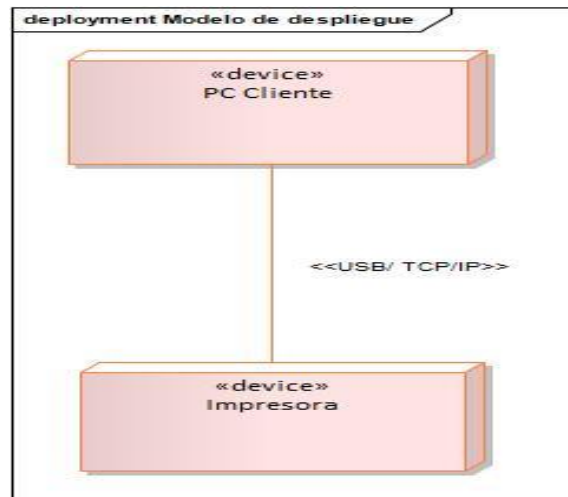


Figura 7. Modelo de Despliegue.

4.4. Nodos del despliegue.

- PC Cliente: Se utiliza para que el cliente pueda acceder a la aplicación que se propone y a los datos con los cuales se trabaja.
- Impresora: es el periférico que el ordenador utiliza para presentar información impresa en papel (55). Se utiliza para imprimir la cantidad de imágenes médicas e informes diagnósticos deseados por el especialista.

4.5. Estándares de codificación.

Los estándares de codificación son modelos de programación a seguir, los cuales describen todos los aspectos de la generación de código que no están enfocadas a la lógica del programa, sino a su estructura y apariencia física para facilitar la lectura, comprensión y mantenimiento del código. Usar técnicas de codificación sólidas y realizar buenas prácticas de programación con vistas a generar un código de alta calidad es de gran importancia para la calidad del software y para obtener un buen rendimiento. Además, si se aplica de forma continua un estándar de codificación bien definido, se utilizan técnicas de programación apropiadas, y, posteriormente, se efectúan revisiones de rutinas, caben muchas posibilidades de que un proyecto informático se convierta en un software fácil de comprender y de mantener. Además permite proveer una guía para el encargado de mantenimiento o actualización del sistema con código legible y bien documentado.

4.5.1 Líneas y espacios en blanco.

Para optimizar la legibilidad del código en ocasiones se utilizan líneas en blanco para separar segmentos de código que pueden corresponder a clases, funciones, declaraciones, implementaciones, comentarios y bloques. A veces se separan con espacios cada operador de su respectivo operando, paréntesis, identificadores, símbolos y algunos lenguajes exigen que se separen las palabras propias del vocabulario de las adyacentes para ser comprendidas por los compiladores. Durante el desarrollo del sistema se colocaron líneas en blanco entre las funciones. Y espacios en blanco después de cada punto y coma, al igual que después de las comas en las funciones y declaraciones.

4.5.2 Llaves.

Para el desarrollo del sistema se definió que las llaves de apertura se colocarán solitarias en la línea siguiente e indentadas al nivel de la línea cabecera del bloque. Las llaves de cierre se colocarán solitarias en la línea que sigue a la última línea dentro del bloque. En el caso de cuerpos de bloque con una sola sentencia se podrá o no usar las llaves a gusto del programador. Este estilo agrega más líneas de código al programa al ubicar las llaves solitarias en una línea, pero a su vez se gana en legibilidad del código.

4.5.3 Comentarios.

Se estableció la utilización de comentarios lineales o de una sola línea cuya sintaxis para el lenguaje C# es `/*`. Se ubicaron encima de la línea a la que se le quiso aplicar el comentario.

4.5.4 Tratamiento de errores.

Uno de los problemas más frecuentes en la ejecución de un sistema surge con las numerosas situaciones de error que puedan existir producto a variables que no pueden ser controladas por el sistema. Frente a situaciones como estas es necesario tomar precauciones para evitar cualquier error emergente que pueda surgir y tratarlo de la forma correcta. Para el manejo de este tipo de situaciones C# ofrece un sistema denominado manejo de excepciones. Los `try` se utilizan para separar el código al que puede afectar una excepción y los bloques `catch` para controlar las excepciones resultantes.

4.5.5 Camel-Case.

CamelCase (56) es la práctica de escribir frases o palabras compuestas eliminando los espacios y poniendo en mayúscula la primera letra de cada palabra. El nombre viene del parecido de estas

mayúsculas, entre las demás letras, con las jorobas de los camellos. Si comienza con mayúscula, se denomina UpperCamelCase y, si no, lowerCamelCase. Para el desarrollo de la aplicación se utilizó:

- UpperCamelCase: Ejemplo: ConfigurationClient, InteropPrinters.

4.6. Conclusiones del capítulo.

A través de la realización de este capítulo se concluye que la confección del modelo de componentes permitió conocer cómo interactúan los paquetes de clases y las diferentes capas de la aplicación dentro del marco de trabajo. La realización del modelo de despliegue brinda una distribución completa del acople de los distintos componentes por lo que está compuesto el sistema. El empleo de los estándares de codificación brindó una mejor organización al trabajo realizado.

CONCLUSIONES

- Después de realizado el estudio sobre los sistemas que realizan la impresión de imágenes médicas en placa y papel se concluye que, los sistemas estudiados no cumplen con los requisitos para ser utilizados.
- Para el desarrollo del componente y para futuras versiones, se generaron las plantillas correspondientes a los flujos de trabajo: “Modelado del Negocio”, “Requerimientos”, “Análisis” y “Diseño” e “Implementación”.
- El componente se implementó de acorde a las pautas de diseño y lo establecido en la especificación de requisitos de software llevada a cabo. Esto garantiza la flexibilidad, robustez y el fácil mantenimiento de la arquitectura basada en el patrón MVVM, lo que permitió que la interfaz de usuario y la lógica del negocio se implementaran independientemente.
- Se desarrolló un componente que posibilita la impresión de las imágenes médicas en papel, facilitando y agilizando de esta manera el trabajo a los especialistas que interactúen con esta aplicación.

RECOMENDACIONES

Con el objetivo de enriquecer la solución propuesta, se proponen las siguientes recomendaciones para el trabajo presentado:

- Realizar pruebas de calidad a la librería con el fin de lograr un mayor perfeccionamiento en las funcionalidades implementadas.
- Integrar el componente propuesto a la Plataforma de Gestión de Imágenes.
- Pilotar el módulo en algún hospital con el objetivo de validarlo.

REFERENCIAS BIBLIOGRÁFICAS

1. **Ortí, Consuelo Belloch.** *LAS TECNOLOGÍAS DE LA INFORMACIÓN Y LAS COMUNICACIONES.* Valencia : s.n.
2. Nanotecnología, Avances científicos y tecnológicos en medicina. *Nanotecnología, Avances científicos y tecnológicos en medicina.* [En línea] [Citado el: 10 de 04 de 2014.] http://www.euroresidentes.com/futuro/avances_cientificos_medicina.htm.
3. Trebol. *Trebol.* [En línea] [Citado el: 10 de 04 de 2014.] <https://sites.google.com/site/treboless/ciencias-y-salud-1/avances-cientificos-y-tecnologicos-en-medicina>.
4. **Sareth, Sarethcyitha.** Slide Share. *Slide Share.* [En línea] 25 de 6 de 2013. [Citado el: 20 de 11 de 2013.] <http://www.slideshare.net/sarethcyitha/importancia-de-la-informatica-en-la-medicina-23448587>.
5. CENETEC-SALUD. *CENETEC-SALUD.* [En línea] 11 de 2009. [Citado el: 4 de 05 de 2014.] http://www.cenetec.salud.gob.mx/descargas/equipo_guias/guias_tec/41gt_PACS.pdf.
6. **Hernández, Juan Carlos Yamacho.** *Componente para la visualización de imágenes médicas.* La Habana : s.n., 2012.
7. Alegs.com.ar. *Diccionario de Informática.* [En línea] [Citado el: 3 de 05 de 2014.] <http://www.alegsa.com.ar/Dic/imprimir.php>.
8. **Cuba, Partido Comunista de.** *Lineamientos de la Política Económica y Social del Partido y la Revolución.* La Habana : s.n., 2011.
9. Proyecto Salón del Hogar. *Proyecto Salón del Hogar.* [En línea] [Citado el: 02 de 05 de 2014.] http://www.proyectosalohogar.com/tecnologia/la_imprensa.htm.
10. Coleccionista de Vitolas de Puros. *Coleccionista de Vitolas de Puros.* [En línea] [Citado el: 02 de 05 de 2014.] <http://www.jaberni-coleccionismo-vitolas.com/1A.4.1-Historia%20de%20la%20Impresion.htm>.
11. Sistema Offset. *Sistema Offset.* [En línea] [Citado el: 02 de 05 de 2014.] <http://offsetgpo2.blogspot.com/2009/02/ventajas-y-desventajas-de-la-impresion.html>.
12. Visual Médica. *Visual Médica.* [En línea] [Citado el: 27 de 11 de 2013.] <http://www.visualmedica.com/visualmedica/index.php/vm-medical-work-station>.

13. Cadime, Cámara de Instituciones de Diagnóstico Médico. *Cadime, Cámara de Instituciones de Diagnóstico Médico*. [En línea] 2013. [Citado el: 30 de 11 de 2013.] <http://www.cadime.com.ar/index.php/es/beneficios-para-prestadores/digitalizacion-del-diagnostico-por-imagen>.
14. Energy Star. *Energy Star*. [En línea] [Citado el: 11 de 04 de 2014.] http://www.energystar.gov/index.cfm?c=home.resources_espanol.
15. TAQ. *TAQ*. [En línea] 2012. [Citado el: 29 de 11 de 2013.] http://www.taq.com.mx/documentos_productos/1348527092.pdf imagenfpo..
16. **Calvo, Carlos**. ABATECH. *ABATECH*. [En línea] [Citado el: 20 de 11 de 2013.] <http://www.abatech.com.ar/medical/descargas/DICOMJetSP-Abatech.pdf>.
17. ABATECH. *ABATECH*. [En línea] [Citado el: 30 de 11 de 2013.] www.abatech.com.ar/medical/impresion_dicom.htm.
18. **D. Ronda, O. Ferrer, N. A. Álvarez**. Habana 2001. *Habana 2001*. [En línea] 23 de Mayo de 2001. [Citado el: 5 de 12 de 2013.] <http://www.memsoctbio.sld.cu/arrepdf/00190.pdf>.
19. **Piqueras, Dr. Joaquim**. Pediatricrad. *Pediatricrad*. [En línea] [Citado el: 25 de 11 de 2013.] <http://www.pediatricrad.info/wo/seram96/dicom.htm>.
20. **Aguilar, Maria de Jesús Cartelle Cruz y Denys Barbaro Vega**. *Sistema para la planificación quirúrgica ortopédica en caderas y rodillas*. Ciudad de la Habana : s.n., 2011.
21. RevistaeSalud.com. *RevistaeSalud.com*. [En línea] mayo de 2011. [Citado el: 22 de 11 de 2013.] <http://www.revistaesalud.com/index.php/revistaesalud/article/view/41/124>.
22. Jatun. *Jatun*. [En línea] [Citado el: 15 de 05 de 2014.] <http://www.jatun.com/web/company/training/javaee5..>
23. Oracle. *Oracle*. [En línea] [Citado el: 15 de 05 de 2014.] <http://www.oracle.com/lad/products/middleware/tools/index.html?ssSourceSiteId=ocomen..>
24. **Network, Microsoft Developer**. Qué es un Patrón de Diseño. [En línea] [Citado el: 15 de mayo de 2014.] <http://msdn.microsoft.com/es-es/library/bb972240.aspx>.

25. Microsoft Developer Network. *MSDN*. [En línea] Microsoft, 2014. [Citado el: 5 de 12 de 2013.] <http://msdn.microsoft.com/es-es/library/zw4w595w%28v=vs.100%29.aspx>.
26. GIZMODO. *GIZMODO*. [En línea] [Citado el: 15 de 05 de 2014.] <http://es.gizmodo.com/que-lenguaje-de-programacion-deberia-aprender-para-emp-1479554075..>
27. **Mendoza, Henry Centro**. SlideShare. *SlideShare*. [En línea] [Citado el: 6 de 12 de 2013.] <http://www.slideshare.net>.
28. **Michael-Pratt**. Pratt. [En línea] 17 de junio de 2013. [Citado el: 23 de mayo de 2014.] <http://www.michael-pratt.com/blog/16/Patrones-de-Diseno-Observador-Observer/>.
29. Microsoft Developer Network. *MSDN*. [En línea] Microsoft. [Citado el: 6 de 12 de 2013.] <http://msdn.microsoft.com/es-es/library/dd460648.aspx..>
30. **González, Enrique**. Aprender a Programar. [En línea] [Citado el: 15 de 05 de 2014.] <http://www.aprenderaprogramar.com..>
31. Scribd. *Scribd*. [En línea] [Citado el: 14 de 06 de 2014.] <http://es.scribd.com/doc/50288837/Caracteristicas-de-PHP>.
32. **Martín, Rafael Leodan Cardero Álvarez y Jesmar Ernesto Fajardo**. *Localización y segmentación de matrículas de vehículos en imágenes digitales*. 2008.
33. **Díaz, Yania Junco López y Ariel Felipe Camacho**. *Desarrollo del módulo de columna para el sistema de Planificación Quirúrgica Ortopédica*. Ciudad de la Habana : s.n., 2012.
34. **Vidal, Rafael Monteagudo Araño y Senia Mercedes González**. *Librería para la gestión de reportes*. Ciudad de la Habana : s.n., 2012.
35. **Software.com.ar**. Software.com.ar. [En línea] [Citado el: 21 de mayo de 2014.] <http://www.software.com.ar/visual-paradigm-para-uml.html>.
36. SparxSystems. *SparxSystems*. [En línea] [Citado el: 17 de 1 de 2014.] <http://www.sparxsystems.com.ar/products/ea.html>.
37. IBM. *IBM*. [En línea] [Citado el: 16 de 1 de 2014.] <http://www-01.ibm.com/software/rational/uml/>.

38. GSI, Grupo de Soluciones Innova. *GSI, Grupo de Soluciones Innova*. [En línea] [Citado el: 16 de 1 de 2014.] <http://www.rational.com.ar/herramientas/rup.html>.
39. **Tigris.org**. Tigris.org. [En línea] [Citado el: 22 de mayo de 2014.] <http://tortoisesvn.tigris.org/>.
40. **GNU, El sistema operativo**. El sistema operativo GNU. [En línea] [Citado el: 22 de mayo de 2014.] <http://www.gnu.org/licenses/licenses.es.html>.
41. **Garzás, Javier**. kybeleconsulting. [En línea] [Citado el: 22 de mayo de 2014.] <http://www.kybeleconsulting.com/articulos/la-certificacion-por-niveles-de-madurez-de-isoiec-15504-spice/>.
42. **Perdigón, Juan Manuel**. *CMMI*.
43. **Pack, Yoandy González Yera y Laura Migdalia Sotolongo**. *Implementación e integración de los módulos en el Planificador Quirúrgico Ortopédico*. La Habana : s.n., 2013.
44. **Martínez, Prof. Ivette Carolina**. LDC, Laboratorio Docente de Computación. *LDC, Laboratorio Docente de Computación*. [En línea] 2008. [Citado el: 20 de 1 de 2014.] http://ldc.usb.ve/~martinez/cursos/ci3715/clase6_AJ2010.pdf.
45. **Laguna, Miguel A**. Departamento de Informática. *Departamento de Informática*. [En línea] 21 de 5 de 2012. [Citado el: 22 de 1 de 2012.] <http://www.infor.uva.es/~mlaguna/is1/apuntes/2-requisitos.pdf>.
46. **Tello, Jesús Cáceres**. UAH, Universidad de Alcalá. *UAH, Universidad de Alcalá*. [En línea] [Citado el: 25 de 1 de 2014.] <http://www2.uah.es/jcaceres/capsulas/DiagramaCasosDeUso.pdf>.
47. **Pacheco, Liliana**. SlideShare. *SlideShare*. [En línea] [Citado el: 18 de 2 de 2014.] <http://www.slideshare.net/lilyPacheco7/arquitectura-de-software-13925226>.
48. Microsoft. *MSDN*. [En línea] [Citado el: 18 de 2 de 2014.] <http://msdn.microsoft.com/es-es/library/windows/apps/jj883732.aspx>.
49. Microsoft. *MSDN*. [En línea] [Citado el: 12 de 2 de 2014.] <http://msdn.microsoft.com/es-es/library/x9afc042.aspx>.
50. **Riesco, Prof. Daniel**. Universidad Nacional de San Luis. *Universidad Nacional de San Luis*. [En línea] [Citado el: 2 de 2 de 2014.] <http://sel.unsl.edu.ar/licenciatura/ingsoft2/UML-DiagramaClaseObjeto.pdf>...
51. **TogetherSoft, Inc**. *Practical UML A Hands-On Introduction for Developers*. 2001.

52. **Fernández, Luis Hernández.** Software Guisho. [En línea] [Citado el: 20 de 05 de 2014.] <http://software.guisho.com/patrones-de-diseno>.
53. Diagrama de Componentes. *Diagrama de Componentes*. [En línea] 18 de 05 de 2009. [Citado el: 10 de 04 de 2014.] http://diagramacomponente.blogspot.com/2009/05/definicion_18.html.
54. **Quisbert, Hugo Michael Marca Huallpara y Susana Nancy Limachi.** *Diagrama de Despliegue*.
55. Conozca su Hardware. *Conozca su Hardware*. [En línea] 2006. [Citado el: 02 de 05 de 2014.] <http://www.conozcasuhardware.com/quees/impres1.htm>.
56. Enciclopedia Universal. *Enciclopedia Universal*. [En línea] [Citado el: 22 de 05 de 2014.] http://enciclopedia_universal.esacademic.com/30677/CamelCase .

BIBLIOGRAFÍA

Ortí, Consuelo Belloch. *LAS TECNOLOGÍAS DE LA INFORMACIÓN Y LAS COMUNICACIONES.* Valencia : s.n.

Nanotecnología, Avances científicos y tecnológicos en medicina. *Nanotecnología, Avances científicos y tecnológicos en medicina.* [En línea] [Citado el: 10 de 04 de 2014.] http://www.euroresidentes.com/futuro/avances_cientificos_medicina.htm.

Trebol. *Trebol.* [En línea] [Citado el: 10 de 04 de 2014.] <https://sites.google.com/site/treboless/ciencias-y-salud-1/avances-cientificos-y-tecnologicos-en-medicina>.

Sareth, Sarethcyitha. Slide Share. *Slide Share.* [En línea] 25 de 6 de 2013. [Citado el: 20 de 11 de 2013.] <http://www.slideshare.net/sarethcyitha/importancia-de-la-informatica-en-la-medicina-23448587>.

CENETEC-SALUD. *CENETEC-SALUD.* [En línea] 11 de 2009. [Citado el: 4 de 05 de 2014.] http://www.cenetec.salud.gob.mx/descargas/equipo_guias/guias_tec/41gt_PACS.pdf.

Hernández, Juan Carlos Yamacho. *Componente para la visualización de imágenes médicas.* La Habana : s.n., 2012.

Alegsa.com.ar. *Diccionario de Informática.* [En línea] [Citado el: 3 de 05 de 2014.] <http://www.alegsa.com.ar/Dic/imprimir.php>.

Cuba, Partido Comunista de. *Lineamientos de la Política Económica y Social del Partido y la Revolución.* La Habana : s.n., 2011.

Proyecto Salón del Hogar. *Proyecto Salón del Hogar.* [En línea] [Citado el: 02 de 05 de 2014.] http://www.proyectosalohogar.com/tecnologia/la_imprensa.htm.

Coleccionista de Vitolas de Puros. *Coleccionista de Vitolas de Puros.* [En línea] [Citado el: 02 de 05 de 2014.] <http://www.jaberni-coleccionismo-vitolas.com/1A.4.1-Historia%20de%20la%20Impresion.htm>.

Sistema Offset. *Sistema Offset.* [En línea] [Citado el: 02 de 05 de 2014.] <http://offsetgpo2.blogspot.com/2009/02/ventajas-y-desventajas-de-la-impresion.html>.

Visual Médica. *Visual Médica.* [En línea] [Citado el: 27 de 11 de 2013.] <http://www.visualmedica.com/visualmedica/index.php/vm-medical-work-station>.

Cadime, Cámara de Instituciones de Diagnóstico Médico. *Cadime, Cámara de Instituciones de Diagnóstico Médico*. [En línea] 2013. [Citado el: 30 de 11 de 2013.] <http://www.cadime.com.ar/index.php/es/beneficios-para-prestadores/digitalizacion-del-diagnostico-por-imagen>.

Energy Star. *Energy Star*. [En línea] [Citado el: 11 de 04 de 2014.] http://www.energystar.gov/index.cfm?c=home.resources_espanol.

TAQ. *TAQ*. [En línea] 2012. [Citado el: 29 de 11 de 2013.] http://www.taq.com.mx/documentos_productos/1348527092.pdf imagenfpo..

Calvo, Carlos. ABATECH. *ABATECH*. [En línea] [Citado el: 20 de 11 de 2013.] <http://www.abatech.com.ar/medical/descargas/DICOMJetSP-Abatech.pdf>.

ABATECH. *ABATECH*. [En línea] [Citado el: 30 de 11 de 2013.] www.abatech.com.ar/medical/impresion_dicom.htm.

D. Ronda, O. Ferrer, N. A. Álvarez. Habana 2001. *Habana 2001*. [En línea] 23 de Mayo de 2001. [Citado el: 5 de 12 de 2013.] <http://www.memso.bio.sld.cu/arrepdf/00190.pdf>.

Piqueras, Dr. Joaquim. *Pediatricrad*. [En línea] [Citado el: 25 de 11 de 2013.] <http://www.pediatricrad.info/wo/seram96/dicom.htm>.

Aguilar, Maria de Jesús Cartelle Cruz y Denys Barbaro Vega. *Sistema para la planificación quirúrgica ortopédica en caderas y rodillas*. Ciudad de la Habana : s.n., 2011.

RevistaeSalud.com. *RevistaeSalud.com*. [En línea] mayo de 2011. [Citado el: 22 de 11 de 2013.] <http://www.revistaesalud.com/index.php/revistaesalud/article/view/41/124>.

Jatun. *Jatun*. [En línea] [Citado el: 15 de 05 de 2014.] <http://www.jatun.com/web/company/training/javaaee5..>

Oracle. *Oracle*. [En línea] [Citado el: 15 de 05 de 2014.] <http://www.oracle.com/lad/products/middleware/tools/index.html?ssSourceSiteId=ocomen..>

Network, Microsoft Developer. Qué es un Patrón de Diseño. [En línea] [Citado el: 15 de mayo de 2014.] <http://msdn.microsoft.com/es-es/library/bb972240.aspx>.

Microsoft Developer Network. *MSDN*. [En línea] Microsoft, 2014. [Citado el: 5 de 12 de 2013.] <http://msdn.microsoft.com/es-es/library/zw4w595w%28v=vs.100%29.aspx>.

GIZMODO. *GIZMODO*. [En línea] [Citado el: 15 de 05 de 2014.] <http://es.gizmodo.com/que-lenguaje-de-programacion-deberia-aprender-para-emp-1479554075..>

Mendoza, Henry Centro. SlideShare. *SlideShare*. [En línea] [Citado el: 6 de 12 de 2013.] <http://www.slideshare.net>.

Michael-Pratt. Pratt. [En línea] 17 de junio de 2013. [Citado el: 23 de mayo de 2014.] <http://www.michael-pratt.com/blog/16/Patrones-de-Diseno-Observador-Observer/>.

Microsoft Developer Network. *MSDN*. [En línea] Microsoft. [Citado el: 6 de 12 de 2013.] <http://msdn.microsoft.com/es-es/library/dd460648.aspx..>

González, Enrique. Aprender a Programar. [En línea] [Citado el: 15 de 05 de 2014.] <http://www.aprenderaprogramar.com..>

Scribd. *Scribd*. [En línea] [Citado el: 14 de 06 de 2014.] <http://es.scribd.com/doc/50288837/Caracteristicas-de-PHP>.

Martín, Rafael Leodan Cardero Álvarez y Jesmar Ernesto Fajardo. *Localización y segmentación de matrículas de vehículos en imágenes digitales*. 2008.

Díaz, Yania Junco López y Ariel Felipe Camacho. *Desarrollo del módulo de columna para el sistema de Planificación Quirúrgica Ortopédica*. Ciudad de la Habana : s.n., 2012.

Vidal, Rafael Monteagudo Araño y Senia Mercedes González. *Librería para la gestión de reportes*. Ciudad de la Habana : s.n., 2012.

Software.com.ar. Software.com.ar. [En línea] [Citado el: 21 de mayo de 2014.] <http://www.software.com.ar/visual-paradigm-para-uml.html>.

SparxSystems. *SparxSystems*. [En línea] [Citado el: 17 de 1 de 2014.] <http://www.sparxsystems.com.ar/products/ea.html>.

IBM. *IBM*. [En línea] [Citado el: 16 de 1 de 2014.] <http://www-01.ibm.com/software/rational/uml/>.

GSI, Grupo de Soluciones Innova. *GSI, Grupo de Soluciones Innova*. [En línea] [Citado el: 16 de 1 de 2014.] <http://www.rational.com.ar/herramientas/rup.html..>

Tigris.org. Tigris.org. [En línea] [Citado el: 22 de mayo de 2014.] <http://tortoisesvn.tigris.org/>.

- GNU, El sistema operativo.** El sistema operativo GNU. [En línea] [Citado el: 22 de mayo de 2014.] <http://www.gnu.org/licenses/licenses.es.html>.
- Garzás, Javier.** kybeleconsulting. [En línea] [Citado el: 22 de mayo de 2014.] <http://www.kybeleconsulting.com/articulos/la-certificacion-por-niveles-de-madurez-de-isoiec-15504-spice/>.
- Perdigón, Juan Manuel.** *CMMI*.
- Pack, Yoandy González Yera y Laura Migdalia Sotolongo.** *Implementación e integración de los módulos en el Planificador Quirúrgico Ortopédico*. La Habana : s.n., 2013.
- Martínez, Prof. Ivette Carolina.** LDC, Laboratorio Docente de Computación. *LDC, Laboratorio Docente de Computación*. [En línea] 2008. [Citado el: 20 de 1 de 2014.] http://ldc.usb.ve/~martinez/cursos/ci3715/clase6_AJ2010.pdf.
- Laguna, Miguel A.** Departamento de Informática. *Departamento de Informática*. [En línea] 21 de 5 de 2012. [Citado el: 22 de 1 de 2012.] <http://www.infor.uva.es/~mlaguna/is1/apuntes/2-requisitos.pdf> .
- Tello, Jesús Cáceres.** UAH, Universidad de Alcalá. *UAH, Universidad de Alcalá*. [En línea] [Citado el: 25 de 1 de 2014.] <http://www2.uah.es/jcaceres/capsulas/DiagramaCasosDeUso.pdf> .
- Pacheco, Liliana.** SlideShare. *SlideShare*. [En línea] [Citado el: 18 de 2 de 2014.] <http://www.slideshare.net/lilyPacheco7/arquitectura-de-software-13925226>.
- Microsoft. *MSDN*. [En línea] [Citado el: 18 de 2 de 2014.] <http://msdn.microsoft.com/es-es/library/windows/apps/jj883732.aspx>.
- Microsoft. *MSDN*. [En línea] [Citado el: 12 de 2 de 2014.] <http://msdn.microsoft.com/es-es/library/x9afc042.aspx>.
- Riesco, Prof. Daniel.** Universidad Nacional de San Luis. *Universidad Nacional de San Luis*. [En línea] [Citado el: 2 de 2 de 2014.] <http://sel.unsl.edu.ar/licenciatura/ingsoft2/UML-DiagramaClaseObjeto.pdf...>
- TogetherSoft, Inc.** *Practical UML A Hands-On Introduction for Developers*. 2001.
- Fernández, Luis Hernández.** Software Guisho. [En línea] [Citado el: 20 de 05 de 2014.] <http://software.guisho.com/patrones-de-diseno>.

Diagrama de Componentes. *Diagrama de Componentes*. [En línea] 18 de 05 de 2009. [Citado el: 10 de 04 de 2014.] http://diagramacomponente.blogspot.com/2009/05/definicion_18.html.

Quisbert, Hugo Michael Marca Huallpara y Susana Nancy Limachi. *Diagrama de Despliegue*.

Conozca su Hardware. *Conozca su Hardware*. [En línea] 2006. [Citado el: 02 de 05 de 2014.] <http://www.conozcasuhardware.com/quees/impres1.htm>.

Enciclopedia Universal. *Enciclopedia Universal*. [En línea] [Citado el: 22 de 05 de 2014.] http://enciclopedia_universal.esacademic.com/30677/CamelCase .

ANEXOS

Anexo 1. Entrevista realizada al Licenciado en Imagenología y Especialista en los TAC, Roberto Miranda Barrabía.

1- ¿Cuántas placas se hacen en el Hospital diariamente?

Actualmente en el Hospital Frank País son muchos los pacientes que vienen a realizarse placas radiográficas, por lo que se realizan aproximadamente alrededor de 200 placas por día.

2- ¿Resulta costosa la impresión de estas placas con respecto al papel?

La impresión de estas placas resultan 10 o 20 veces más costosas que las impresiones de estas imágenes en papel, debido a que los reactivos con los cuales se realizan las placas (acetato) deben ser cambiados cada 15 días. Además los equipos donde se realizan las mismas, son equipos muy caros.

3- ¿Qué tiempo demora un especialista en realizar las impresiones en placa?

Para realizar estas impresiones un especialista demora como máximo 15 minutos aproximadamente.

4- ¿Qué tipo de placa se vería mejor en papel?

Algunas placas por sus características no deben ser impresas en papel, pero en el caso de la tomografía y resonancia se podrían hacer en papel y se ahorrarían los productos que pudieran ser utilizados luego en otras impresiones que si se necesite ver en una placa.

5- ¿Qué ventajas y desventajas traería consigo la impresión en papel?

La impresión en papel traería como ventaja la rapidez en que se realizaría este proceso, además del ahorro de dinero que proporcionará al país. Además si un paciente pierde la placa que se llevó a su hogar sería más fácil y rápido hacer la impresión de la misma en papel.

Como desventaja se puede decir que no todas las imágenes médicas pueden ser impresas en papel, además el papel puede romperse con facilidad e incluso puede hasta con el tiempo borrarse la imagen.

Anexo 2. Prefijos de los requerimientos no funcionales.

Categoría	Prefijo	Ejemplo
-----------	---------	---------

Componente para la impresión convencional de imágenes médicas. Anexos

Usabilidad	RNU	RNU 1. Requisito de usabilidad
Diseño e Implementación	RNDI	RNDI 1. Requisito de diseño e implementación
Legal	RNL	RNL 1. Requisito legal
Fiabilidad	RNF	RNF 1. Requisito de fiabilidad
Seguridad	RNS	RNS 1. Requisito de seguridad
Eficiencia	RNE	RNE 1. Requisito de eficiencia
Funcionamiento	RNFO	RNFO 1. Requisito de funcionamiento
Soporte	RNSO	RNSO 1. Requisito de soporte
Interfaz de usuario	RNIU	RNIU 1. Requisito de interfaz de usuario
Interconexión	RNI	RNI 1. Requisito de interconexión
Componentes Comprados	RNCC	RNCC 1. Requisito de componentes comprados
Licencia	RNLI	RNLI 1. Requisito de licencia

Anexo 3. Descripción detallada de los casos de uso del sistema.

CU # 1. Gestionar impresora	
Objetivo	Gestionar las impresoras a las que podrá conectarse el sistema. De esta forma se podrá adicionar, editar y eliminar impresoras.
Actores	Especialista (Inicia)
Resumen	El caso de uso inicia cuando el Especialista selecciona la opción Configuración / Impresoras NO DICOM; el sistema muestra las impresoras registradas y las opciones para adicionar, modificar o eliminar alguna impresoras.
Complejidad	Media
Prioridad	Alta
Referencias	RF 1.1, 1.2, 1.3.

Componente para la impresión convencional de imágenes médicas.

Anexos

Precondiciones	No aplica
Postcondiciones	Se agregaron los nuevos datos de la impresora. Se despliega un menú con las impresoras registradas. Se modificaron los valores de las impresoras. Se eliminó la impresora seleccionada.
Flujo de eventos	
Flujo básico <Adicionar impresora >	
<ol style="list-style-type: none"> 1. El usuario abre el módulo de configuración. 2. El usuario selecciona la pestaña Impresoras NO DICOM. 3. El usuario selecciona la opción Adicionar. 4. El sistema muestra una interfaz para seleccionar el método de instalación. 5. El usuario selecciona el método que utilizará para instalar la impresora y presiona el botón siguiente. 6. El sistema muestra una interfaz para escribir la dirección de la impresora. 7. El usuario llena los siguientes campos: <ul style="list-style-type: none"> ➤ Tipo de dispositivo. ➤ Dirección ip. ➤ Nombre del puerto. 8. El usuario selecciona la opción Adicionar. 9. El sistema valida los datos y luego muestra un menú con las impresoras registradas. 10. Termina el caso de uso. 	
Flujos alternos	
9a < Existen campos en blanco >	
<ol style="list-style-type: none"> 1. El sistema muestra un mensaje de error si existe algún campo en blanco. (“Existen campos vacíos. Llene los campos e intente de nuevo.”) 2. Regresar al paso 6 del flujo básico. 	
9a < Existen datos incorrectos >	
<ol style="list-style-type: none"> 3. El sistema muestra un indicador sobre los campos incorrectos. 4. Regresar al paso 6 del flujo básico. 	
9a <Existen datos incompletos>	
<ol style="list-style-type: none"> 1. El sistema muestra un indicador sobre los campos incompletos. 2. Regresar al paso 6 del flujo básico. 	

<p>Flujo básico <Modificar impresora></p> <ol style="list-style-type: none"> 1. El usuario abre el módulo de Configuración. 2. El usuario selecciona la pestaña Impresoras NO DICOM. 3. El sistema muestra un listado de todas las impresoras que se encuentran registradas. 4. El usuario selecciona la impresora a la cual va a modificarle sus valores. 5. El sistema brinda la posibilidad de cambiar los valores de los atributos de la impresora. 6. El usuario introduce los nuevos valores y selecciona la opción Modificar. 7. El sistema valida los datos. 8. Termina el caso de uso.
<p>Flujos alternos</p>
<p>7a <Existen datos incompletos></p> <ol style="list-style-type: none"> 3. El sistema muestra un indicador sobre los campos incompletos. 4. Regresar al paso 5 del flujo básico.
<p>7b <Existen datos incorrectos></p> <ol style="list-style-type: none"> 1. El sistema muestra un indicador sobre los campos incorrectos. 2. Regresar al paso 5 del flujo básico.
<p>7b <Existen campos en blanco></p> <ol style="list-style-type: none"> 3. El sistema muestra un mensaje de error si existe algún campo en blanco. (“Existen campos vacíos. Llene los campos e intente de nuevo.”) 4. Regresar al paso 5 del flujo básico.
<p>Flujo básico <Eliminar impresora></p> <ol style="list-style-type: none"> 1. El usuario abre el módulo de configuración. 2. El usuario selecciona la pestaña Impresoras NO DICOM. 3. El sistema muestra un listado de todas las impresoras que se encuentran registradas. 4. El usuario selecciona la impresora que desea eliminar. 5. El usuario selecciona la opción Eliminar. 6. El sistema muestra el mensaje “¿Desea eliminar la impresora seleccionada?”. 7. El usuario selecciona la opción de aceptar. 8. El sistema elimina la impresora, quitándola del listado de impresoras registradas. 9. Termina el caso de uso.

Flujos alternos		
7b< Eliminar impresora en uso>		
<ol style="list-style-type: none"> 1. El sistema muestra un mensaje de error. (“La impresora que desea eliminar está siendo usada en estos momentos.”) 2. Regresar al paso 3 del flujo básico. 		
No aplica		
Relaciones	CU Incluidos	No aplica
	CU Extendidos	No aplica

CU # 2. Imprimir imágenes.	
Objetivo	Seleccionar el tipo de impresora a utilizar, configurar impresoras y acceder a la lista de los estudios pendientes de impresión. Seleccionar las diferentes opciones de papel, la cantidad de copias a realizar, las imágenes que se desean imprimir en el papel y de permitir borrar imágenes de la selección a imprimir.
Actores	Especialista (Inicia)
Resumen	El caso de uso inicia cuando el actor selecciona la opción Impresión en papel ubicada en la pestaña Archivo, el sistema muestra los atributos que se deben seleccionar para imprimir las imágenes.
Complejidad	Baja
Prioridad	Alta
Referencias	RF 2.1, 2.2, 2.3, 2.4, 2.5, 2.6, 2.7, 2.8.
Precondiciones	Deben existir impresoras registradas en la aplicación.
Postcondiciones	Se seleccionó la impresora deseada. Se verificó la configuración de la impresora. Se mostró la cola de impresión. Se seleccionaron las opciones de papel. Se especificó el número de copias a imprimir. Se seleccionaron las

	imágenes a imprimir. Se borraron las imágenes deseadas.
Flujo de eventos	
Flujo básico < Seleccionar impresora >	
<ol style="list-style-type: none"> 1. El especialista selecciona la pestaña Archivo en el visor. 2. El especialista selecciona la opción Impresión en papel. 3. El sistema muestra las secciones: <ol style="list-style-type: none"> a. Impresora: muestra las siguientes opciones: <ol style="list-style-type: none"> i. Impresoras ii. Cola de impresión b. Opciones de papel c. Imprimir d. Imágenes para imprimir: contiene las opciones Seleccionarlas todas/Deseleccionarlas todas y Borrar imágenes. 4. El especialista selecciona la impresora deseada, que puede ser una impresora físicas (convencionales) o impresoras virtuales (OneNote, PDF, XPS y Fax). 5. El sistema busca la impresora deseada en su lista de impresoras instaladas y la muestra al tope de la lista. 6. Termina el caso de uso. 	
Flujo básico <Configurar Impresoras>	
<ol style="list-style-type: none"> 1. El especialista selecciona la opción Impresoras. 2. El sistema muestra una interfaz para seleccionar el método de instalación. 3. El usuario selecciona el método que utilizará para instalar la impresora y presiona el botón siguiente. 4. El sistema muestra una interfaz para escribir la dirección de la impresora. 5. El usuario llena los campos siguientes: <ul style="list-style-type: none"> ➤ Tipo de dispositivo. ➤ Dirección ip. ➤ Nombre del puerto. 6. El usuario selecciona la opción Finalizar. 7. Termina el caso de uso. 	
Flujos alternos	
5ª < Existen campos en blanco >	

<ol style="list-style-type: none"> 1. El sistema muestra un mensaje de error si existe algún campo en blanco. (“Dirección no válida. Escriba una dirección válida e intente de nuevo.”) 2. Regresar al paso 4 del flujo básico.
<p>Flujo básico <Mostrar cola de impresión></p>
<ol style="list-style-type: none"> 1. El especialista selecciona la opción Cola de impresión. 2. El sistema muestra una tabla listando los estudios pendientes de impresión organizada con los siguientes atributos: <ul style="list-style-type: none"> ➤ Paciente ➤ Estado ➤ Hora de creado ➤ Formato ➤ Error 3. El especialista visualiza la lista de estudios pendientes de impresión. 4. Termina el caso de uso.
<p>Flujos alternos <No aplica>.</p>
<p>Flujo básico <Seleccionar opciones de papel></p>
<ol style="list-style-type: none"> 1. El especialista selecciona la sección “Opciones de papel” 2. El sistema muestra las siguientes opciones: <ul style="list-style-type: none"> ➤ Tamaño ➤ Orientación ➤ Mosaico ➤ Otras opciones de impresión: permite elegir el papel con anotaciones. 3. El especialista selecciona los parámetros de presentación deseados. 4. El sistema configura el papel para la impresión según las opciones seleccionadas. 5. Termina el caso de uso.
<p>Flujo básico <Especificar número de copias></p>
<ol style="list-style-type: none"> 1. El especialista elije o escribe en la opción Copias la cantidad de copias que quiere imprimir. 2. El sistema salva esa información para ser enviada a la impresora. 3. Termina el caso de uso.
<p>Flujo básico <Seleccionar imágenes a imprimir></p>
<ol style="list-style-type: none"> 1. El especialista selecciona las imágenes deseadas marcándolas una a una o si desea

<p>marcarlas todas accede al botón Seleccionarlas todas.</p> <ol style="list-style-type: none"> El sistema selecciona las imágenes deseadas. Termina el caso de uso. 		
Flujos alternos		
1a Desechar selección		
<ol style="list-style-type: none"> El especialista desmarca las imágenes que no desee imprimir o si quiere quitar toda la selección accede a la opción Deseleccionarlas todas. El sistema deselecciona las imágenes deseadas. Termina el caso de uso. 		
Flujo básico <Borrar imágenes seleccionadas>		
<ol style="list-style-type: none"> El especialista luego de haber seleccionado las imágenes a borrar selecciona la opción Borrar imágenes. Termina el caso de uso. 		
Relaciones	CU Incluidos	No aplica
	CU Extendidos	No aplica

CU # 3. Adicionar imágenes a imprimir.	
Objetivo	Adicionar una imagen o una serie de imágenes que serán enviadas a la vista de la impresión.
Actores	Especialista (Inicia)
Resumen	El caso de uso inicia cuando el especialista selecciona la pestaña Principal y la opción Imprimir en papel, el sistema muestra las opciones que permiten adicionar la serie o una imagen.
Complejidad	Baja
Prioridad	Alta
Referencias	RF 3.1, 3.2.

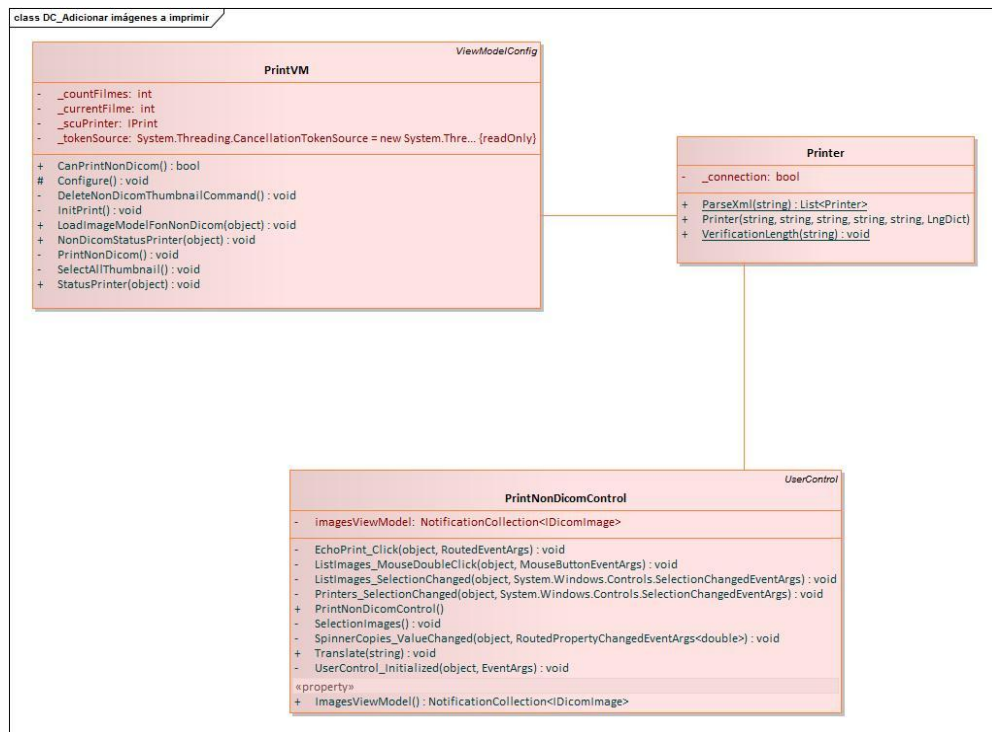
Precondiciones	Para adicionar imágenes o series a imprimir deben encontrarse las mismas cargadas en el visor.
Postcondiciones	Se adicionó una serie. Se adicionó una imagen.
Flujo de eventos	
Flujo básico <Adicionar imagen>	
<ol style="list-style-type: none"> 1. El especialista selecciona la pestaña Principal en el visor. 2. El especialista selecciona la imagen deseada. 3. El especialista selecciona la opción Imprimir en papel. 4. El sistema muestra las opciones: <ul style="list-style-type: none"> • Adicionar imagen. • Adicionar serie. 5. El especialista selecciona la opción Adicionar imagen. 6. El sistema envía la imagen deseada a la lista de imágenes a imprimir. 7. Termina el caso de uso. 	
Flujos alternos	
2ª. No existen imágenes seleccionadas	
<ol style="list-style-type: none"> 1. El sistema inhabilita el botón que permite Imprimir. 2. Regresar al paso 2 del flujo básico. 	
Flujo básico <Adicionar serie>	
<ol style="list-style-type: none"> 1. El especialista selecciona la serie que desea imprimir. 2. El especialista selecciona la opción Adicionar serie. 3. El sistema envía la serie deseada a la lista de imágenes a imprimir. 4. Termina el caso de uso. 	
Flujos alternos	
1ª. No existen series seleccionadas	
<ol style="list-style-type: none"> 1. El sistema inhabilita el botón que permite Imprimir. 2. Regresar al paso 2 del flujo básico. 	

Componente para la impresión convencional de imágenes médicas. **Anexos**

Relaciones	CU Incluidos	No aplica
	CU Extendidos	No aplica

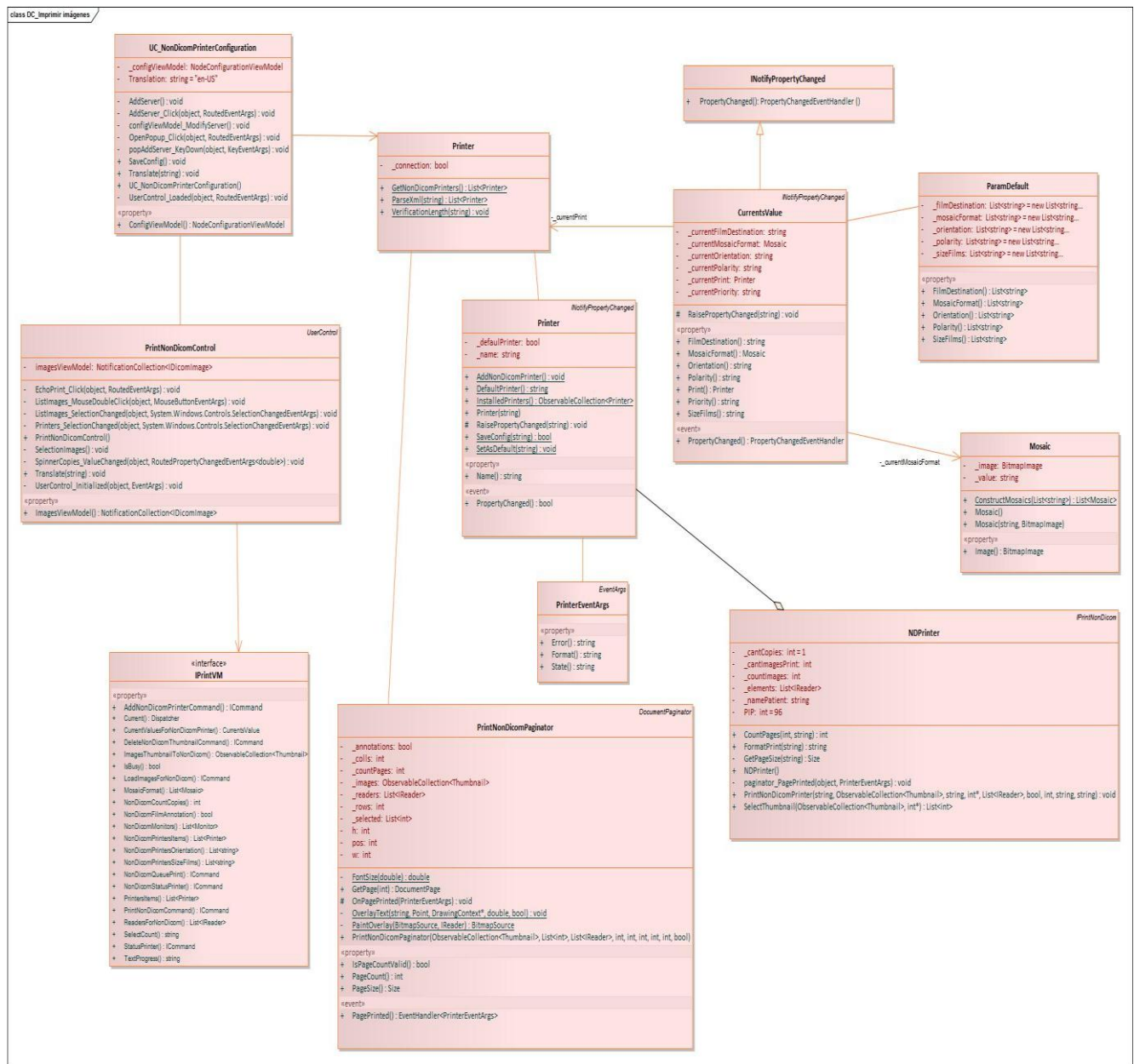
Anexo 4. Diagramas de las clases del sistema.

- DC_Adicionar imágenes a imprimir.



Componente para la impresión convencional de imágenes médicas. Anexos

➤ DC_Imprimir imágenes.



Clase NDPrinter.

Propósito: Es la clase principal del sistema, posee los métodos que inicializan la impresión de las imágenes. Enlaza el código del sistema con la interfaz visual.

Descripción: Permite realizar la configuración de las impresiones en el papel para luego inicializar la impresión de las imágenes.

```
NDPrinter IPrintNonDicom
- _cantCopies: int = 1
- _cantImagesPrint: int
- _countImages: int
- _elements: long
- _namePatient: string
- PIP: int = 96
+ CountPages(int, string) : int
+ FormatPrint(string) : string
- GetPageSize(string) : Size
+ NDPrinter()
- paginator_PagePrinted(object, PrinterEventArgs) : void
+ PrintNonDicomPrinter(string, ObservableCollection<Thumbnail>, string, int*, List<IReader>, bool, int, string, string) : void
+ SelectThumbnail(ObservableCollection<Thumbnail>, int*) : List<int>
```

Clase PrintNonDicomPaginator

Propósito: Permite manejar la cantidad de páginas que serán impresas.

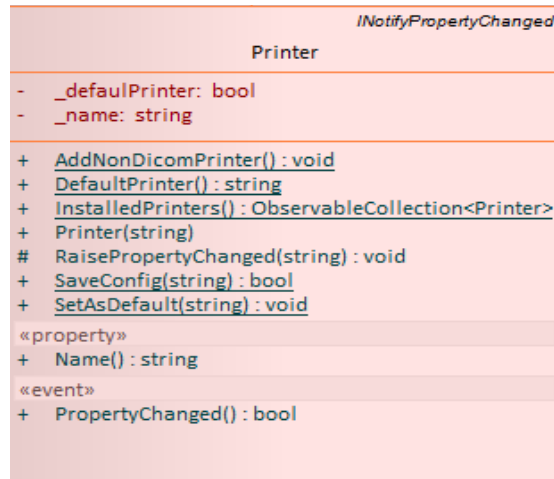
Descripción: Esta clase contiene las variables y los métodos que permiten realizar la impresión de varias hojas en caso de que se impriman muchas imágenes.

```
PrintNonDicomPaginator DocumentPaginator
- _annotations: bool
- _colls: int
- _countPages: int
- _images: ObservableCollection<Thumbnail>
- _readers: List<IReader>
- _rows: int
- _selected: List<int>
- h: int
- pos: int
- w: int
- FontSize(double) : double
+ GetPage(int) : DocumentPage
# OnPagePrinted(PrinterEventArgs) : void
- OverlayText(string, Point, DrawingContext*, double, bool) : void
- PaintOverlay(BitmapSource, IReader) : BitmapSource
+ PrintNonDicomPaginator(ObservableCollection<Thumbnail>, List<int>, List<IReader>, int, int, int, int, int, bool)
«property»
+ IsPageCountValid() : bool
+ PageCount() : int
+ PageSize() : Size
«event»
+ PagePrinted() : EventHandler<PrinterEventArgs>
```

Componente para la impresión convencional de imágenes médicas. **Anexos**

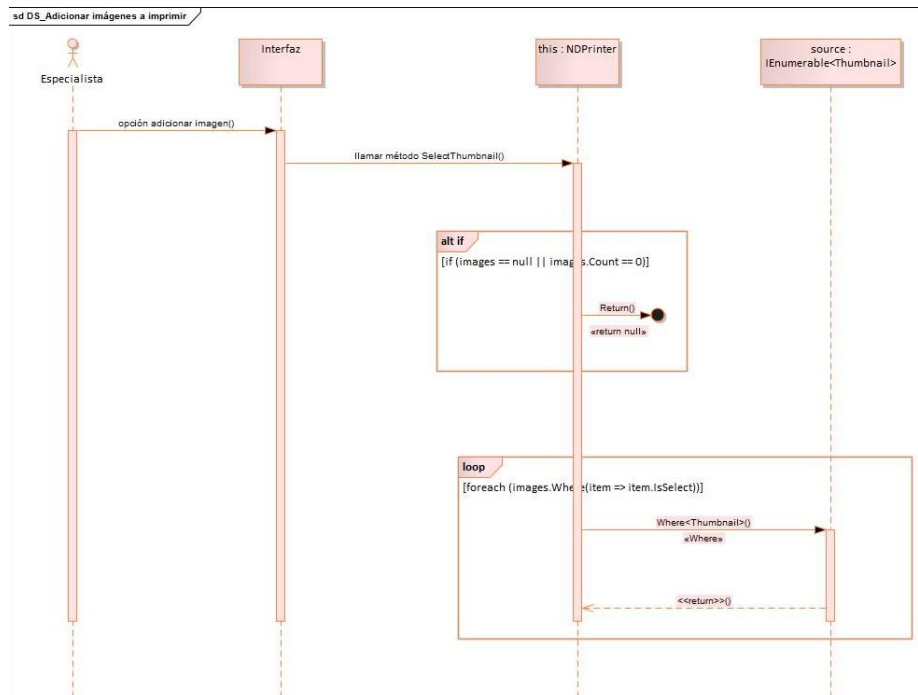
Propósito: Permite configurar las impresoras nuevas.

Descripción: Esta clase contiene los métodos y propiedades que permiten configurar una nueva impresora.



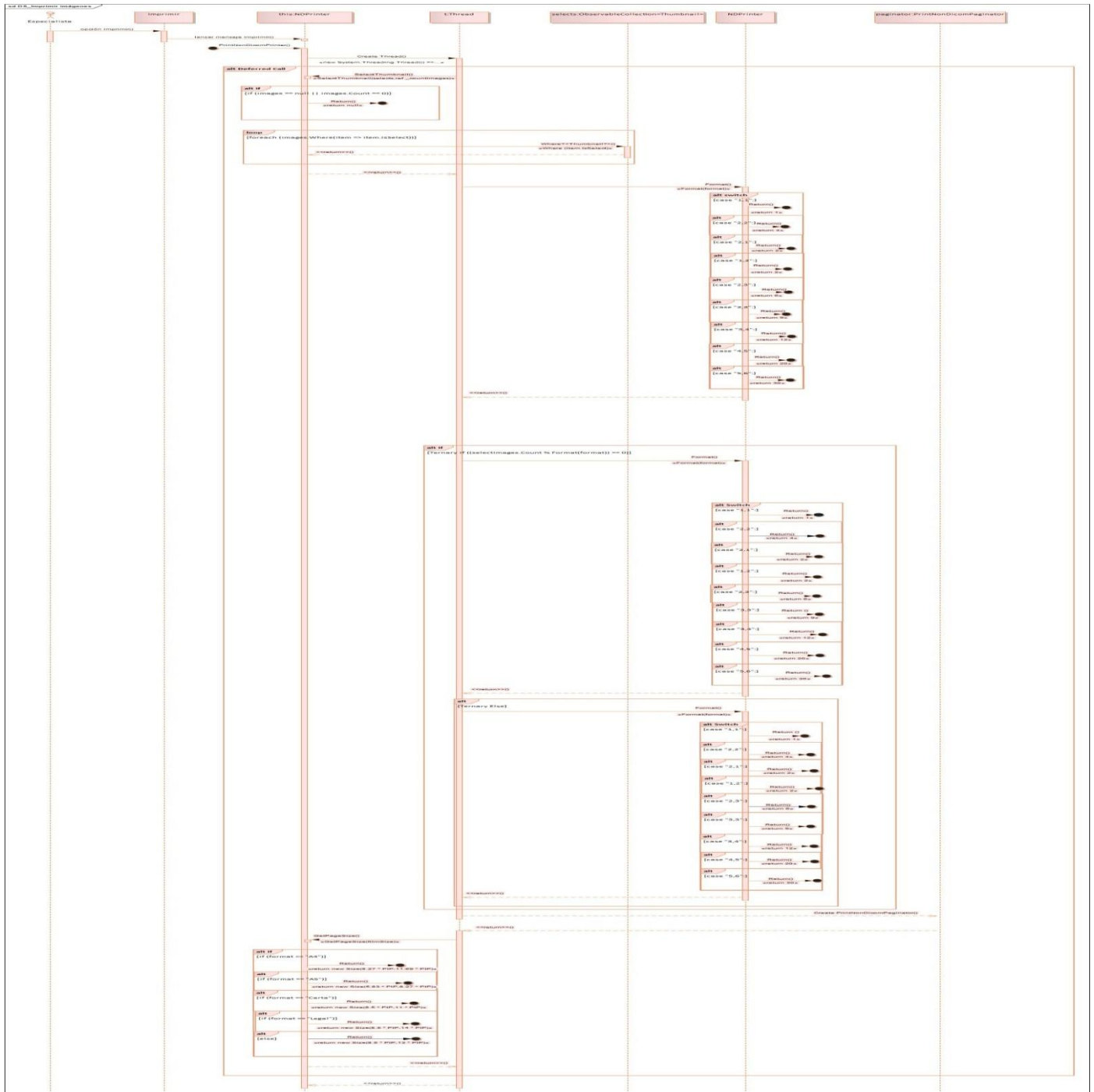
Anexo 5. Diagramas de secuencia de los casos de uso.

- DS_Adicionar imágenes a imprimir.



Componente para la impresión convencional de imágenes médicas. **Anexos**

➤ DS_Imprimir imágenes.



GLOSARIO DE TÉRMINOS

TIC: Las tecnologías de la información y la comunicación (TIC) agrupan los elementos y las técnicas usadas en el tratamiento y la transmisión de las informaciones, principalmente de informática, internet y telecomunicaciones.

DICOM: (Digital Imaging and Communication in Medicine): Estándar para el tratamiento de imágenes digitales y comunicaciones para el campo de la medicina, que facilita el manejo de la información médica entre hospitales y centros de investigación.

Estándar: Sirve como tipo, modelo, norma, patrón o referencia.

Librerías: En Inglés library, cuando se habla de ordenadores, refiere al conjunto de rutinas que realizan las operaciones usualmente requeridas por los programas. Las librerías pueden ser compartidas, lo que quiere decir que las rutinas de la librería residen en un fichero distinto de los programas que las utilizan. Los programas enlazados con bibliotecas compartidas no funcionarán a menos que se instalen las bibliotecas o librerías necesarias.

PACS: Picture Archiving and Communication System (PACS). Sistema para el almacenamiento y comunicación de imágenes médicas. **Patrón arquitectónico:** Un patrón arquitectónico expresa un esquema estructural fundamental de la organización para un sistema de software, que consiste en subsistemas, sus responsabilidades e interrelaciones.

Framework: Es una estructura conceptual y tecnológica de soporte definido, normalmente con artefactos o módulos de software concretos, con base a la cual otro proyecto de software puede ser más fácilmente organizado y desarrollado.

NEMA: (National Electrical Manufacturers Association): Asociación comercial líder en los Estados Unidos en representación de los fabricantes de productos del electro-industria.

Plataforma: Determinado software y/o hardware con el cual una aplicación puede ejecutarse o desarrollarse.

Requisitos: Capacidades, condiciones o cualidades que el sistema debe cumplir y tener.

GPL: (General Public License): El uso de la GPL ordinaria para una biblioteca la hace disponible únicamente para programas libres.

RIS: Sistema de Información Radiológica. Es un sistema que organiza y almacena la información generada en un servicio de imagenología.

ACR: American College Radiology. Los códigos ACR son un sistema de codificación decimal, en árbol jerárquico, que permite codificar buena parte de la anatomía y la patología de una forma consistente y fácil.

Patrón arquitectónico: Un patrón arquitectónico expresa un esquema estructural fundamental de la organización para un sistema de software, que consiste en subsistemas, sus responsabilidades e interrelaciones.

Log: es un registro oficial de eventos durante un rango de tiempo en particular.

Servidor: Computadora que forma parte de una red, brindando servicios a otras computadoras que reciben el nombre de clientes.