



Universidad de las Ciencias Informáticas

Facultad 3

Facultad 2

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

**Herramienta para el proceso de grabado y anonimización de imágenes
DICOM para el sistema XAVIA PACS 3.0**

Autores:

Sindia Trujillo Rodríguez

Cristhian B. Seguí González

Tutores:

Ing. Jessie Romero Pérez

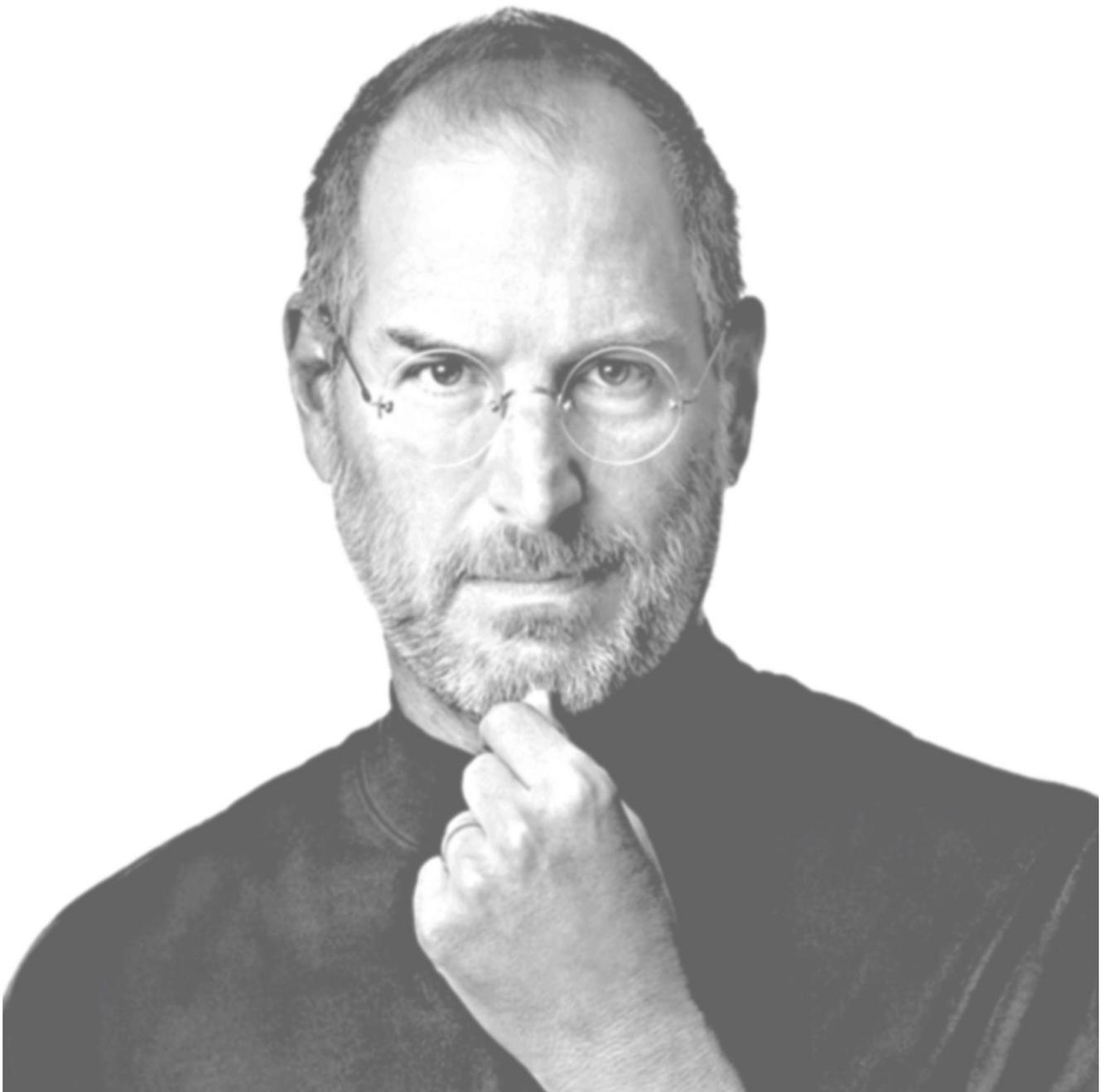
Dr.C.Arturo Orellana García

La Habana, 2020

“Año 62 de la Revolución”

“Si hoy fuese el último día de tu vida ¿querrías hacer lo que vas a hacer hoy? Si la respuesta es no durante demasiados días seguidos, debes cambiar algo.”

Steve Jobs



DECLARACIÓN DE AUTORÍA

Declaramos ser autores del presente trabajo de diploma y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmamos la presente a los ____ días del mes de _____ del año _____.

Firma del Autor:
Sindia Trujillo Rodríguez

Firma del Autor:
Cristhian B. Seguín González

Firma del Tutor :
Jessie Romero Pérez

Firma del Tutor :
Arturo Orellana García

Dedicatoria

Sindia

Este gran logro en mi vida se lo dedico a mis padres, por creer en mí, apoyarme siempre y educarme como lo han hecho.

Cristhian

A mi mamá por ser ejemplo de sacrificio y apoyarme a lo largo de toda mi vida en el cumplimiento de mis aspiraciones y a mi novia por siempre estar y brindarme tanto apoyo.

Agradecimientos

Sindia

Este tiempo de esfuerzo y sacrificio no hubiera sido posible sin el apoyo de aquellas personas que de una forma u otra me han brindado su ayuda para que consiguiera mis objetivos, por eso le quiero agradecer a:

Mis padres, por ser la columna vertebral en mi desarrollo en la vida, por apoyarme siempre en todas mis decisiones, y darme los mejores consejos para que me superara y me hiciera una profesional, por darme siempre todo lo que han podido o incluso lo que no han podido, por ayudarme a ser la mujer que soy hoy, por amarme y ser los mejores padres del mundo, gracias, los amo.

A mis hermanos Dany y Liría ("Sistema"), por confiar siempre en mí, por hacerme reír con sus payasadas y locuras, por quererme y respetarme siempre como su hermana MAYOR, y por ser los mejores hermanos del mundo mundial, los amo (aunque no se los diga).

A mi novio, mi grandote con panza, por estar siempre a mi lado ayudándome en todo, por hacerme reír siempre con tus boberías y payasadas, por comprenderme y apoyarme siempre en cada locura que se me ocurre, porque a pesar de la distancia que hemos tenido estemos más unidos que nunca, por todo y por ser como eres, gracias, te amo.

A mi compañero Cristhian, por siempre estar disponible, por comprenderme y ayudarme siempre, por no perder nunca la paciencia, por trabajar con esfuerzo y esmero, gracias.

A las chicas mis compañeras, confidentes y buenas amigas Rachelo, Bety y Yi, por compartir tantas risas, cuentos, chismes, por todas las noches que nos pasamos "estudiando", por informarme siempre de todo, yo que nunca me entero de nada, por cubrirme siempre en los turnos de clase que estaba de "cuartelera", por saber comprenderme y entregarme su amistad, me siento muy feliz y agradecida de tener amigas como ustedes, muchas gracias PP.

A mis amigos José Luis (Kakashi) y Lizander (Kabito), por siempre estar jodiendo y pidiendo café, por invitarme siempre a "algún party", por hacerme reír, por esas comidas tan ricas que hace José y enseñarme dos o tres cositas de mago, "el mejor mago UCI" dice él, gracias por su amistad.

A un amigo que me ayudó muchísimo en mis primeros años de carrera Alien Góngora Rodríguez, no se me olvida las noches que te pasaste explicándome programación o cálculo o discreta o álgebra o lo que fuera, por ser un buen amigo, un confidente, un gran apoyo, por todas nuestras charlas locas y nuestras risas y aunque ya me abandonaste siempre te tengo presente, miles de gracias.

A mis tutores Arturo y Jessie, por apoyarnos y dedicarnos tiempo durante el desarrollo de la tesis, por no dejarnos nunca desamparados, muchas gracias.

En fin agradezco a todos y a cada una de las personas que me dieron su apoyo para que me fuera posible alcanzar esta meta. Gracias.

Cristhian

Muchas fueron las personas que de una forma u otra colaboraron con la realización de este trabajo de diploma y me ayudaron a salir adelante y ver concluida la realización de este sueño.

A mis compañeros de aula, que aunque no hemos compartido la mayor parte del tiempo este curso, siempre me apoyaron en los momentos cruciales especialmente a Lázaro y Jonathan.

A mi preciada compañera de tesis agradezco todo el trabajo y la colaboración que hicimos.

A mis tutores por apoyarnos y guiarnos por el buen camino.

A mi novia Cecilia, que siempre estuvo a mi lado apoyándome y mucho de lo que he logrado ha sido gracias a ella.

A su familia por el cariño que me han brindado.

Les doy las gracias a mi hermano, mi mamá, mi papá, mis tías que son la razón por la que sigo adelante día a día en la senda de la profesionalidad.

Resumen

XAVIA PACS-RIS, es una plataforma diseñada para ofrecer al personal médico que labora en los departamentos de diagnóstico por imágenes, un conjunto de herramientas, para la visualización y procesamiento de imágenes médicas. Cuenta con cuatro componentes que conforman la solución de *software*, ellos son: PACS Server, PACS Reporter, PACS Web Viewer y el PACS Viewer el cual cuenta con diferentes componentes entre los que se encuentra el PACS Burner, que permite el grabado de estudios médicos completos en dispositivos externos sea CD, DVD o memoria USB, pero no permite la gestión del grabado, ni tampoco permite adjuntar un informe o grabar estudios con identidad anónima para conservar la información. Este trabajo presenta el desarrollo de una herramienta, para la especialidad de Radiología, en el que su propósito es proveer al médico de una herramienta de trabajo personalizada a las necesidades existentes, el grabado de estudios imagenológicos que permitan la anonimización de datos, manejo y gestión de la información y adjuntar informes al grabado de dichos estudios.

Para su desarrollo se realizó el análisis de los procesos de negocio. Se evaluaron sistemas existentes con objetivos similares, así como las tendencias tecnológicas para su implementación. El desarrollo de esta herramienta se encuentra guiado por la metodología de desarrollo AUP-UCI, se basa en tecnologías libres, multiplataforma y sobre el patrón arquitectónico MVVM. Utiliza como lenguaje de programación *C#*.

Como resultados del trabajo de diploma se obtienen nuevas funcionalidades que se integran al componente PACS Viewer, se apoyan los procesos de grabado de estudios imagenológicos y seguridad de la información del paciente.

Índice

Índice.....	VIII
Índice de Tablas.....	X
Índice de Figuras.....	XI
Introducción.....	1
Capítulo 1: Fundamentación teórica.....	5
1.1 Grabado en discos ópticos.....	5
1.1.1 Estándar de grabado.....	5
1.1.2 Bibliotecas para el grabado en media.....	5
1.1.3 Selección de la información relevante.....	6
1.2 Estándar DICOM.....	6
1.2.1 Ficheros DICOM.....	8
1.3 Anonimizar datos.....	10
1.3.1 Técnicas de anonimización de datos.....	10
1.4 Estado del Arte.....	11
1.4.1 Ámbito Internacional.....	12
1.4.2 Ámbito Nacional.....	13
1.5 Ambiente de desarrollo.....	15
1.5.1 Metodología de desarrollo del software.....	15
1.5.2 Entorno de desarrollo integrado.....	16
1.5.3 Lenguaje de programación.....	16
1.5.4 Lenguaje de modelado.....	17
1.5.5 Herramienta CASE.....	17
1.5.6 Interfaz de programación de aplicaciones (API).....	17
1.5.7 Gestor de Base de Datos.....	18
1.5.8 Librería MyDICOM.NET.....	18
Conclusiones del capítulo.....	18
Capítulo 2: Propuesta de solución.....	19
2.1 Características de la propuesta de solución.....	19
2.1.1 Descripción general de la solución.....	19
2.2 Modelado de los procesos del negocio.....	20
2.2.1 Descripción de requisitos por proceso.....	20
2.3 Análisis y diseño.....	28
2.3.1 Arquitectura de <i>software</i>	28
2.3.2 Patrones de diseño.....	29

2.3.3	Diagrama de paquetes	31
2.3.4	Diagrama de clases de diseño	32
	Conclusiones del capítulo	34
Capítulo 3:	Implementación y pruebas	35
3.1	Modelo de datos.....	35
3.2	Estrategias de reutilización.....	36
3.3	Tratamiento de errores.....	36
3.4	Estándares de codificación	37
3.5	Integración al XAVIA PACSViewer.....	38
3.6	Pruebas de <i>software</i>	38
3.6.1	Método de prueba de caja blanca.....	39
3.6.2	Métodos de pruebas de caja negra.....	40
	Conclusiones del Capítulo.....	45
Conclusiones	46
Referencias Bibliográficas	47
Anexos	51
Anexo 1.	Property CantImageCopy.....	51
Anexo 2.	Vista principal del PACSBurner.....	52
Anexo 3.	Vista en el visor Lite de los datos antes de ser anonimizados.....	53
Anexo 4.	Ventana que se muestra para validar que los datos han sido exportados correctamente.	53
Anexo 5.	Vista en el Explorador de Archivos del estudio copiado a la ruta seleccionada.....	54
Anexo 6.	Ventana que se muestra para validar que los datos han sido grabados correctamente.	55
Anexo 7.	Ventana Grabar al finalizar el grabado.....	55
Anexo 8.	Vista en el Explorador de Archivos en el CD, del estudio grabado.....	56
Anexo 9.	RF1 Grabar CD/DVD extraído del Expediente de proyecto Desarrollo de XAVIA PACS-RIS ..	57
Anexo 10.	RF2 Exportar a dispositivo extraído del Expediente de proyecto Desarrollo de XAVIA PACS-RIS .	58

Índice de Tablas.

TABLA 1: ANÁLISIS DE LOS SISTEMAS.	14
TABLA 2. REQUISITOS FUNCIONALES.	20
TABLA 3. DESCRIPCIÓN DEL REQUISITO "ADJUNTAR INFORME "	21
TABLA 4. DESCRIPCIÓN DEL REQUISITO "SELECCIONAR IMÁGENES A GRABAR/EXPORTAR "	22
TABLA 5. DESCRIPCIÓN DEL REQUISITO "SELECCIONAR LOS DATOS A ANONIMIZAR DEL PACIENTE "	22
TABLA 6. DESCRIPCIÓN DE REQUISITO NO FUNCIONAL. ATRIBUTO DE CALIDAD: SEGURIDAD.	23
TABLA 7. DESCRIPCIÓN DE REQUISITO NO FUNCIONAL. ATRIBUTO DE CALIDAD: USABILIDAD.	24
TABLA 8. DESCRIPCIÓN DE REQUISITO NO FUNCIONAL. ATRIBUTO DE CALIDAD: EFICIENCIA DE DESEMPEÑO.	25
TABLA 9. DESCRIPCIÓN DE REQUISITO NO FUNCIONAL. ATRIBUTO DE CALIDAD: MANTENIBILIDAD.	27
TABLA 10. DESCRIPCIÓN DE REQUISITO NO FUNCIONAL. ATRIBUTO DE CALIDAD: PORTABILIDAD.	27
TABLA 11. PRUEBA DE UNIDAD UTILIZANDO COMPLEJIDAD CICLOMÁTICA AL MÉTODO PREPARAR DATOS QUEMADOR().	39

Índice de Figuras.

FIGURA 1. EJEMPLO DE VISTA DE ÁRBOL.	6
FIGURA 2. CALIDAD DEL PRODUCTO DE SOFTWARE.	23
FIGURA 3. FUNCIONAMIENTO DE LA ARQUITECTURA MVVM.	29
FIGURA 4. FRAGMENTO DE CÓDIGO DE LA CLASE <i>CONNECTIONMANAGER</i>	30
FIGURA 5. FRAGMENTO DE CÓDIGO DE LA CLASE <i>MAINVIEWMODELS</i>	31
FIGURA 6. FRAGMENTO DE CÓDICE DE LA CLASE <i>HEADERTOLMAGECONVERTER</i>	31
FIGURA 7. DIAGRAMA DE PAQUETES.	32
FIGURA 8. DIAGRAMA DE CLASES DE DISEÑO GRABAR/EXPORTAR.	33
FIGURA 9. MODELO DE DATOS.	36
FIGURA 10. REPRESENTACIÓN DEL TRATAMIENTO DE ERRORES.	37
FIGURA 11. VISTA DEL BOTÓN EXPORTAR DE LA BANDEJA DE CASOS DEL <i>PACSVIEWER</i>	38
FIGURA 12 . VISTA DEL ESTUDIO RADIOLÓGICO EN LA HERRAMIENTA <i>PACSBURNER</i>	41
FIGURA 13. RESULTADO DESPUÉS DE BORRAR LA INFORMACIÓN DESEADA.	42
FIGURA 14. VISTA DE INFORMACIÓN A ANONIMIZAR.	42
FIGURA 15 . VISTA EN EL VISOR LITE DE LOS DATOS ANONIMIZADOS.	43
FIGURA 16. VISTA AL PRESIONAR EL BOTÓN EXPORTAR.	44
FIGURA 17. VISTA DE LA VENTANA GRABAR DE LA HERRAMIENTA <i>PACSBURNER</i>	45
FIGURA 18. FRAGMENTO DE CÓDIGO DE LA PROPERTY <i>CANTIMAGECOPY</i>	51
FIGURA 19. VISTA DE LA VENTANA PRINCIPAL DE LA HERRAMIENTA <i>PACSBURNER</i>	52
FIGURA 20. DATOS DE UN ESTUDIO RADILÓGICO OBSERVADO EN UN VISOR LITE.	53
FIGURA 21. EXPORTADO CORRECTAMENTE.	53
FIGURA 22. ESTUDIO COPIADO EN LA RUTA SELECCIONADA.	54
FIGURA 23. GRABADO CORRECTAMENTE.	55
FIGURA 24. VENTANA GRABAR AL FINALIZAR EL GRABADO.	55
FIGURA 25. ESTUDIO COPIADO EN EL CD.	56
FIGURA 26. RF1 GRABAR CD/DVD.	57
FIGURA 27. RF2 EXPORTAR A DISPOSITIVO.	58

Introducción

El sector de la salud, ha tenido numerosos beneficios tras el avance de las Tecnologías de la Información y las Comunicaciones, dando paso al campo multidisciplinario de la Informática Médica, encargado del perfeccionamiento de los procesos de gestión de información y calidad en la salud(Katz 2009). Impulsada por la digitalización de la imagen y la adquisición de equipos de alta tecnología en los centros hospitalarios, surge la radiología digital, dando lugar a la Imagenología como especialidad médica dedicada al análisis y diagnóstico de patologías a partir de imágenes, generadas al incorporarse varias modalidades radiológicas con adquisición digital: tomografía axial computarizada(Amieiro Paz 2018), imagen por resonancia magnética(Geosalud 2014), ecografía(Schmidt 2008), medicina nuclear(Programa Teórico 2012)y radiografía computarizada(Vergara E, Sepúlveda R y Vega T 2006). Dicho surgimiento llevó al desarrollo de sistemas informáticos, debido a la necesidad de optimizar los flujos de trabajo en las áreas de diagnóstico por imágenesy a la creación de estándares que permitieran la interoperabilidad entre los equipos imagenológicos.

Para la lograr la comunicación entre los sistemas médicos, el Colegio Americano de Radiología (ACR por sus siglas en inglés) y la Asociación Nacional de Fabricantes de Material Eléctrico (NEMA por sus siglas en inglés), formaron un comité conjunto en el que se desarrollaría un método estándar que permitiera el desarrollo y expansión de los Sistemas para el Almacenamiento y Transmisión de Imágenes Digitales (PACS, por sus siglas en inglés). Dicho método se crearía con el fin de gestionar, controlar y almacenar las imágenes digitales obtenidas desde equipos de imagenología de nueva generación, dando lugar a la creación del estándar DICOM (WD Bidgood y SC Horii 1992).

DICOM es un conjunto de normas que convierte la comunicación y el almacenamiento de las informaciones médicas en un formato electrónico único. Este proporciona servicios para la transmisión, visualización, almacenamiento de imágenes médicas y la información asociada a estas entre equipos de diferentes proveedores.Actualmente el estándar DICOM está en su tercera versión y busca facilitar el desarrollo y expansión de los sistemas PACS, convirtiéndose así en el estándar que los rige, debido a que permite que estos sistemas funcionen correctamente con modalidades y estaciones de trabajo de diferentes fabricantes, además de que es el estándar por excelencia para el trabajo con las imágenes médicas digitales. Los sistemas PACS permiten el almacenamiento digital, transmisión y descarga de imágenes radiológicas. Estos sistemas se componen de partes *software* y *hardware*, que directamente se comunican con modalidades y obtienen las imágenes de éstas. Las imágenes son transferidas a una estación de trabajo (*workstation*) para su visualización y emisión de informes radiológicos(Mildenberger, Eichelberg y Martin 2002).

Los PACS ofrecen una alternativa para el manejo de imágenes digitales a gran escala debido a que se obtiene una mejoría sustancial en la precisión con la elaboración de los informes, así como la posibilidad de que el departamento radiológico mejore su efectividad, pero no gestionan el flujo de trabajo del Departamento de Radiología, de esta forma, fueron creados los Sistemas de Información Radiológica (RIS, por sus siglas en inglés) con el objetivo de facilitar las herramientas para la organización de los servicios que ofrecen las áreas de diagnóstico por imágenes, y evolucionó hasta convertirse en la solución informática por excelencia para complementar al PACS. Estas características han logrado que los PACS y RIS sean usados en diversos países como Venezuela, España, Angola entre otros, incluyendo Cuba (Piqueras, Carreño y Lucaya 1993) (Daudinot López y Miller Clemente 2016).

En Cuba, uno de los objetivos del proyecto de informatización de la sociedad es el desarrollo de aplicaciones informáticas. Entre las principales instituciones que trabajan en el país para este fin, se encuentra el Centro de Informática Médica (CESIM), de la Universidad de las Ciencias Informáticas (UCI), que desarrolla la plataforma XAVIA PACS-RIS. Esta plataforma está desarrollada para ofrecer al personal médico que labora en los servicios de diagnóstico por imágenes un conjunto de herramientas de propósito general para la visualización y procesamiento de las imágenes médicas digitales y posterior edición de los informes emitidos. XAVIA PACS-RIS está formado por varios módulos integrados, compatibles con el estándar internacional DICOM, los cuales ofrecen una solución escalable y adaptable a los requerimientos de distintos hospitales. Estos módulos son: PACS Server (Servidor de Imágenes Médicas), PACS Reporter (Herramienta para la Edición de Informes Diagnósticos), PACS Web Viewer (Visor Web de Imágenes Médicas) y PACS Viewer (Estación de Diagnóstico General) el cual cuenta con diferentes componentes entre los que se encuentra el Visor DICOM, Configuración y el PACS Burner, que permite el grabado de estudios médicos completos en dispositivos externos sea CD, DVD o memoria USB.

En las clínicas imagenológicas cubanas, en el proceso de realización de los estudios médicos, la entrega de las imágenes a los pacientes se ha convertido en una práctica de la atención. Para lograr la entrega de estos estudios imagenológicos se requieren medios de almacenamiento de bajo costo para la institución y que a la vez facilite a los pacientes la disponibilidad de la información ya sea en un CD, DVD o un dispositivo USB, en los cuales también se almacena un Visor Lite.

Los estudios imagenológicos pueden tener múltiples imágenes sin valor diagnóstico, reconstrucciones 3d, topogramas, calibraciones, además de otros archivos de configuración. Estos estudios y su información asociada, actualmente no se gestiona de forma correcta, ya que la herramienta PACS Burner que realiza el proceso de grabado en los estudios imagenológicos, en su versión actual, no cuenta con opciones para la gestión del grabado, de esta manera se ocupa innecesariamente

espacio en los medios de almacenamiento utilizados, atentando con los gastos de materiales en la institución hospitalaria y la separación de la información clínica de valor diagnóstico. Otra limitante que posee la versión actual del PACSBurner está relacionada a la imposibilidad de adjuntar el informe o grabar estudios con identidad anónima que se genera por los especialistas de radiología para conservar la información.

Lo anterior limita la evolución de la herramienta PACSBurner debido a que en su estado actual no propicia la conservación de la información de los estudios imagenológicos realizados a los pacientes con la información requerida y la seguridad debida.

Teniendo en cuenta la problemática antes descrita, se identifica como **problema a resolver**: ¿cómo propiciar la selección de información relevante para el proceso de grabado y anonimización de estudios imagenológicos en el PACSBurner? A partir del problema identificado anteriormente se plantea como **objeto de estudio**: el proceso de grabado de información en dispositivos de almacenamiento CD, DVD o USB.

Para dar solución al problema planteado, se define como **objetivo** de la presente investigación: desarrollar la evolución de la herramienta PACSBurner para propiciar la selección de información relevante en el proceso de grabado y anonimización de estudios imagenológicos. Lo anterior propicia enmarcar la investigación en el **campo de acción**: herramientas para la selección de información relevante en el grabado y anonimización de estudios imagenológicos.

El objetivo general se desglosa en las siguientes **tareas de investigación**:

- Elaboración de los fundamentos teóricos y metodológicos de la investigación asociados al proceso de grabado y anonimización de imágenes DICOM.
- Análisis de los principales estándares, herramientas y bibliotecas de clases existentes en la actualidad, para el grabado en CD, DVD y USB.
- Desarrollo del componente PACSBurner para el proceso de grabado y anonimización de imágenes DICOM para el sistema XAVIA PACS 3.0
- Integración de la propuesta de solución al Sistema XAVIA PACS.
- Validación del resultado obtenido a partir de los métodos definidos para la investigación.

Métodos Teóricos:

Método Histórico-Lógico: el uso de este método permite analizar el surgimiento y evolución de los procesos de grabación de información en dispositivos de almacenamiento y las herramientas informáticas para el grabado de estudios imagenológicos, de manera selectiva y anonimizada.

Analítico-sintético: la utilización de este método permite realizar un estudio de las teorías y documentos más relevantes sobre la gestión del grabado de estudios imagenológicos, de manera selectiva y anonimizada, permitiendo así, extraer los elementos más importantes sobre los mismos.

Modelación: el uso de este método permite el modelado de los diagramas de casos de usos del sistema y del modelo conceptual a desarrollar.

Métodos Empíricos:

Consulta de fuentes de información: se emplea para la elaboración del marco teórico de la investigación y permite realizar un estudio actual de las distintas bibliografías que existen del tema tratado.

Observación: este método permite definir las características comunes de los sistemas que permiten la gestión del grabado y la anonimización de los estudios imagenológicos.

El documento estará estructurado en tres capítulos organizados de la siguiente forma:

- **Capítulo 1. Fundamentación teórica:** en este capítulo se definen los elementos teóricos necesarios para el desarrollo de la investigación y los principales conceptos que se emplearán durante el trabajo. Se realiza un análisis de las soluciones similares y se selecciona la metodología de desarrollo de *software*, las herramientas y tecnologías que se deben utilizar para el desarrollo de la propuesta de solución.

- **Capítulo 2. Propuesta de solución:** en este capítulo se describe la propuesta de solución y los artefactos ingenieriles que se desarrollaron en la fase de planificación. Además, se presenta la arquitectura de *software* del componente desarrollado.

- **Capítulo 3. Implementación y pruebas:** en este capítulo se explica el estándar de codificación utilizado en la implementación del componente y se presentan las tareas ingenieriles desarrolladas. Además, se presenta el resultado de las pruebas realizadas a la aplicación para validar su correcto funcionamiento.

Capítulo 1: Fundamentación teórica

En este capítulo se definen los principales conceptos que estarán presentes en la investigación, tales como: que es un *software* de grabado de discos ópticos, que es anonimizar y algunas de sus técnicas y la descripción del estándar DICOM. Se realiza un estudio del estado del arte, tanto en el ámbito nacional como internacional y algunos sistemas similares al que se desarrollará. Además, se seleccionan las herramientas y tecnologías a utilizar para el desarrollo de la propuesta de solución.

1.1 Grabado en discos ópticos

Para realizar el grabado en discos ópticos CD, DVD, HD DVD y Blu-Ray, se requiere de una unidad de discos especial denominada "quemadora", así como discos escribibles o re-escribibles. Los datos a escribir pueden ser compilados por el usuario en el propio momento de la grabación o venir incluidos en algún fichero que contenga la estructura exacta y reproducible de un medio físico (soporte original). Este tipo de ficheros es lo que habitualmente se conoce como una imagen de un cd o dvd. Aunque es lo más habitual, una imagen no tiene por qué contener los datos de un CD/DVD, sino que puede contener cualquier estructura de ficheros compatible (Harris, Demaree y Lu 2011).

1.1.1 Estándar de grabado

Formato Universal del Disco (UDF, por sus siglas en inglés) es un estándar de sistema de archivos de CD-ROM y DVD desarrollado como un medio para garantizar la coherencia entre los datos escritos en varios medios ópticos, al facilitar tanto el intercambio de datos como la implementación del estándar ISO / IEC 13346. Se requiere UDF para DVD-ROM y DVD lo utiliza para contener transmisiones de audio/video MPEG. Originalmente desarrollado como un reemplazo para las especificaciones del sistema de archivos en el estándar original de CD-ROM, ISO 9660, UDF es utilizado por CD-R y CD-RW en un proceso llamado escritura de paquetes que hace que la escritura de CD sea más eficiente en términos de tiempo y se requiere espacio en disco (Bjork et al. 2005).

1.1.2 Bibliotecas para el grabado en media

IMAPI, la interfaz de programación de aplicaciones de grabación de imágenes (IMAPI por sus siglas en inglés), es un componente del sistema operativo *Microsoft Windows* utilizado para el grabado de discos CD o DVD. La primera versión de IMAPI (IMAPIv1) fue expuesta por primera vez en *Windows XP* y fue completamente rediseñada en su segunda versión (IMAPIv2) para *Windows Vista* y *Windows Server 2008*. IMAPI permite a una aplicación organizar y quemar audio y datos en un dispositivo CD-R y CD-RW. Los formatos específicos que puede soportar son *Redbook* de datos y *Joliet* de audio, ambos con ISO 9660. IMAPI cuenta con varios escenarios que demuestran su gran utilidad, algunos de estos son: comprobar la compatibilidad de la unidad, comprobar los medios, grabar una imagen de disco, adicionar una imagen de arranque, crear discos multisesión, seguir

procesos con eventos, prevenir cierre de sesión o de suspensión durante una grabación y proporcionar los permisos de usuario para el quemado en media (McMurdie, Polfer y Evers 2005).

BurnLib, es una librería de grabado a dispositivos extraíbles CD, DVD o USB, desarrollada y usada por el equipo de desarrollo del sistema XAVIA PACS. El contenido de esta librería brinda funcionalidades que permiten grabar o borrar unidades ópticas (CD/DVD) mediante el acceso a la unidad de disco denominada "quemadora", también facilita verificar varios parámetros y propiedades de estos medios físicos como son: la velocidad de lectura, el espacio disponible y el tipo de dispositivo insertado. Dadas las funcionalidades y facilidades que ofrece esta librería será usada para complementar el desarrollo de la herramienta PACSBurner, además de que es la propuesta por el equipo de desarrollo del sistema XAVIA PACS.

1.1.3 Selección de la información relevante

Vista de árbol (**tree view** en inglés) es un elemento de interfaz gráfica de usuario que representa una vista jerárquica de información. Cada elemento llamado rama o nodo, puede tener una serie de sub elementos que pueden ser visualizados como tabulado en una lista, ver Figura 1. Esto le permite al usuario la selección y visualización de las imágenes que desee grabar o exportar.

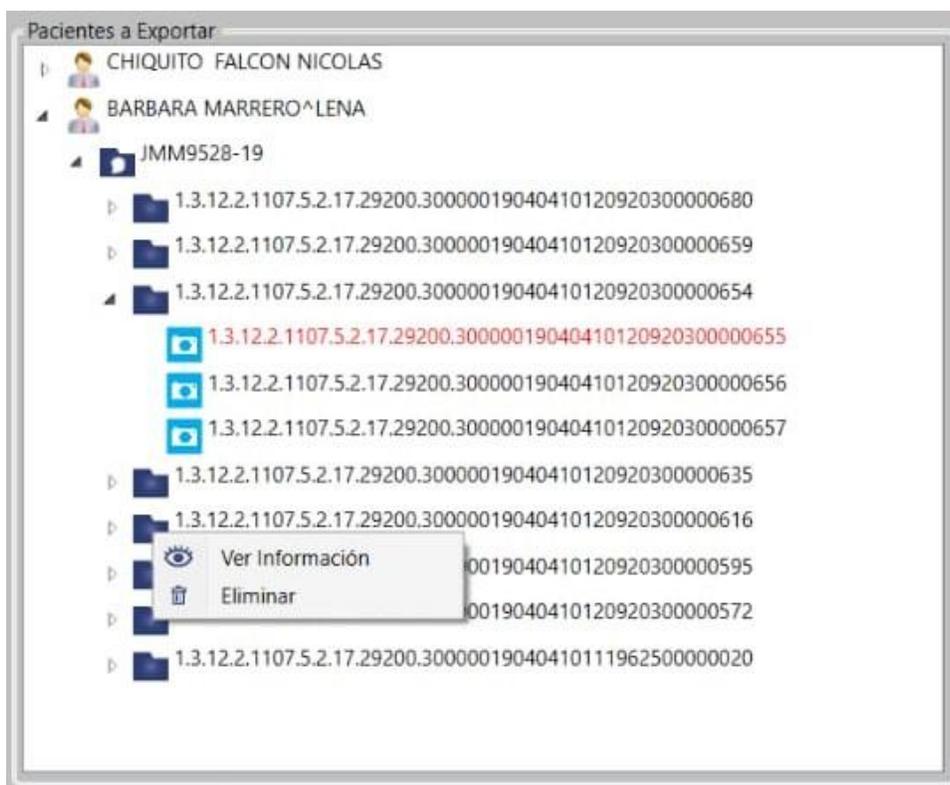


Figura 1. Ejemplo de vista de árbol.

1.2 Estándar DICOM

El estándar de comunicación más aceptado y especializado en entornos de imágenes médicas es el DICOM 3.0. Sus antecedentes aparecen en 1985 con un producto llamado ACR-NEMA 1.0, ya que

fue desarrollado por el ACR en colaboración con la NEMA. La evolución de este producto dio lugar a ACR-NEMA 2.0 en 1988. En 1993 aparece ya con el nombre de DICOM 3.0. En su desarrollo participan varias instituciones de la comunidad internacional como el Comité Europeo de Normalización (CEN por sus siglas en inglés) y la Industria Japonesa de Equipos de Radiología (JIRA por sus siglas en inglés). Esta versión es considerada como un estándar completo, compatible con las versiones anteriores. Según (Mildenberger, Eichelberg y Martín 2002)(Clunie 2000) los objetivos fundamentales de DICOM son:

- Promover la comunicación de imágenes digitales sin importar el fabricante del dispositivo.
- Facilitar el desarrollo y expansión de los PACS que puedan interconectarse con otros sistemas de información de un hospital.
- Permitir la creación de bases de datos de información de diagnóstico que puedan ser consultadas por una amplia variedad de dispositivos distribuidos geográficamente.

El estándar DICOM 3.0 cuenta con varias partes, que a pesar de estar relacionadas, son documentos independientes que se modifican eventualmente en dependencia de las necesidades. El contenido abordado en el presente trabajo tiene sus principales fundamentos en las siguientes partes del estándar, debido a que estas partes permiten el almacenamiento de información de las imágenes médicas en dispositivos extraíbles, la interoperabilidad entre ellas y facilita el intercambio de información entre diferentes dispositivos; estas partes son:

- **PS 3.10: *Media Storage and File Format for Data Interchange*:** Especifica un modelo general para el almacenamiento de información de imágenes médicas en dispositivos extraíbles. El propósito de esta parte es proporcionar un marco de trabajo que permita el intercambio de varios tipos de imágenes médicas e información relacionada sobre una amplia gama de dispositivos de almacenamiento físico (National Electrical Manufacturers Association 2009).
- **PS 3.11: *Media Storage Application Profiles*:** Establece los subconjuntos específicos de la aplicación del estándar al cual una aplicación puede demandar conformidad. Estos subconjuntos son referidos como Perfiles de Aplicación. Tal declaración de conformidad se aplica al intercambio interoperable de imágenes médicas y de la información relacionada sobre los medios de almacenamiento para usos clínicos específicos (National Electrical Manufacturers Association 2009).
- **PS 3.12: *Storage Functions and Media Formats for Data Interchange*:** Facilita el intercambio de información entre aplicaciones en un entorno médico. Describe la relación entre el modelo de almacenamiento y cada uno de los dispositivos físicos permitidos por el estándar (National Electrical Manufacturers 2009).

Representación de valor (VR):

La representación del valor de un elemento de datos que describe el tipo de datos y el formato de los valores de ese elemento de datos. PS3.6(NEMA 2013a) enumera la VR de cada elemento de datos por etiqueta de elemento de datos. Los valores con VR contruidos con cadenas de caracteres, excepto en el caso de la IU de VR, se rellenarán con caracteres de ESPACIO (20H, en el Repertorio de caracteres predeterminado) cuando sea necesario para lograr una longitud uniforme. Los valores con una VR de IU se rellenarán con un solo carácter *NULL* (00H) final cuando sea necesario para lograr una longitud uniforme. Los valores con un VR de OB se rellenarán con un solo valor de byte *NULL* final (00H) cuando sea necesario para lograr una longitud uniforme. Todos los nuevos VR definidos en futuras versiones de DICOM deberán tener la misma estructura de elementos de datos que se define en la Sección 7.1.2(NEMA 2013c) (es decir, seguir el formato para VR como OB, OW, SQ y UN) (NEMA 2013a).

1.2.1 Ficheros DICOM

Estos ficheros contienen las características que distinguen las imágenes DICOM, como: formato de archivo, identificadores únicos de cada imagen, y otros ficheros que permiten el acceso a los datos de la imagen, permitiendo su modificación, eliminación y creación.

DICOM File

Una de las características más importantes que distinguen a DICOM de otros estándares es la posibilidad de contener en un mismo fichero tanto imágenes como información asociada a estas; para lograrlo DICOM define un formato de archivo (*DICOM File*). Desde el punto de vista del implementador, un fichero DICOM se puede dividir en cuatro partes diferenciadas:

- Preámbulo y prefijo identificativo del fichero.
- Meta-cabecera.
- Cabecera.
- Imagen; aunque desde el punto de vista del formato, la imagen es un elemento más de la cabecera.

El preámbulo tiene un tamaño fijo de 128bytes, y está pensado para tener un uso definido por la implementación. Puede contener información sobre el nombre de la aplicación usada para crear el fichero, o información que permita a aplicaciones acceder directamente a los datos de la imagen almacenada en el fichero. En caso de no ser usado, el preámbulo debe estar presente, con todos sus bytes puestos al valor 00h.

El prefijo consiste en cuatro bytes que contienen la cadena de caracteres "DICM". Esta cadena debe estar codificada siempre con las letras en mayúscula. El propósito de este prefijo es permitir a las implementaciones diferenciar si un fichero es DICOM o no.

La cabecera y la meta-cabecera de un fichero DICOM consisten en una serie de campos con toda la información necesaria sobre la imagen en cuestión, incluyendo la propia imagen. Entre estos campos se encuentran, por ejemplo, datos sobre el paciente (nombre, sexo), sobre el tipo de imagen, entre otros, y campos que contienen información necesaria para procesar y visualizar la imagen correctamente (NEMA 2013b)(NEMA 2013d).

File-set

Un File-set es una colección de archivos que comparten un espacio común de nombres en el que los identificadores de los archivos (*File ID*) son únicos. Cada *File-set* contiene varios ficheros, entre los que está un fichero con ID *DICOMDIR* que guarda la información correspondiente al *File-set* e información inherente a las imágenes médicas.

Cada *File-set* se identifica de forma única por un único identificador (UID) que se registrará de acuerdo con las normas de registro UID. Según esta norma cuando los archivos se agregan o quitan de un *File-set*, el identificador no deberá modificarse.

Un *File-set* también puede ser identificado por un ID, que proporciona una simple referencia legible por humanos. Un ID es una cadena de cero (0) a dieciséis (16) caracteres (NEMA 2013d)(NEMA, DICOM 2013b)(NEMA, DICOM 2013a).

File DICOMDIR

Un único fichero con *File ID*, *DICOMDIR*, existe como miembro de cada *File-set* y contiene representada la organización de la información en el *File-set*, debe residir en el directorio raíz de la jerarquía de directorios. Si el archivo *DICOMDIR* no existiese en un *File-set* significa que este no se ajusta a lo establecido en el estándar DICOM. Si el *DICOMDIR* está presente, entonces todos los archivos referenciados en el *DICOMDIR* estarán presentes también. Un ejemplo del uso del archivo *DICOMDIR* es que un CD DICOM de datos contiene un *DICOMDIR* en el nivel raíz el cual contiene una descripción y el acceso a la información de los estudios en el CD (NEMA 2013d)(NEMA, DICOM 2013c).

File-set Creator

Una aplicación cuyas entidades actúan como *File-set Creator* (FSC) es capaz de generar un File-set y asignarle un *File ID*. La aplicación debe ser capaz de crear un fichero *DICOMDIR* con la representación de la organización de los ficheros que están en el *File-set*. Si un conjunto de archivos, por ejemplo, un estudio, no cabe en un CD/DVD, el FSC debe ser capaz de crear múltiples *File-sets*

independientes, de forma tal que cada File-set pueda residir en un solo CD/DVD controlado por su *DICOMDIR*.

El FSC ofrece la capacidad de cerrar el disco con la terminación de la última sesión escrita (ninguna información se puede agregar posteriormente al disco) o de crear varias sesiones (información adicional se puede agregar posteriormente al disco)(NEMA 2013d)(NEMA, DICOM 2013a).

File-set Reader

El File-set Reader (FSR) es uno de los roles de una entidad de aplicación DICOM que accede a uno o más archivos en un File-set a través de operaciones de lectura, este puede leer cero o muchos bytes de contenido de un archivo. Un FSR no puede modificar ninguno de los archivos de un File-set, por consecuente tampoco puede modificar el archivo *DICOMDIR*(NEMA 2013d)(NEMA, DICOM 2013d).

File-set Updater

El rol de File-set Updater (FSU) es utilizado por las entidades que reciben un File-set transferido y lo actualizan. El FSU lee pero no modifica el contenido de cualquiera de los archivos DICOM en un File-set, excepto el archivo *DICOMDIR*, puede además crear archivos adicionales o borrar los archivos existentes en un File-set.

El FSU ofrece la capacidad de cerrar el disco con la terminación de la última sesión o de crear varias sesiones. Si el disco no se ha cerrado, el FSU podrá actualizar la información asumiendo que hay bastante espacio para escribir un nuevo *DICOMDIR*, la información y las estructuras de control fundamentales(NEMA 2013d)(NEMA, DICOM 2013e).

1.3 Anonimizar datos

La finalidad del proceso de anonimización es eliminar o reducir al mínimo los riesgos de reidentificación de los datos anonimizados manteniendo la veracidad de los resultados del tratamiento de los mismos, es decir, además de evitar la identificación de las personas, los datos anonimizados deben garantizar que cualquier operación o tratamiento que pueda ser realizado con posterioridad a la anonimización no conlleva una distorsión de los datos reales. Un análisis masivo de los datos o macrodatos que puedan derivar de los datos anonimizados no debería diferir del análisis que pudiera obtenerse si hubiera sido realizado con datos no anonimizados(Vilches 2018)(Piqueras 2019).El proceso de anonimización cuenta por un conjunto de técnicas, según (Firma-e 2014) y (González 2019) algunas de ellas son las presentes a continuación.

1.3.1 Técnicas de anonimización de datos

Desidentificación: consiste en eliminar del conjunto de datos todos aquellos identificadores directos del individuo. Algunos identificadores directos típicos son el nombre y domicilio, número de teléfono o el número del documento nacional de identidad.

Eliminación de registros: cuando las características de la población de la que se ha obtenido el fichero de datos hacen que determinados individuos puedan ser fácilmente identificables (ejemplo: en un fichero de pacientes de una institución, en donde sólo hay un paciente diagnosticado con un determinado trastorno), una alternativa es eliminar del fichero original estos registros.

Datos derivados: consiste en sustituir un valor que podría permitir la identificación de un individuo (ejemplo: la fecha de nacimiento), por otro valor derivado que presenta menor granularidad, dificultando la individuación (ejemplo: la edad o la reclasificación en valores como joven, maduro, etc..).

Redondeo: consiste en redondear los datos numéricos al alza o la baja, de modo que dificulte la reidentificación.

Seudoanonimización: consiste en la asignación seudónimos a los datos de identificación directa de los individuos. La identificación de esa persona solo la conocen los implicados en esa operación, pero los demás empleados no pueden establecer esa relación.

Existen tres tipos de técnicas de seudoanonimización, ellas son:

- Sustituir el dato original por la versión encriptada del mismo.
- Tokenización: consiste en crear una tabla de códigos arbitrarios que se asocian de manera consistente con los valores originales.
- Aleatorización: se genera un código que puede ser asociado a diferentes valores en el fichero original; es decir, no mantiene la unicidad de los registros.

A partir del estudio antes descrito, se define como técnica de anonimización a utilizar, la seudoanonimización de tipo aleatorización, debido a que genera un código que es asignado al dato de la imagen que se desee anonimizar y asigna valores aleatorios a estos, cumpliendo así con la necesidad de la conservación de los datos de los pacientes de manera fácil y fiable.

1.4 Estado del Arte

En este epígrafe se define el análisis del estado del arte, tanto en el ámbito nacional como internacional y algunos sistemas similares al que se desarrollará, algunas de sus características más importantes, como gestión y anonimización de los datos, requisitos mínimos del sistema y compatibilidad en diferentes plataformas.

A medida que una mayor cantidad de hospitales, utilizan las aplicaciones PACS con el uso de las tecnologías de imágenes digitales, la gestión del almacenamiento se torna cada vez más compleja. Este hecho se agrava debido a que en la mayoría de los casos, se le exigen a los hospitales conservar las imágenes médicas durante más de cinco años. Por otra parte, la entrega de las

imágenes a los pacientes en dispositivos extraíbles como CD, DVD, o memoria USB, se ha convertido en algo necesario en el proceso de realización de estudios imagenológicos; así mismo como la anonimización de los datos de los pacientes ha tomado gran importancia, debido a que estos estudios imagenológicos pueden ser usados para realizar posteriores estudios en distintos centros hospitalarios.

1.4.1 **Ámbito Internacional**

En el mundo existen gran cantidad de servidores y componentes de PACS que realizan la función de grabado. Algunos de estos sistemas son:

OsiriX Imaging Software: visor DICOM exclusivo para el sistema operativo Macintosh, el software es de código abierto. Permite grabar CD's con imágenes médicas, y junto a ellas un Visor Lite llamado "k-PACS" para visualizar las imágenes en entornos que no cuenten con los servicios de un PACS(OsiriX 2010).

Pacslink Eco: tiene adaptaciones especiales en sus pantallas para la operatoria habitual del ecografista. Así un ecógrafo sin salida Dicom puede ser Dicomizado. Variando pequeñas cosas en su configuración permite realizar múltiples tareas, como el envío de estudios, la impresión de los estudios y el grabado en medios extraíbles con visor propio auto ejecutable. A su vez cuenta con un buscador de listas de trabajo (*worklist*) para mayor productividad a menor tiempo(Pacslink 2005).

Dicom Simplex: fue creado por la empresa XPRO Systems más exactamente por la división XPRO MEDICAL. *Dicom Simplex* es una estación económica y de fácil manejo que incluye funciones exclusivas para el grabado de las imágenes y estudios médicos en CD/DVD, entre las que se contemplan: producción de un CD/DVD de paciente, Informe Diagnóstico adjunto, visualizador DICOM y funcionalidad especial para archivar. Puede ser configurado para grabar CD's o DVD's de manera automática con todas las imágenes del estudio, aunque también es posible hacerlo de forma manual con la selección de los datos que se quieran grabar(Cabrejas 2008).

Vitre Core: es un *software* de radiología e incluye características tales como fusión de imágenes, programación de pacientes y entradas de escaneo. Posee una función llamada DICOM *Export* (Exportación DICOM), donde se gestiona el grabado del estudio, y la opción *Anonymize on Export*(Exportación anónima), donde se gestiona la anonimización de los datos(Vitre 2016).

eFilm: es una aplicación utilizada para la visualización y manipulación de imágenes médicas. Utilizando este programa pueden mostrarse, analizarse, procesarse, almacenarse y comunicarse a través de redes de ordenadores las imágenes digitales y datos de diversas fuentes. Realiza funciones como la gestión del grabado en media, la integración con el archivo de la nube del *iConnect* e impresión de DICOM(Merge 2019).

1.4.2 **Ámbito Nacional**

Cuba, se ha desarrollado mundialmente en el área de la medicina, ello lo demuestra el estado de salud alcanzado por la población cubana. Son muchos los recursos que el estado cubano pone a disposición del Sistema Nacional de Salud, para mejorar los servicios que se brindan en todos los centros hospitalarios y de atención médica.

En el proceso de informatización de la sociedad cubana, se han adquirido nuevos equipos médicos que garantizan diagnósticos rápidos y fiables. Sufragar los gastos de la implantación de un sistema, de los existentes en el mundo, para la gestión de la información generada por estos equipos es muy difícil para países subdesarrollados como Cuba, es por ello que se han desarrollado soluciones propias para facilitar el proceso de diagnóstico, como la creación de los sistemas:

IMAGIS, desarrollado por el Centro de Biofísica Médica para la telemedicina y la transmisión de imágenes médicas multimodales en el Sistema Nacional de Salud Cubano. Este sistema posibilita la adquisición, procesamiento, almacenamiento, recuperación e impresión de imágenes médicas a través del estándar DICOM 3.0. El almacenamiento en CD se realiza exportando las imágenes médicas y los ficheros del visor hacia una carpeta, solamente se gestiona la carpeta. Versiones posteriores permitían organizar las imágenes en varios volúmenes, que se guardaban en carpetas previamente seleccionadas; pero el almacenamiento y la organización siguen siendo de forma manual (Daudinot López y Miller Clemente 2016) («La magia del Imagis» 2018).

XAVIAPACS, desarrollado por el centro CESIM de la Universidad de Ciencias Informáticas (UCI), está diseñado para ofrecer al personal médico que labora en los departamentos de diagnóstico por imágenes, un conjunto de herramientas de propósito general, para la visualización y procesamiento de imágenes médicas y posterior edición de los informes que son emitidos, facilitando además el acceso a las imágenes desde las estaciones diagnósticas. Este cuenta con cuatro componentes que conforman la solución de *software*, ellos son:

PACS Server (Servidor de imágenes médicas), posibilita la gestión de la información de los estudios que se generan en las diferentes modalidades diagnósticas, soporta asociaciones simultáneas y garantiza el almacenamiento de cada uno de estos estudios de forma ordenada.

PACS Reporter (Herramienta para la edición de informes diagnósticos), permite construir un informe del estudio médico imagenológico realizado a un paciente, este es almacenado en el servidor de informes del sistema XAVIA RIS.

PACS Web Viewer (El visor web de imágenes médicas), permite la visualización y manipulación de imágenes DICOM, a través del uso de la web, trayendo consigo la facilidad de acceso y uso para puestos remotos.

PACSViewer(Estación de diagnóstico general), con potencialidades para el procesamiento, análisis y visualización de las imágenes médicas con herramientas básicas o de post procesamiento 3D. Este componente del sistema permite la conexión remota desde las estaciones de trabajo hasta el servidor de imágenes del hospital, la generación de informes imagenológicos, la exportación a formatos comunes de imágenes, videos digitales y la impresión DICOM. Está compuesto por varias herramientas que permiten la visualización y edición de las imágenes, entre ellas se encuentra el PACSBurner, que permite el grabado de estudios médicos completos en dispositivos externos sea CD, DVD o memoria USB, incluyendo un visor lite, que permite la visualización de las imágenes médicas sin necesidad de tener instalado la plataforma XAVIA PACS-RIS. A pesar de las funcionalidades actuales de la herramienta PACSBurner, se hace necesario su actualización, ya que esta no permite la gestión del grabado, ni la gestión de la anonimización de los datos del paciente y tampoco permite adjuntar un informe. En la Tabla 1 se puede observar el análisis de estos sistemas.

Tabla1: Análisis de los sistemas.

Sistema	Acceso al código fuente	Gestión del grabado	Gestión de la anonimización	Adjuntar Informe
OsiriX Imaging	Sí	Sí	No	Sí
Pacslink Eco	No	No	No	No
Dicom Simplex	No	Sí	No	Sí
Vitrea Core	No	Sí	Sí	Sí
eFilm	No	Sí	No	Sí
IMAGIS	Sí	No	No	No
XAVIA PACS-RIS	Sí	No	No	No

Teniendo en cuenta las características deseadas para la herramienta PACSBurner, los sistemas expuestos anteriormente no cumplen con los requisitos estipulados, impidiendo la posibilidad de adjuntar un informe, la gestión del grabado de los datos de los estudios imagenológicos o la anonimización de los datos de dichos estudios. Algunas de ellas no presentan acceso al código fuente lo cual dificulta el desarrollo y modificación del *software* según las necesidades que se requieren, tampoco cumplen con las políticas de seguridad tecnológica en Cuba o no se integran al PACS, haciendo imposible el uso de uno de estos sistemas para las necesidades expuestas, haciendo necesario el desarrollo de la herramienta PACSBurner.

1.5 Ambiente de desarrollo

Para el desarrollo de la plataforma XAVIA PACS-RIS se utilizaron herramientas, tecnologías, lenguajes, arquitectura y metodología que guían el desarrollo de *software* para el almacenamiento, visualización y transmisión de imágenes médicas. Las decisiones de *software* empleadas en el desarrollo de la herramienta PACSBurner, derivan del sistema general, es decir, del PACS-RIS. De esta forma se mantiene la integridad tecnológica en la solución general. En este epígrafe se describe cada una de ellas y se indica la función que realiza dentro del sistema.

1.5.1 Metodología de desarrollo del software.

Una Metodología de Desarrollo de *Software* es un marco de trabajo usado para estructurar, planificar y controlar el proceso de desarrollo en sistemas de información. En un proyecto de desarrollo de *software* la metodología ayuda a definir: Quién debe hacer, Qué, Cuándo y Cómo debe hacerlo. La metodología para el desarrollo de *software* es un modo sistemático de realizar, gestionar y administrar un proyecto para llevarlo a cabo con altas posibilidades de éxito. Estas metodologías comprenden actividades a seguir para idear, implementar y mantener un producto de *software* desde que surge la necesidad del producto hasta que se cumple el objetivo por el cual fue creado (SÁNCHEZ 2015). Dos de los principales criterios que se usan para diferenciarlas son el tamaño del personal y la criticidad del sistema.

Para el desarrollo de la investigación se adoptó la metodología AUP variante UCI como parte de la investigación, ya que es la metodología utilizada por el XAVIA PACS-RIS. La metodología de desarrollo AUP-UCI tiene como objetivo aumentar la calidad del *software* que se produce, para ello se apoya en CMMI-DEV v1.3. Este modelo constituye una guía para aplicar las mejores prácticas en una entidad desarrolladora de *software*. Estas prácticas se centran en el desarrollo de productos y servicios de calidad. En el caso de la variación de la metodología AUP definida para la actividad productiva de la UCI, esta propone para el ciclo de vida de los proyectos las fases: Inicio, Ejecución y Cierre. Posee 7 disciplinas: Modelado de negocio, Requisitos, Análisis y diseño, Implementación, Pruebas internas, Pruebas de liberación y Pruebas de aceptación. Además, esta metodología propone 4 escenarios posibles en los proyectos de desarrollo de *software*:

- Escenario 1: proyectos que modelan el negocio con casos de uso del negocio (CUN) solo pueden modelar el sistema con casos de uso del sistema (CUS).
- Escenario 2: proyectos que modelan el negocio con modelo conceptual (MC) solo pueden modelar el sistema con casos de uso del sistema (CUS).
- Escenario 3: proyectos que modelan el negocio con descripción de proceso de negocio (DPN) solo pueden modelar el sistema con descripción de requisitos de procesos (DRP).
- Escenario 4: proyectos que no modelen negocio solo pueden modelar el sistema con historias de usuario (HU).

El escenario a utilizar para el desarrollo del sistema fue el Escenario 3, de acuerdo a lo definido para el proyecto Desarrollo de XAVIA PACS 3.0.

1.5.2 Entorno de desarrollo integrado

Microsoft Visual Studio, un entorno de desarrollo integrado (IDE, por sus siglas en inglés) para *Windows*, provee un entorno de desarrollo flexible y la posibilidad de desarrollar soluciones para una gran gama de plataformas y dispositivos. Utiliza plataformas de desarrollo de *software* de *Microsoft* como *Windows API*, *Windows Forms*, *Windows Presentation Foundation*, *Windows Store* y *Microsoft Silverlight*. Admite 36 lenguajes de programación diferentes y permite que el editor de código y el depurador admitan casi gran variedad de lenguajes de programación. Los lenguajes integrados incluyen *C*, *C++*, *C++ / CLI*, *Visual Basic .NET*, *C#*, *F#*, *JavaScript*, *TypeScript*, *XML*, *XSLT*, *HTML* y *CSS*. La compatibilidad con otros idiomas como *Python*, *Ruby*, *Node.js* y *M*, entre otros, está disponible a través de complementos. *Java* (y *J#*) fueron compatibles en el pasado (TerryGLee 2010).

1.5.3 Lenguaje de programación

C#, un lenguaje de programación desarrollado por *Microsoft*, orientado a objetos, que ha sido diseñado para compilar diversas aplicaciones que se ejecutan en *.NET Framework*. Se trata de un lenguaje simple, eficaz y con seguridad de tipos. Las numerosas innovaciones de *C#* permiten desarrollar aplicaciones rápidamente y mantener la expresividad y elegancia de los lenguajes de estilo de *C* (Anders Hejlsberg, Scott Wiltamuth, Peter Golde 2012). Algunas de sus características son:

- Posee una sintaxis sencilla
- Orientado a objetos
- Orientación a componentes
- Es multiplataforma
- Integración con otros lenguajes
- Mejora en la gestión de memoria
- Multihilo

1.5.4 Lenguaje de modelado

El Lenguaje de Modelado Unificado (UML, por sus siglas en inglés) es un lenguaje que permite especificar, visualizar y construir los artefactos de los sistemas de *software*. Es el estándar más utilizado por los desarrolladores, autores y proveedores de Herramientas para Ingeniería de *Software* Asistida por Computadoras (CASE, por sus siglas en inglés).

Según Pressman el UML es “un lenguaje estándar para escribir diseños de *software*”, puede usarse para visualizar, especificar, construir y documentar los artefactos de un sistema de software intensivo (Pressman 2010). Este lenguaje está pensado para utilizarse en todos los métodos de desarrollo, etapas del ciclo de vida de un software, dominios de aplicación y medios. Es un sistema notacional destinado a los sistemas de modelado que utilizan conceptos orientados a objetos. Es el estándar mundial que utilizan los desarrolladores, autores y proveedores de CASE.

1.5.5 Herramienta CASE

Enterprise Architect 7.5, es una herramienta comprensible de diseño y análisis UML, cubriendo el desarrollo de *software* desde el paso de los requerimientos a través de las etapas del análisis, modelos de diseño, pruebas y mantenimiento. Es una herramienta multi-usuario, basada en *Windows*, diseñada para ayudar a construir *software* robusto y fácil de mantener. Ofrece salida de documentación flexible y de alta calidad (Sparx Systems 2018). Algunas de sus características son:

- Crear elementos del modelo UML para un amplio alcance de objetivos.
- Crear conectores entre elementos.
- Documentar los elementos que ha creado.
- Generar código para el *software* que está construyendo.
- Realizar ingeniería reversa del código existente en varios lenguajes.
- Importación/Exportación XMI 2.1.

1.5.6 Interfaz de programación de aplicaciones (API)

WPF, es la abreviación de *Windows Presentation Foundation*. Es una serie de ensamblados y herramientas del *framework .NET*. Está destinado a proporcionar una API (Interfaz de programación de aplicaciones) para crear interfaces de usuario enriquecidas y sofisticadas para *Windows*. Está soportado desde *Windows XP* hasta la última versión de *Windows*, la versión 10 (MacDonald 2010). Algunas de sus características principales son:

- **Interfaz gráfica declarativa:** permite crear interfaces de usuario utilizando un lenguaje de marcado llamado XAML.
- **Diseño dinámico:** aporta «inteligencia» cuando se producen cambios de tamaño de pantalla y ajustes de los componentes.

- **Gráficos basados en vectores:** los gráficos en WPF están basados en vectores, estos pueden ser escalados sin deformaciones, y ocupan menos espacio al ser almacenados.
- **Plantillas:** se puede crear elementos reutilizables para la interfaz gráfica. Existen dos tipos de plantillas en WPF: plantillas de control y plantillas de datos.
- **Binding:** utiliza el patrón de diseño MVVM (*ModelView-ViewModel*).
- **Estilos:** permite reutilizar los estilos en los diferentes controles y plantillas de la aplicación.

1.5.7 Gestor de Base de Datos

SQLite, un sistema completo de bases de datos que soporta múltiples tablas, índices, *triggers* y vistas. No necesita un proceso separado funcionando como servidor ya que lee y escribe directamente sobre archivos que se encuentran en el disco duro (Owens, Mike 2006) (Owens, Michael 2006). Algunas de sus características son:

- El formato de la base de datos es multiplataforma.
- Dependiendo de la plataforma, toda la biblioteca se maneja desde un único archivo (*shell* de comandos).
- Se utiliza el mismo lenguaje SQL, por lo que es de fácil uso.
- Las bases de datos se guardan en un archivo con extensión .bd.

1.5.8 Librería MyDICOM.NET

MyDICOM.NET, es un conjunto de librerías que implementan el estándar DICOM. Las mismas están desarrolladas sobre el lenguaje C#. Entre sus servicios más importantes están los de transmisión, visualización, impresión y mensajería DICOM.

Conclusiones del capítulo

La especificación de los conceptos asociados al objeto de estudio permitió contextualizar los principales términos abordados en el capítulo y la investigación en general. Además, el análisis de los sistemas informáticos estudiados constituyó un punto de referencia para el desarrollo de las funcionalidades de la herramienta PACSBurner, como parte del módulo PACSViewer de la plataforma para el Almacenamiento y Transmisión de Imágenes Digitales XAVIA PACS-RIS.

La arquitectura, herramientas, técnicas, métodos y lenguajes empleados en el proceso de desarrollo de *software* de XAVIA PACS-RIS son los mismos que se emplean en el desarrollo de la herramienta PACSBurner, lo cual permite mantener la integridad del sistema general.

Capítulo 2: Propuesta de solución

En el presente capítulo se describen los elementos fundamentales que integran la herramienta para el grabado de estudios imagenológicos en dispositivos externos sea CD, DVD o memoria USB. Se presentan los aspectos principales que permiten la descripción de la solución propuesta, así como de sus requisitos funcionales y no funcionales y el modelado de los casos de uso que materializarán la presente investigación.

2.1 Características de la propuesta de solución

Para dar solución al objetivo propuesto al inicio de la investigación, se decide desarrollar una herramienta para el grabado de los estudios imagenológicos en dispositivos externos sea CD, DVD o memoria USB. Esta herramienta debe estar integrada a la plataforma XAVIA PACS-RIS, para ser usada por los especialistas que laboran en el departamento de radiología. Además, debe integrar los elementos técnicos que se presentaron en el primer capítulo de esta investigación como la gestión del grabado y de la anonimización de los datos de los pacientes, también que permita adjuntar un informe y mantener los sectores del CD o DVD abiertos para el grabado de otros estudios. En las siguientes secciones se describen detalladamente los elementos que integran la propuesta de solución.

2.1.1 Descripción general de la solución

Para el desarrollo de la herramienta se divide el trabajo en cinco partes como son: grabar en CD/DVD, gestión de datos, exportar a dispositivo, gestión de la anonimización y adjuntar informe; posibilitando tener organización durante su desarrollo.

Grabar en CD/DVD:

Para lograr el grabado de los datos, se hará uso de un directorio temporal que contiene los datos seleccionados y otros archivos de interés. Se implementará un método que hará uso de la librería *BurnLib* contenedora de los métodos necesarios para el grabado en estos dispositivos.

Gestión de datos:

Para lograr la gestión de los datos se hará uso de una lista contenedora de todos los pacientes. Esta lista será obtenida mediante la elaboración de un método que permitirá obtener los datos de la base de datos del sistema XAVIA PACS, la información obtenida será mostrada en un árbol de estructura, lo que permitirá la selección de los datos.

Exportar a dispositivo:

Para exportar a dispositivo se hará uso de un directorio temporal que contiene los datos seleccionados y otros archivos de interés. Se desarrollará un método que permita copiar este directorio temporal en una ruta seleccionada por el usuario.

Gestión de la anonimización:

Para lograr la anonimización de los datos de los pacientes, se utilizará la técnica pseudoanonimización por aleatorización, que permite la asignación de seudónimos a los datos de identificación directa de los individuos. Para la gestión de los datos a anonimizar, se hará uso de la librería MyDICOM que permite obtener acceso a los datos de la imagen DICOM, estos son mostrados en la interfaz y se le permitirá al usuario su selección.

Adjuntar informe:

Se consumirá un servicio web que genera un informe emitido por el especialista, el cual será convertido a formato pdf, se adjuntará a un directorio temporal creado juntos a los demás datos a grabar, para posteriormente ser exportado o grabado a un dispositivo externo.

2.2 Modelado de los procesos del negocio

Se define el modelado de negocios como una herramienta conceptual que contiene un conjunto de objetos, conceptos y sus relaciones con el objetivo de expresar la lógica del negocio de una empresa. Proporciona una vista simplificada de la estructura de negocios que actúa como la base para la comunicación, mejoras o innovación y define los requisitos de los sistemas de información que apoyan la empresa (IAN SOMMERVILLE 2005). El presente modelado es descrito de acuerdo a la metodología seleccionada en el epígrafe 1.5 del Capítulo 1.

2.2.1 Descripción de requisitos por proceso

Los requerimientos para un sistema son la descripción de los servicios proporcionados por el sistema y sus restricciones operativas. Estos requerimientos reflejan las necesidades de los clientes de un sistema que ayude a resolver algún problema como el control de un dispositivo, hacer un pedido o encontrar información (IAN SOMMERVILLE 2005).

Requisitos Funcionales

Son declaraciones de los servicios que debe proporcionar el sistema, de la manera en que éste debe reaccionar a entradas particulares y de cómo se debe comportar en situaciones particulares. En algunos casos, los requerimientos funcionales de los sistemas también pueden declarar explícitamente lo que el sistema no debe hacer (IAN SOMMERVILLE 2005). Los requisitos funcionales designados para el desarrollo de la herramienta, son los expuestos en la Tabla 2:

Tabla 2. Requisitos Funcionales.

Número	Requisito	Descripción
RF1	Grabar CD/DVD	Permite grabar el estudio imagenológico a un CD o DVD
RF2	Exportar a dispositivo	Permite exportar el estudio imagenológico a un dispositivo externo, sea memoria USB, disco duro

		HDD o a la propia computadora
RF3	Adjuntar Informe	Permite seleccionar si se desea o no adjuntar el informe al estudio
RF4	Seleccionar imágenes a grabar/exportar	Permite seleccionar las imágenes que se deseen grabar o exportar
RF5	Seleccionar los datos a anonimizar del paciente	Permite seleccionar los datos de los pacientes que se deseen anonimizar

Las tablas 3, 4 y 5, contienen la descripción de los requisitos funcionales RF3, RF4 y RF5, según el formato establecido en el expediente de proyecto Desarrollo de XAVIA PACS-RIS para la descripción de requisitos por procesos. Los requisitos RF1 (Ver Anexo 9) y RF2 (Ver Anexo 10) son los especificados en el Expediente de proyecto Desarrollo de XAVIA PACS-RIS(Proyecto Desarrollo de la Plataforma para la gestión de la información imagenológica (PACS-RIS) 2017a).

Tabla 3. Descripción del requisito "Adjuntar Informe".

Descripción textual	Permite la opción de si se desea adjuntar un informe o no al grabado.
Actores	Especialista.
Precondición	
Flujo de eventos	
Flujo básico "Adjuntar Informe"	
	1. El Especialista selecciona "Adjuntar Informe" de la ventana "Grabar/Exportar estudio".
Pos-condiciones	
	1. Se graba el disco con el informe adjunto.
Flujos alternos	
Flujo alternativo "No existen informes a adjuntar"	
	1. El Sistema muestra un mensaje de error informando que no existen informes a adjuntar.

2. El **Especialista** selecciona la opción “Aceptar” o la “X” ubicada en la esquina superior derecha.
3. El **Sistema** muestra la ventana “Grabar/Exportar estudio” con la opción “Adjuntar Informe” deshabilitada.
4. Fin del requisito.

Tabla 4. Descripción del requisito "Seleccionar imágenes a grabar/exportar ".

Descripción textual	Permite la selección de el/los paciente(s), estudio(s) o serie(s) que se desean grabar/exportar.
Actores	Especialista.
Precondición	Que exista al menos un paciente, estudio o serie en la Bandeja de casos.
Flujo de eventos	
Flujo básico “Seleccionar imágenes a grabar/exportar”	
1. El Especialista selecciona el/los paciente(s), estudio(s) o serie(s), de la ventana “Grabar/Exportar estudio”.	
Pos-condiciones	
1. Se graba el disco con el/los paciente(s), estudio(s) o serie(s) seleccionados.	
Flujos alternos	
N/A	

Tabla 5. Descripción del requisito "Seleccionar los datos a anonimizar del paciente ".

Descripción textual	Permite la selección de los datos de los pacientes que se desean anonimizar.
Actores	Especialista.
Precondición	Que exista al menos un paciente, estudio o serie en la Bandeja de casos.

Flujo de eventos
Flujo básico “Seleccionar los datos a anonimizar del paciente”
2. El Especialista selecciona los datos a anonimizar de la ventana “Grabar/Exportar estudio”.
Pos-condiciones
2. Se graba el disco con el/los paciente(s), estudio(s) o serie(s) con los datos seleccionados anonimizados.
Flujos alternos
N/A

Requisitos No Funcionales

Los requerimientos no funcionales, son aquellos requerimientos que no se refieren directamente a las funciones específicas que proporciona el sistema, sino a las propiedades emergentes de éste como la fiabilidad, el tiempo de respuesta y la capacidad de almacenamiento. De forma alternativa, definen las restricciones del sistema como la capacidad de los dispositivos de entrada/salida y las representaciones de datos que se utilizan en las interfaces del sistema (IAN SOMMERVILLE 2005). Los requisitos no funcionales están en concordancia con los requisitos definidos para el proyecto Desarrollo de XAVIA PACS-RIS en la documentación “Especificación de requisitos de software(Proyecto Desarrollo de la Plataforma para la gestión de la información imagenológica (PACS-RIS) 2017b)” del producto. En la Figura 2 se definen los atributos de calidad a tener en cuenta para la descripción de requisitos no funcionales y las tablas 6, 7, 8, 9, 10 contienen la especificación de los requisitos no funcionales.

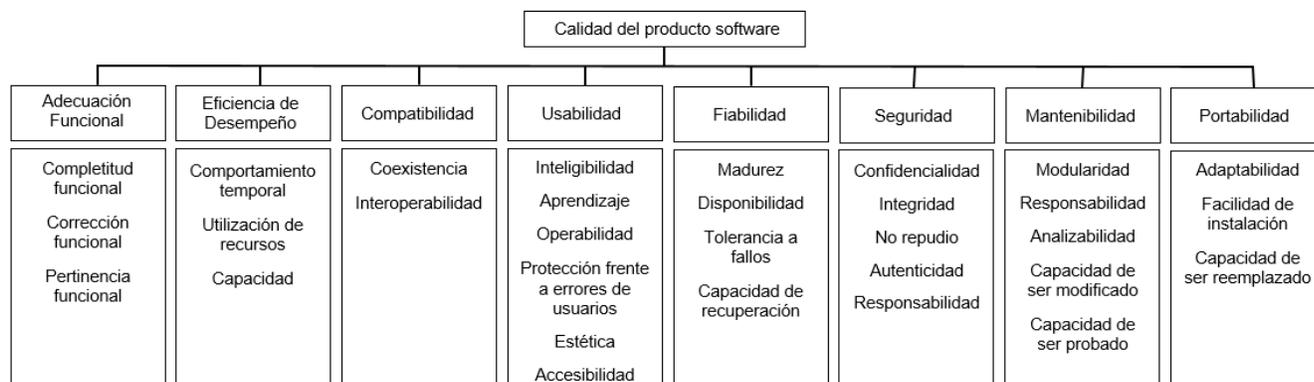


Figura 2. Calidad del producto de software.

Tabla 6. Descripción de requisito no funcional. Atributo de Calidad: Seguridad.

Atributo de Calidad	Seguridad
Sub-atributos/Sub-características	Confidencialidad, Integridad
Objetivo	Mantener seguridad y control a nivel de usuario (especialista, transcriptor), garantizando el acceso de los mismos sólo a los niveles establecidos (en el XAVIA PACS Reporter) de acuerdo a la función que realizan.
Origen	Usuario
Artefacto	Todo el sistema
Entorno	El sistema funcionando correctamente.
Estímulo	Respuesta: Flujo de eventos (Escenarios)
1.a Acceso al sistema XAVIA PACSReporter	
Acceso de un usuario al sistema XAVIA PACSReporter.	El sistema permite el acceso del usuario al sistema, mostrando las entidades a las cuales tiene permiso. Dentro de las funcionalidades permitidas, se limitará a los permisos que tenga asignado el usuario en cuanto a departamentos, servicios o especialidades.
Medida de respuesta	
El sistema notifica cuando no se tienen permisos para el recurso solicitado.	

Tabla 7. Descripción de requisito no funcional. Atributo de Calidad: Usabilidad.

Atributo de Calidad	Usabilidad
Sub-atributos/Sub-características	Inteligibilidad
Objetivo	Identificar las funcionalidades por un nombre, ícono y figura ilustrativa que brinde información al usuario.
Origen	Sistema
Artefacto	Sistema
Entorno	El sistema funcionando correctamente.
Estímulo	Respuesta: Flujo de eventos (Escenarios)
1. a Acceso a una de las funcionalidades del sistema XAVIA PACSViewer	
El usuario accede a una de las funcionalidades del sistema.	Al acceder a una de las funcionalidades del sistema XAVIA PACSViewer, se muestra una breve descripción de esta (al pasar el mouse por encima del ícono de dicha funcionalidad).
Medida de respuesta	
Se visualiza una breve descripción de la funcionalidad. Cada funcionalidad se encuentra identificada por un ícono representativo y su nombre ilustra su objetivo en el sistema.	

Atributo de Calidad	Usabilidad
Sub-atributos/Sub-características	Operabilidad
Objetivo	Facilitar el uso y navegación en el sistema por parte del usuario, mediante el uso de pautas de diseño que estandaricen las funcionalidades y flujos de información en el sistema.
Origen	Sistema

Artefacto	Interfaz de usuario
Entorno	El sistema funcionando correctamente.
Estímulo	Respuesta: Flujo de eventos (Escenarios)
1. a Interacción con diferentes funcionalidades del sistema	
Los usuarios interactúan con diferentes funcionalidades del sistema XAVIA PACSViewer.	El sistema permite una fácil navegación por las funcionalidades con diseño o comportamientos similares.
Medida de respuesta	
Las funcionalidades del sistema presentan un diseño o comportamiento estándar.	

Atributo de Calidad	Usabilidad
Sub-atributos/Sub-características	Protección frente a errores de usuarios
Objetivo	Notificar en el sistema la existencia de datos incorrectos o incompletos, así como confirmaciones de acciones que así lo requieran.
Origen	Sistema
Artefacto	Sistema
Entorno	El sistema funcionando correctamente.
Estímulo	Respuesta: Flujo de eventos (Escenarios)
1. a Acceso a una de las funcionalidades del sistema XAVIA PACSViewer	
El usuario accede a una de las funcionalidades del sistema y realiza una acción.	Una vez completada la acción el sistema notifica la existencia de datos incorrectos o incompletos y/o realiza las confirmaciones sobre las acciones que así lo requieran.
Medida de respuesta	
El sistema notifica al usuario la existencia de datos incorrectos o incompletos cuando accede a una funcionalidad. Para las funcionalidades que así lo requieran, el sistema muestra mensajes de confirmación.	

Tabla 8. Descripción de requisito no funcional. Atributo de Calidad: Eficiencia de desempeño.

Atributo de Calidad	Eficiencia de desempeño
Sub-atributos/Sub-características	Utilización de recursos
Objetivo	Ejecutar el sistema XAVIA PACSViewer con los requerimientos necesarios.
Origen	Cliente.
Artefacto	El sistema.
Entorno	El sistema no se encuentra en ejecución.
Estímulo	Respuesta: Flujo de eventos (Escenarios)
1.a Ejecutar el sistema con los requerimientos necesarios para su funcionamiento	

<p>Se ejecuta el sistema XAVIA PACSViewer con los requerimientos necesarios para su funcionamiento.</p> <p>Características de Software:</p> <ul style="list-style-type: none"> • Computadora con Sistema Operativo (SO) Windows XP sp3 y Marco de trabajo .NET Framework 4.0. <p>Características del Hardware:</p> <ul style="list-style-type: none"> • <i>Para la consulta y visualización de imágenes. Así como, quemado de estudios imagenológicos, transcripción y/o impresión de reportes diagnósticos:</i> HDD: 80 GB (la capacidad de almacenamiento de las estaciones de trabajo tiene relación directa a la cantidad de estudios que permitirá almacenar de manera local y temporal para su visualización.), TR: 100 Mbps, RAM: 1 GB, CPU: Pentium 4 3.0 Ghz. • <i>Para el diagnóstico mediante el empleo de la reconstrucción multiplanar y por referencia:</i> HDD: 80 GB, TR: 100 Mbps, RAM: 2 GB, CPU: Pentium 4 3.0 Ghz. • <i>Para diagnóstico mediante el empleo de la reconstrucción 3D:</i> HDD: 80 GB, TR: 100 Mbps, RAM: 4 GB, CPU: Dual Core 2.5 Ghz, TV: 512 MB. <p>Leyenda:</p> <p>TV: Tarjeta de video.</p> <p>TR: Tarjeta de red.</p> <p>RAM: Memoria de acceso aleatorio.</p>	<p>El sistema funciona correctamente.</p>
--	---

HDD: Disco duro.	
CPU: Procesador.	
SO: Sistema operativo.	
FW: Marco de trabajo (Framework).	
Medida de respuesta	
El sistema se encuentra disponible y funciona correctamente. En caso que no se cumplan estos requerimientos el sistema no funcionara correctamente.	

Tabla 9. Descripción de requisito no funcional. Atributo de Calidad: Mantenibilidad.

Atributo de Calidad	Mantenibilidad
Sub-atributos/Sub-características	Capacidad de ser probado
Objetivo	Establecer criterios de prueba para verificar la calidad del sistema.
Origen	Probador
Artefacto	El sistema XAVIA PACSViewer.
Entorno	El sistema XAVIA PACSViewer se encuentra funcionando correctamente.
Estímulo	Respuesta: Flujo de eventos (Escenarios)
1. a Ejecución de los casos de prueba	
El probador ejecuta las pruebas a partir del diseño de los casos de prueba definidos para cada requisito.	La respuesta del sistema se encuentra en correspondencia con el diseño de caso de prueba definido para cada requisito.
Medida de respuesta	
Existe un diseño de caso de prueba por cada requisito.	

Atributo de Calidad	Mantenibilidad
Sub-atributos/Sub-características	Reusabilidad
Objetivo	Permitir la reutilización de elementos comunes en el desarrollo del sistema.
Origen	Desarrolladores
Artefacto	El sistema
Entorno	El sistema se encuentra en desarrollo.
Estímulo	Respuesta: Flujo de eventos (Escenarios)
1. a Reutilización	
El desarrollador reutiliza componentes y plantillas.	El sistema funciona correctamente.
Medida de respuesta	
El sistema funciona correctamente y los componentes y/o plantillas reutilizadas se encuentran acorde a las pautas de diseño definidas.	

Tabla 10. Descripción de requisito no funcional. Atributo de Calidad: Portabilidad.

Atributo de Calidad	Portabilidad
Sub-atributos/Sub-características	Capacidad de ser reemplazado (reemplazabilidad)

Objetivo	Permitir la reemplazabilidad del sistema por una nueva versión del propio sistema.
Origen	Usuario
Artefacto	Todo el sistema.
Entorno	El sistema funcionando correctamente.
Estímulo	Respuesta: Flujo de eventos (Escenarios)
1.a Actualizar versión del sistema XAVIA PACSViewer	
El usuario actualiza la versión del sistema XAVIA PACSViewer.	El sistema funciona correctamente.
Medida de respuesta	
Sistema funcionando correctamente con las nuevas actualizaciones.	

2.3 Análisis y diseño

El modelo de diseño constituye el conjunto de diagramas que describen el diseño lógico de un sistema. Comprende los diagramas de clases de *software*, diagramas de interacción, diagramas de paquetes, etc., ofreciendo una perspectiva de especificación o implementación, como quiere el modelador.

2.3.1 Arquitectura de *software*

Modelo–vista–vista modelo (MVVM, por sus siglas en inglés) es un patrón de arquitectura de *software*, que tiene como idea fundamental el organizar la estructura de un proyecto de *software*. Este patrón tiene como propósito separar el código de la interfaz de usuario (UI, por sus siglas en inglés) y el código que no es de la UI. El código de la UI utiliza lenguajes declarativos como el XAML que permiten interactuar con las demás capas del aplicativo. Está contenido en tres capas:

Model. Esta es la capa Modelo y es 100% independiente de las otras dos capas. Aquí se encuentra lo que se conoce como la capa de lógica de negocio y la capa de acceso a datos. La idea es manejar las reglas de negocio para acceder a los datos y cómo se va a manipular.

View. La capa de Vista tiene la UI. La interfaz de usuario está creada por un lenguaje declarativo como XAML donde se va a crear el enlace entre los componentes de la UI y el código. Permite mostrar en pantalla transformando el modelo en UI.

ViewModel. La capa VistaModelo tiene como finalidad que el enlace con el Model directamente o de forma encapsulada. Entonces, expone métodos, componentes, comandos y demás medios para interactuar con el modelo después de tener una acción en la vista (Gaudioso 2010). La Figura 2 representa un esquema del patrón arquitectónico MVVM.

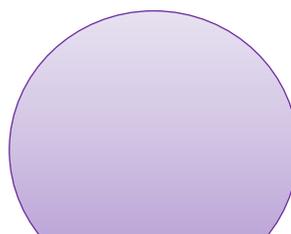




Figura 3. Funcionamiento de la arquitectura MVVM.

2.3.2 Patrones de diseño

Los patrones de diseño son técnicas para resolver 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, además debe ser reutilizable, lo que significa que es aplicable a diferentes problemas de diseño en distintas circunstancias. A continuación se explican los Patrones de Asignación de Responsabilidades (GRASP, por sus siglas en inglés), utilizados en la solución (Rising 1998).

Patrones GRASP

Los GRASP presentan una serie de buenas prácticas para el diseño de *software*. El objetivo de estos patrones es describir los principios fundamentales del diseño de objetos y la asignación de responsabilidades. Entre los principales patrones GRASP se encuentran: Experto, Creador, Alta cohesión, Bajo acoplamiento y Controlador. Cada uno describe un problema, una solución y los beneficios de implementarlos (Rising 1998). A continuación, se explican los patrones empleados en la solución:

Experto: clase que tiene la responsabilidad de ejecutar una acción determinada y cuenta con el acceso

a los datos necesarios. Este patrón se evidencia en la clase *ConnectionManager*. En la Figura 3 se puede observar un fragmento de código de la clase antes mencionada.

```

namespace ClientDataTools
{
    6 references
    public class ConnectionManager
    {
        #region Fields
        private SQLiteConnection _connection;
        #endregion

        #region Methods
        19 references
        public SQLiteConnection GetConnection()
        {
            _connection = new SQLiteConnection("Data Source=" + Path.Combine(Directories.DirDB, "DicomExpress.s3db") +
                ";New=False;Compress=False;Synchronous=Off;UTF8Encoding=False;Version=3");
            return _connection;
        }
    }
}

```

Figura 4. Fragmento de código de la clase *ConnectionManager*.

Creador: las clases que tienen la responsabilidad de instanciar los objetos o componentes. Este patrón se evidencia en la clase *MainViewModels*. En la Figura 4 se puede observar un fragmento de código de la clase antes mencionada.

```

namespace BurnerPro.ViewModels
{
    2 references
    public class MainViewModels:INotifyPropertyChanged
    {
        Campos
        #region Properties
        1 reference
        public int CantImageCopy
        {
            get
            {
                for (int i = 0; i < _PacientesCargados.Count; i++)
                {
                    _imagenes += Pacientes[i].CantImage;
                }
                return _imagenes;
            }
            set { _imagenes = value; NotifyPropertyChanged("ImagenesToCopy"); }
        }
        1 reference
        public List<string> ListaNombres...
        5 references
        public List<ECPatient> Pacientes...
        1 reference
        public string DirTmpBurn...
        #endregion
    }
}

```

Figura 5. Fragmento de código de la clase *MainViewModels*.

Controlador: patrón evidenciado en las clases que se encargan de ser intermediarias entre una determinada interfaz y el algoritmo que la implementa, siendo estas las que reciben la información entrada por el usuario y la que los envía a las distintas clases según el método llamado, como es la clase *MainViewModels*. En la Figura 4 se puede observar un fragmento de código de la clase antes mencionada.

Alta Cohesión: cuando cada clase realiza pocas funciones. Se puede observar el uso de este patrón en la clase *HeaderToImageConverter*. En la Figura 5 se puede observar un fragmento de código de la clase antes mencionada.

```
namespace BurnerPro
{
    [ValueConversion(typeof(string), typeof(BitmapImage))]
    2 references
    class HeaderToImageConverter : IValueConverter
    {
        public static HeaderToImageConverter Intance = new HeaderToImageConverter();
        0 references
        public object Convert(object value, Type targetType, object parameter, CultureInfo culture)
        { var nombre = (string)value;

            if (nombre == null)
                return null;
            var image = "Images/Pictures.png";

            if (MainWindow.SaberPaciente(nombre))
            {
                image = "Images/client.png";
            }
            else if (MainWindow.Saberestudio(nombre))
                image = "Images/openFile.png";
            else if (MainWindow.Saberserie(nombre))
                image = "Images/openFolder.png";
            else if (MainWindow.SaberImagen(nombre))

                image = "Images/Pictures.png";

            return new BitmapImage(new Uri($"pack://application:,,/{image}"));
        }
    }
}
```

Figura 6. Fragmento de código de la clase *HeaderToImageConverter*.

Bajo Acoplamiento: cuando una clase no depende de muchas clases. Este patrón se evidencia en la clase *HeaderToImageConverter*. En la Figura 5 se puede observar un fragmento de código de la clase antes mencionada.

2.3.3 Diagrama de paquetes

Un modelo de paquetes o diagrama de paquetes en UML, representa la dependencia entre los paquetes que componen un modelo, muestra como un sistema está dividido en agrupaciones lógicas y las dependencias entre ellas. Estos diagramas suministran una descomposición de la jerarquía lógica de un sistema. Entre los paquetes pueden existir relaciones de dependencia, esta denota que los elementos de un paquete dependen de los elementos de otro y relaciones de generalización que suele utilizarse para especificar familias de paquetes, es decir los paquetes hijos heredan los elementos de los paquetes padre (IAN SOMMERVILLE 2005). En la Figura 6 se puede observar la

representación del modelo de paquetes del sistema donde cada uno de estos paquetes está compuesto por diversos subpaquetes que a su vez contienen los diagramas de clases del diseño. Entre los paquetes existe un repositorio de clases que contiene a su vez 3 subpaquetes donde se agrupan las vistas, el modelo de negocio y el acceso a datos. El subpaquete del modelo de negocio contiene los componentes de funcionalidad. El subpaquete de acceso a datos está conformado por el componente de persistencia y la base de datos. Por último, el paquete de las vistas contiene las vistas que interactúan con el usuario, a través de la cual se reciben y se muestran los datos.

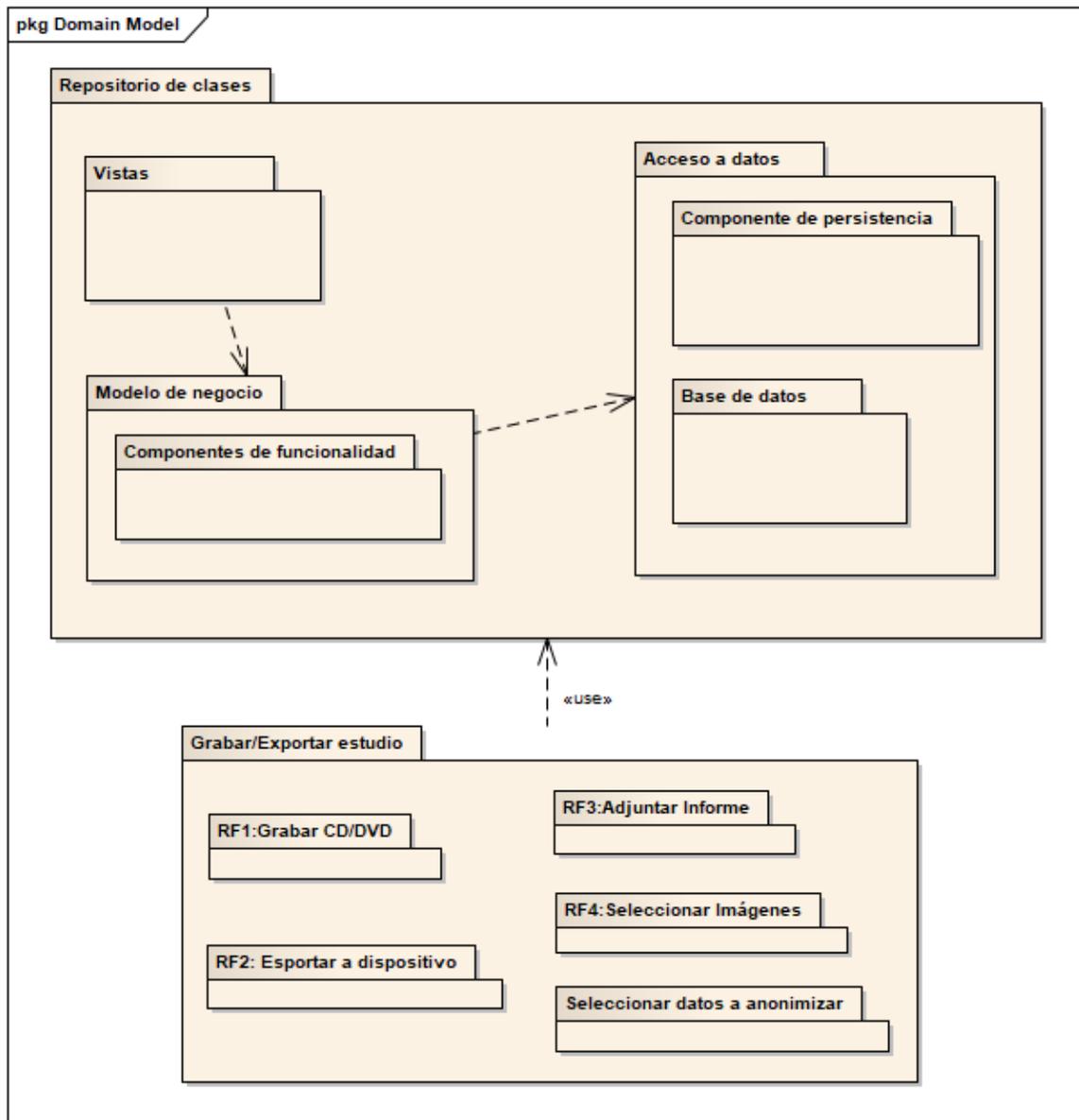


Figura 7. Diagrama de paquetes.

2.3.4 Diagrama de clases de diseño

Un diagrama de clases representa las relaciones entre clases que involucran el sistema, estas relaciones pueden ser asociativas, de herencia, de uso y de agregación; las clases están compuestas

por atributos y métodos(Pressman 2010). La Figura 7 representa el diagrama de clases, exportado desde el entorno de desarrollo integrado *Visual Studio*, a partir del código del sistema.

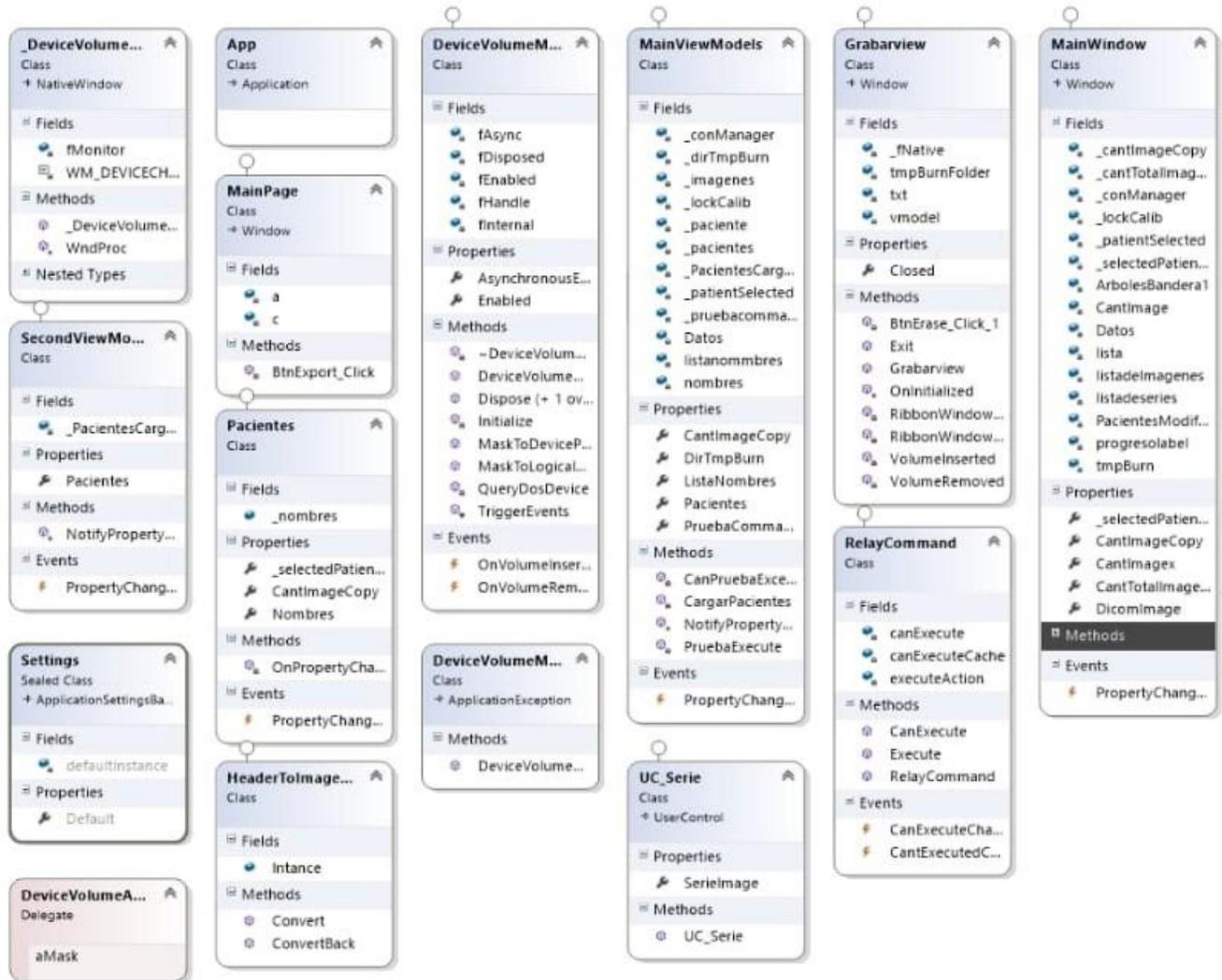


Figura8. Diagrama de clases de diseño Grabar/Exportar.

El diagrama de clases de diseño está compuesto por 15 clases, de las cuales a partir de la arquitectura utilizada MVVM, las clases *MainViewModels* y *SecondViewModels* pertenecen a la Vista-Modelo, encargadas de la comunicación entre los datos y la interfaz de la aplicación. Las clases *MainPage*, *Grabarview* y *MainWindow* son las encargadas de la parte visual de la aplicación, es decir la interfaz de usuario, denominadas a partir de la arquitectura usada Vistas. La clase *RelayCommand* es quien contiene la estructura de los comandos usados en la vista para ejecutar una operación. Las clases *App* y *Settings* son autogeneradas por el entorno de desarrollo y contienen elementos de configuración de la aplicación. La clase *UC_Serie* permite el control visual de una serie en la interfaz de la herramienta. La clase *HeaderToImageConverter* es la encargada de convertir elementos visuales de la interfaz del *treeview* mostrar una imagen para el elemento deseado (pacientes,

estudios, series) y la clase *DeviceVolumeMonitor* se encarga de reconocer los dispositivos extraíbles que han sido conectados.

Conclusiones del capítulo

La documentación que se obtuvo, de la metodología AUP UCI, permitió un mejor entendimiento de la gestión de información a realizar, a partir de la cual se identificaron los requisitos funcionales y se especificaron los requisitos no funcionales, todo ello necesario para el correcto desarrollo de la herramienta propuesta. Además, se identificaron las clases fundamentales, a partir del modelado del negocio, que deben ser definidas para que la herramienta gestione la información correctamente. El uso de los patrones de diseño seleccionados fortalece la construcción del código proporcionándole legibilidad, independencia y mantenibilidad. Se realizó la descripción de requisitos por procesos, posibilitando recoger todos los detalles para un correcto desarrollo de la herramienta PACSBurner.

Capítulo 3: Implementación y pruebas

En el presente capítulo se describen los estándares de codificación utilizados para obtener un código limpio y legible. Así como se hace referencia a las tareas ingenieriles que rigen el desarrollo de la herramienta y las pruebas realizadas a la misma. Con los resultados obtenidos se demuestra la eficiencia y el correcto funcionamiento de la solución que se desarrolló para dar solución al problema planteado.

3.1 Modelo de datos.

Un modelo de datos es un conjunto de conceptos utilizados para organizar los datos de interés y describir su estructura en forma comprensible para un sistema informático. Constituye una definición lógica y abstracta de los objetos y operadores que en conjunto constituyen la máquina abstracta con la que interactúan los usuarios. Este modelo proporciona una representación visual y física de los datos persistentes del sistema, que en el futuro serán la base de datos. Los objetos permiten modelar la estructura de los datos y los operadores permiten modelar su comportamiento. Se obtiene a partir del diagrama de clases persistentes y su forma se expresa mediante un diagrama de UML, siendo sus elementos esenciales las entidades, atributos y relaciones entre entidades. La Figura 8 representa el modelo de datos que se realiza a partir de los diagramas de clases del diseño. El diagrama consta de 4 entidades relacionadas, con los atributos correspondientes a cada una y la cardinalidad o multiplicidad existente.

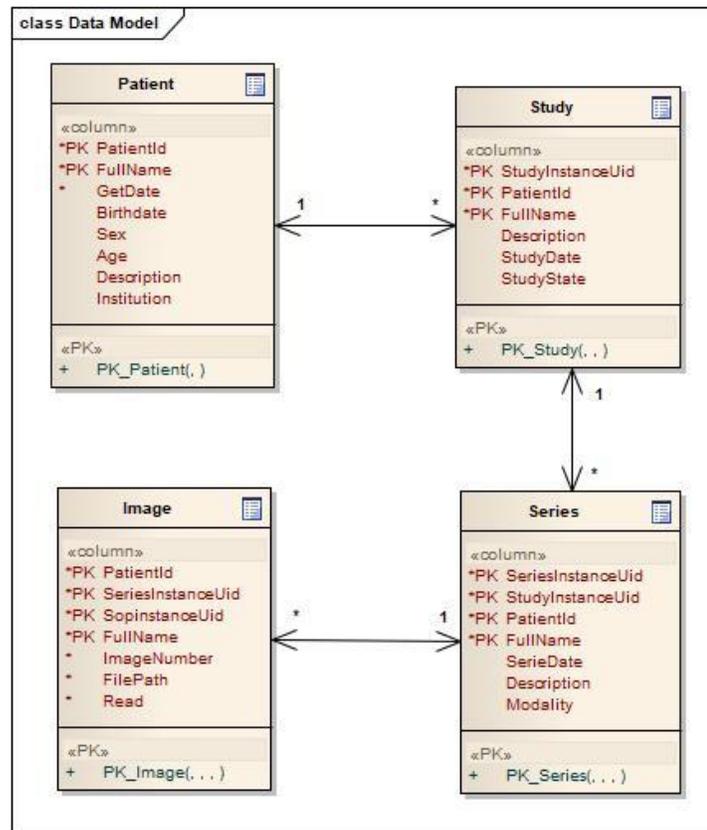


Figura 9. Modelo de datos.

3.2 Estrategias de reutilización

La práctica de reutilización de código fuente trae ventajas para toda aplicación que se encuentre en desarrollo y permite principalmente reducir tiempo, minimizar las redundancias y aprovechar el trabajo anterior. La forma más eficiente de la reutilización de código es la creación de componentes reutilizables para evitar la duplicidad del mismo.

Para el desarrollo de la herramienta PACSBurner se utilizaron varios componentes del módulo XAVIA PACSViewer creados por el equipo de desarrollo de la plataforma XAVIA PACS-RIS, con el objetivo de lograr uniformidad en el desarrollo y mejorar la calidad del trabajo. Entre los componentes principales que se reutilizan está: la librería *BurnLib*, encargada del grabado de la información deseada en un dispositivo externo sea CD, DVD o un dispositivo USB y el modelo de datos *ClientEntities* que contiene los tipos de datos usados en el desarrollo de la herramienta, entre ellos se encuentra *ECPatient*, que contiene todos los datos del paciente, *ECStudy*, *ECSeries* y *ECImage*.

3.3 Tratamiento de errores

Durante el tiempo de ejecución de un sistema pueden fracasar diferentes rutinas; es esto a lo que comúnmente se le llama excepción. Mediante el tratamiento de excepciones se restaura a un estado en el que la rutina pueda seguir la ejecución, lo que permite obtener un sistema robusto y fiable.

En la herramienta desarrollada, el control de las excepciones se lleva a cabo a toda porción de código donde pueda surgir alguna situación inesperada, especialmente donde se ejecutan sentencias que manipulan los datos que viajan desde y hacia la base de datos.

Para el manejo de las excepciones o errores, en las clases controladoras de procesos, se utiliza el bloque *try* para detectar cuándo ocurra algún fallo y un bloque *catch* donde se manejarán dichas excepciones, mediante mensajes que se muestran en la interfaz de usuario. En la Figura 9 se puede observar un fragmento de código que evidencia el tratamiento de errores.

```
async Task CargarAsync2()
{
    try
    {
        bool res = await Task.Run(() => Exportar());
    }
    catch (Exception ex)
    {
        MessageBox.Show("Error leyendo los datos desde archivo: " + ex.ToString());
    }
}
```

Figura 10. Representación del tratamiento de errores.

3.4 Estándares de codificación

Según Guido van Rossum (científico de la computación, conocido por ser el autor del lenguaje de programación *Python*), el código es leído muchas más veces de lo que es escrito. Los estándares de programación son reglas que se aplican para lograr uniformidad en el código producido por un grupo de desarrollo de un sistema. Estos reducen perceptiblemente el riesgo de que los desarrolladores introduzcan errores. Los estándares de codificación no destapan problemas existentes, evitan más bien que los errores ocurran, lo que permite obtener un código de alta calidad. Para el desarrollo de este proyecto se utilizó como base los estándares de codificación utilizados por C#, mediante el uso de reglas para codificación en C# (Diez 2003). Entre las pautas más destacadas de la guía citada, se encuentran:

- Mantener las clases y ficheros cortos, con no más de 2 000 líneas de código y que estén claramente divididas sus estructuras.
- Los comentarios de línea se utilizan para explicar línea a línea el código fuente. También se utilizan para comentar líneas de código temporalmente.
- Iniciar variables locales lo antes posible; si es posible, durante la declaración.
- Cada línea debe contener sólo una sentencia.
- Las líneas en blanco mejoran la legibilidad del código. Separan los bloques de código que están relacionados lógicamente.

3.5 Integración al XAVIA PACSViewer

Para lograr la correcta integración de la herramienta desarrollada PACSBurner al sistema XAVIA PACS 3.0 se debe habilitar un nexo entre el componente PACSViewer y la herramienta, el cual será un botón dentro de la interfaz de la bandeja de casos. En la Figura 10 se puede observar el botón creado en la bandeja de casos del PACSViewer.

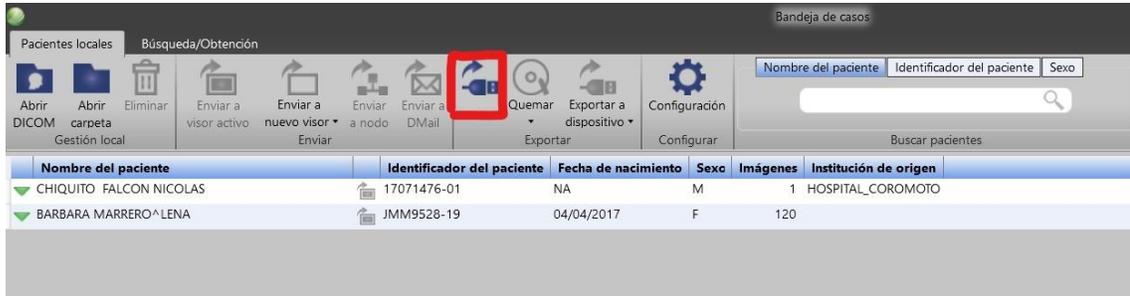


Figura11. Vista del botón Exportar de la bandeja de casos del PACSViewer.

Dentro del código de la herramienta PACSBurner, se debe modificar la vista principal *MainPage* y sustituirla por la bandeja de casos del XAVIA PACSViewer, por medio de un botón encargado de llamar a la vista *MainWindow* del PACSBurner, tomando como parámetros una lista de pacientes seleccionados por el usuario (*_selectedPatientList*), la cantidad de imágenes a copiar (*_CantImageCopy*) y un directorio temporal (*_dirTmpBurn*) que será empleado para el almacenamiento de los archivos a exportar y generalmente se encuentra en el disco donde está instalado el sistema.

El código de la bandeja de casos debe impedir el llamado de la herramienta PACSBurner sin previamente haber seleccionado pacientes. Esto se realiza con una simple consulta al parámetro (*_selectedPatientList*) que debe contener elementos. Finalmente se debe agregar la *propertyCantImageCopy* (Ver Anexo 1) que será utilizada como parámetro para el llamado a la vista *MainWindow* (Ver Anexo 2) del PACSBurner.

3.6 Pruebas de software

Las pruebas de *software* son las investigaciones empíricas y técnicas cuyo objetivo es proporcionar información objetiva e independiente sobre la calidad del producto. Cuando se realiza el proceso de pruebas el programa es ejecutado con datos de ejemplo cumpliendo con los tipos de datos de cada campo. Es necesario verificar los resultados de la prueba que se ejecuta para buscar errores, anomalías o información de atributos no funcionales del programa. El proceso de prueba tiene dos metas distintas: Demostrar al desarrollador y al cliente que el *software* cumple con los requerimientos y encontrar situaciones donde el comportamiento del *software* sea incorrecto, indeseable o no esté de acuerdo con su especificación (Pressman 2010).

3.6.1 Método de prueba de caja blanca.

La prueba de caja blanca, es una filosofía de diseño de casos de prueba que usa la estructura de control descrita como parte del diseño a nivel de componentes para derivar casos de prueba. Al usar los métodos de prueba de caja blanca, puede derivar casos de prueba que garanticen que todas las rutas independientes dentro de un módulo se revisaron al menos una vez, revisen todas las decisiones lógicas en sus lados verdadero y falso, ejecuten todos los bucles en sus fronteras y dentro de sus fronteras operativas y revisen estructuras de datos internas para garantizar su validez (Pressman 2010).

La técnica de prueba de caja blanca utilizada en la investigación es el camino básico, que permite obtener una medida de la complejidad lógica de un diseño procedimental y usar esa medida como guía para la definición de un conjunto básico de caminos de ejecución.

Complejidad ciclomática

La complejidad ciclomática se realiza a todos los métodos o algoritmos de un sistema informático. En la Tabla 6, se presenta la prueba realizada al método PrepararDatosQuemador(), que posibilita crear una carpeta temporal con los datos seleccionados por el usuario para su posterior grabado en un dispositivo extraíble.

Tabla 11. Prueba de unidad utilizando complejidad ciclomática al método PrepararDatosQuemador().

Fórmula 1	Fórmula 2	Fórmula 3	Grafo resultante
$V(G) = (A - N) + 2$ $V(G) = (7-7) + 2$ $V(G) = 2$	$V(G) = P + 1$ $V(G) = 1 + 1 = 2$	$V(G) = R$ $V(G) = 2$	<pre> graph TD 1((1)) --> 2((2)) 2 --> 3((3)) 3 --> 5((5)) 3 --> 4((4)) 4 --> 6((6)) 6 --> 7((7)) </pre>
<p>A: es la cantidad de aristas.</p> <p>N: la cantidad de nodos.</p> <p>P: es el número de nodos predicado contenidos en el grafo de flujo G.</p> <p>Nodo predicado: cada nodo que contiene una condición.</p> <p>R: representa la cantidad de regiones en el grafo.</p>			
<p>Complejidad ciclomática</p> <p>Como se puede observar, después de aplicadas las fórmulas 1, 2 y 3 a la funcionalidad PrepararDatosQuemador(), posee</p>			

una complejidad ciclomática igual a 2, lo cual demuestra que las tres fórmulas son efectivas.

Prueba de camino básico

Una vez calculada la complejidad ciclomática se define como límite superior 2, lo que indica que hay que realizarle al código dos pruebas, para garantizar que este se ejecute completamente al menos una vez. La funcionalidad PrepararDatosQuemador() posee poco riesgo debido a que el resultado arrojado por la métrica está en el intervalo entre 1 - 9.

El total de caminos independientes establecidos fue de 2 y a continuación se muestran:

- Camino 1: 1, 2, 3, 4, 5, 6, 7.
- Camino 2: 1, 2, 3, 4, 6, 7.

Al comprobar los caminos posibles, se verificó que los diseños de casos de pruebas elaborados, cubrían los 2 caminos posibles, de manera tal, que siempre fuera posible comprobar el correcto funcionamiento de la aplicación para este método.

```
1reference
public Task PrepararDatosQuemador()
{
    return Task.Run(() => 1
    {
        string temporal = tmpBurn;
        var lite = new DirectoryInfo(System.Windows.Forms.Application.StartupPath + "\\Lite"); 2
        if (Directories.ExistsDirectory(temporal)) 3
        {
            Directory.Delete(temporal, true); 4
        }
        CopyDirectory(lite, new DirectoryInfo(tmpBurn)); 5
        CopyPatients(PacientesModificados, tmpBurn, CantImage, false); 6
        DicomDir.DicomDirectory.CreateDicomDir_MyDicom(tmpBurn + "\\Images"); 7
    }
    );
}
```

3.6.2 Métodos de pruebas de caja negra

El método de pruebas de caja negra, también llamado pruebas de comportamiento, se enfocan en los requerimientos funcionales del *software*; es decir, las técnicas de prueba de caja negra le permiten derivar conjuntos de condiciones de entrada que revisarán por completo todos los requerimientos funcionales para un programa (Pressman 2010). En la presente investigación se determinó seleccionar un estudio radiológico de prueba, este fue obtenido por el Proyecto de Desarrollo de la

plataforma XAVIA PACS-RIS, el cual será usado para observar los resultados arrojados por la herramienta propuesta en la presente investigación.

Composición de un estudio radiológico

En la Figura 11 se puede observar la composición del estudio radiológico seleccionado, el cual está compuesto por series, cada una de ellas puede tener dentro una o varias imágenes DICOM relacionadas. También este tipo de estudios tienen consigo información adjunta, ya sean datos personales del paciente (sexo, edad, nombre, etc.), información sobre la institución hospitalaria donde se realizó el estudio, sobre el especialista que realizó el estudio y otros datos relacionados.

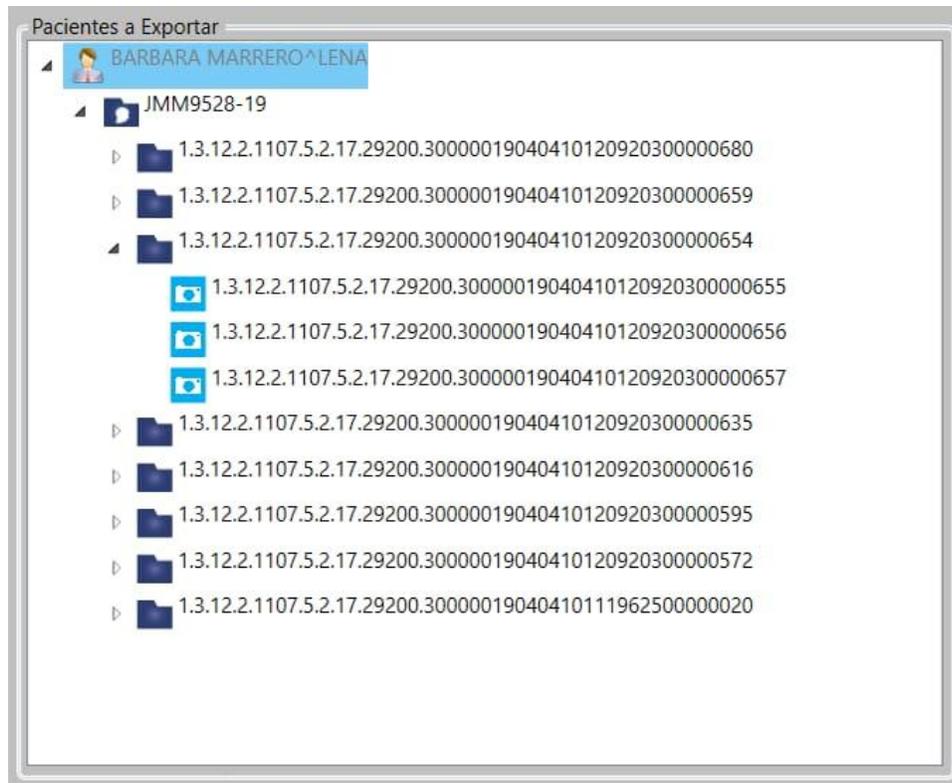


Figura12. Vista del estudio radiológico en la herramienta PACSBurner.

Comprobación de funcionalidades

Para comprobar el correcto funcionamiento de la herramienta, se probarán cada una de sus funcionalidades, haciendo uso del estudio escogido anteriormente.

- **Eliminar datos que no se deseen grabar (Gestión de la información a grabar):**

En la Figura 12 se puede observar el resultado después de haber eliminado las series e imágenes que no se deseaban grabar. Se han eliminado 3 series en su totalidad y 2 imágenes pertenecientes a diferentes series.

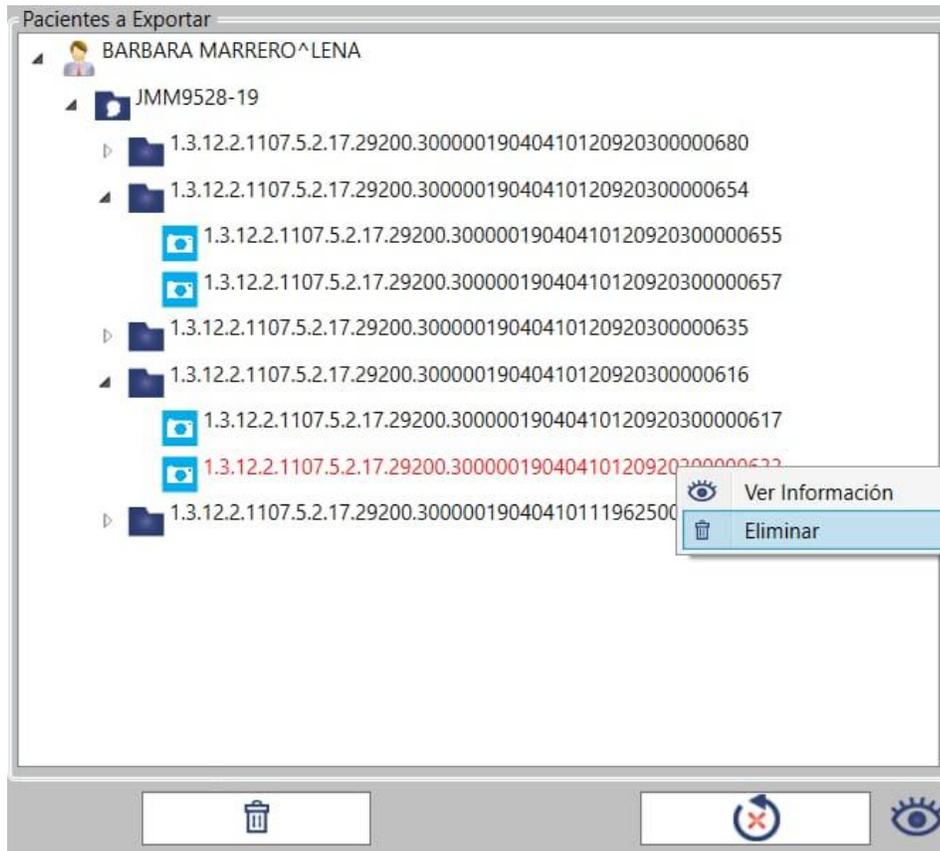


Figura13. Resultado después de borrar la información deseada.

- **Anonimizar datos deseados:**

En la Figura 13 se puede observar a la izquierda la información que contiene el estudio seleccionado, en la derecha se añaden los campos que se deseen anonimizar. Se puede apreciar que los datos seleccionados para ser anonimizados son el nombre del paciente, el identificador, la fecha de nacimiento y el sexo.

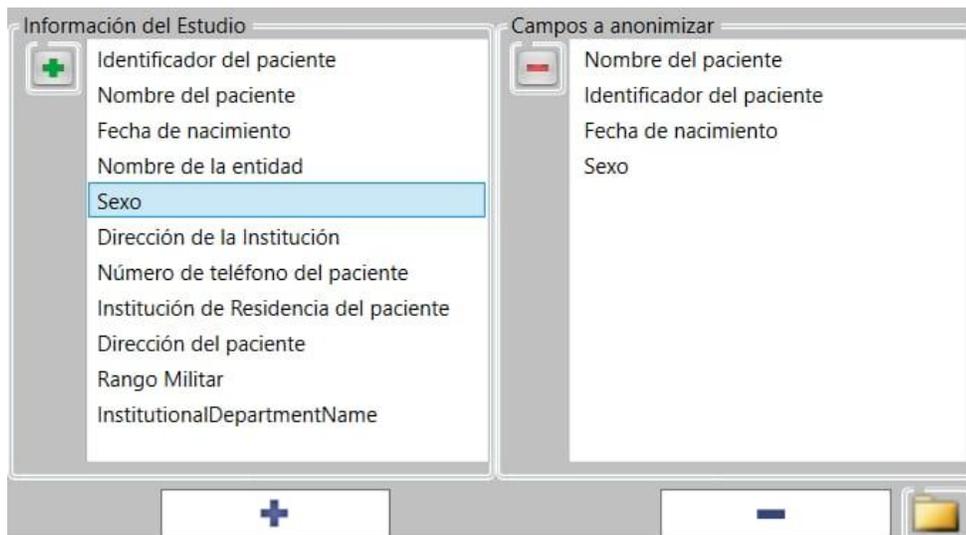


Figura14. Vista de información a anonimizar.

En la Figura 14 se puede observar en un visor Lite el resultado después de realizar la anonimización, se puede comprobar que los datos seleccionados anteriormente son los datos que se encuentran anonimizados. Para observar el estudio antes de realizar la anonimización de los datos ver el Anexo 3.

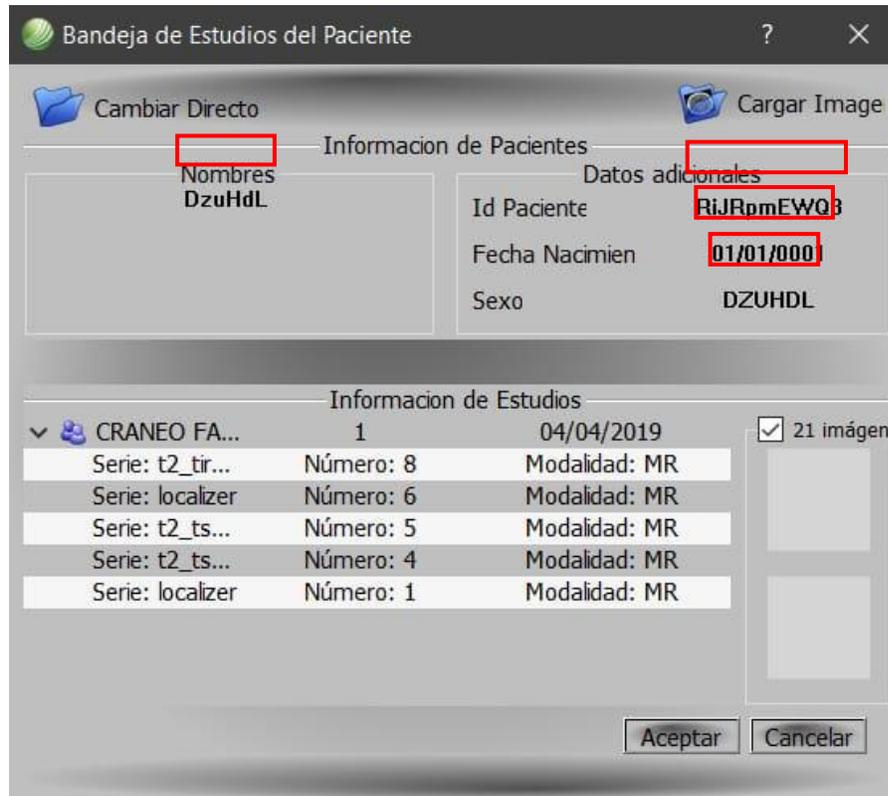


Figura15. Vista en el visor Lite de los datos anonimizados.

- **Exportar estudio:**

Al presionar el botón Exportar de la herramienta PACSBurner, se ejecuta una ventana (Ver Figura 15) que permite seleccionar la ruta de destino donde se desee guardar el estudio previamente modificado. Al presionar el botón Aceptar, se muestra una ventana refiriéndose que se ha exportado correctamente el estudio (Ver Anexo 4), después de esto se puede verificar que en la carpeta seleccionada se encuentra el estudio con todas las modificaciones realizadas anteriormente (Ver Anexo 5).

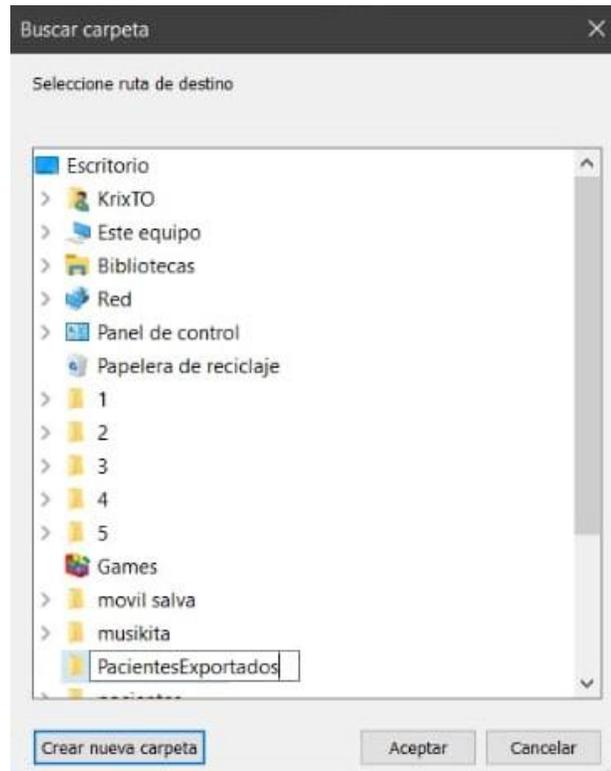


Figura16. Vista al presionar el botón Exportar.

- **Grabar estudio:**

La Figura 16 representa una ventana que se ejecuta cuando se presiona el botón Grabar en la interfaz principal de la herramienta PACSBurner. En ella se puede observar si se inserta un CD o DVD para el grabado, y el espacio que este tiene disponible, también permite borrar siempre que el CD o DVD sea regrabable y expulsarlo una vez terminadas las operaciones; la barra de color verde que se encuentra en la parte inferior de la ventana, representa el progreso de grabado. Al finalizar el grabado muestra una ventana que confirma que se ha grabado correctamente (Ver anexo 6) y se actualiza la ventana Grabar con el espacio disponible en el dispositivo correspondiente (Ver Anexo 7). Si se revisa el disco en el Explorador de archivos se puede ver que se encuentra el estudio con todas las modificaciones realizadas anteriormente (Ver Anexo 8).



Figura17. Vista de la ventana Grabar de la herramienta PACSBurner.

Análisis de los resultados obtenidos

Al comprobar las funcionalidades de la herramienta a partir de las pruebas realizadas anteriormente se puede determinar que la herramienta PACSBurner logró cumplir con los objetivos esperados. Permite eliminar la información irrelevante, anonimizar los datos deseados y posteriormente grabar o exportar dicho estudio a un dispositivo extraíble, por lo tanto los resultados arrojados son fiables para poder establecer que la herramienta desarrollada en la investigación cumple con las funcionalidades que se requieren.

Conclusiones del Capítulo

- El estándar de codificación utilizado permite una mejor comprensión del código creado, manteniéndolo legible y consistente. Permite a la vez, la posibilidad de realizar cambios con facilidad.
- El desarrollo de la herramienta PACSBurner permitió obtener resultados que permiten al XAVIA PACS-RIS anonimizar los datos de los pacientes y grabar en dispositivos extraíbles los estudios imagenológicos de manera personalizada.
- El desarrollo con estándares propició generar un componente capaz de integrarse al sistema XAVIA PACS 3.0.
- Mediante las pruebas de caja blanca y caja negra, se pudo verificar el correcto funcionamiento de la herramienta.

Conclusiones

Al culminar el desarrollo de la presente investigación, se puede concluir que:

- Los fundamentos teóricos propiciaron elaborar las bases de la investigación y adquirir los conocimientos para el desarrollo de las nuevas funcionalidades de la herramienta PACSBurner.
- El análisis de los principales estándares, bibliotecas de clases, herramientas y tecnologías existentes en la actualidad, propiciaron el desarrollo exitoso de la herramienta PACSBurner.
- La propuesta de solución propició obtener los resultados que permiten al XAVIA PACS-RIS anonimizar los datos de los pacientes y grabar en dispositivos extraíbles los estudios imagenológicos de manera personalizada.
- La validación de la herramienta desarrollada, a partir de las pruebas de *software* definidas, permitió constatar la calidad en la implementación realizada, debido al resultados satisfactorios de las mismas.
- La integración al componente PACSViewer de la herramienta desarrollada mejora el manejo y gestión de la información radiológica, asegurando la integridad y seguridad de los datos de identificación confidencial de los estudios imagenológicos.

Referencias Bibliográficas

- AMIEIRO PAZ, M., 2018. La aplicación de la Tomografía Axial Computarizada (TAC) en el método clínico. *Correo Científico Médico*, vol. 22, no. 2, pp. 196-198. ISSN 1560-4381.
- ANDERS HEJLSBERG, SCOTT WILTAMUTH, PETER GOLDE, 2012. *C# Language Specification* [en línea]. Addison-Wesley Longman Publishing Co., Inc. 75 Arlington Street, Suite 300 Boston, MA United States. S.l.: s.n. Disponible en: <https://dl.acm.org/doi/book/10.5555/861332>.
- BJORK, T.A., CHRISTOFFERSON, A.H., GREGG, L.E. y TILBURY, J.L., 2005. Universal disk format volumes with variable size [en línea]. US6961812B2. US6961812B2. Disponible en: <https://patents.google.com/patent/US6961812B2/en>. US10/263,953
- CABREJAS, J., 2008. *DICOM SimPleX®* [en línea]. 2008. S.l.: s.n. Disponible en: http://www.codemedical.es/PDF/Dicom_SimPleX_Spanish_V1_1.pdf.
- CLUNIE, D.A., 2000. *DICOM Structured Reporting*. S.l.: PixelMed Publishing. ISBN 978-0-9701369-0-9.
- DAUDINOT LÓPEZ, M. y MILLER CLEMENTE, R.A., 2016. Una solución pacs cubana bajo software libre que sirve de plataforma a especializaciones médicas. *Revista Cubana de Informática Médica*, vol. 8, no. 2, pp. 186-196. ISSN 1684-1859.
- DIEZ, L., 2003. *Reglas para codificación en C#*. 2003. S.l.: s.n.
- FIRMA-E, 2014. Técnicas de anonimizar datos personales según las Autoridades Europeas de Protección de Datos. En: Library Catalog: www.firma-e.com [en línea]. Disponible en: <https://www.firma-e.com/blog/anonimizacion-proteccion-de-datos/>.
- GAUDIOSO, V., 2010. MVVM: Model-View-ViewModel. En: V. GAUDIOSO, T. BROWN, B. RENOW-CLARKE, C. COLLINS, C. ANDRES, S. ANGLIN, M. BECKNER, E. BUCKINGHAM, G. CORNELL, J. GENNICK, J. HASSELL, M. LOWMAN, M. MOODIE, D. PARKES, J. PEPPER, F. POHLMANN, D. PUNDICK, D. SHAKESHAFT, M. WADE y T. WELSH (eds.), *Foundation Expression Blend 4 with Silverlight* [en línea]. Berkeley, CA: Apress, pp. 341-367. ISBN 978-1-4302-2974-2. Disponible en: https://doi.org/10.1007/978-1-4302-2974-2_15.
- GEOSALUD, 2014. Que es una Imagen por Resonancia Magnetica ? *Geosalud.com* [en línea]. Disponible en: https://www.geosalud.com/imagenes_medicas/imagen-resonancia-magnetica.html.
- GONZÁLES, A., 2019. Anonimización de datos personales - ¿Qué es? En: Library Catalog: ayudaleyprotecciondatos.es, *Ayuda Ley Protección Datos (LOPDGDD)* [en línea]. Disponible en: <https://ayudaleyprotecciondatos.es/2019/04/29/anonimizacion-datos-personales/>.

- HARRIS, G., DEMAREE, K. y LU, P., 2011. Methods and systems for burning compact discs [en línea]. US8042121B2. US8042121B2. Disponible en:
<https://patents.google.com/patent/US8042121B2/en>. US10/186,227
- IAN SOMMERVILLE, 2005. *Ingeniería del software*. Séptima edición. S.I.: Departamento Ciencia de la Computación e Inteligencia Artificial. ISBN 84-7829-074-5.
- KATZ, R.L., 2009. *El Papel de las TIC en el Desarrollo*. S.I.: Raul Katz. ISBN 978-84-08-09116-5.
- La magia del Imagis. En: Library Catalog: www.juventudrebelde.cu [en línea], 2018. Disponible en:
<http://www.juventudrebelde.cu/suplementos/en-red/2018-12-08/la-magia-del-imagis>.
- MACDONALD, M., 2010. *Pro WPF in VB 2010: Windows Presentation Foundation in .NET 4*. S.I.: Apress. ISBN 978-1-4302-7241-0.
- MCMURDIE, M.S., POLFER, D.A. y EVERS, D.L., 2005. Image mastering API [en línea]. US6882795B1. US6882795B1. Disponible en:
<https://patents.google.com/patent/US6882795B1/en>. US09/703,167
- MERGE, 2019. Merge Healthcare - eFilm Workstation 4.2.2, | eFilm Solutions. [en línea]. Disponible en: <https://estore.merge.com/na/estore/content.aspx?productID=444>.
- MILDENBERGER, P., EICHELBERG, M. y MARTIN, E., 2002. Introduction to the DICOM standard. En: Company: SpringerDistributor: SpringerInstitution: SpringerLabel: Springernumber: 4publisher: Springer-Verlag, *European Radiology*, vol. 12, no. 4, pp. 920-927. ISSN 1432-1084. DOI 10.1007/s003300101100.
- NATIONAL ELECTRICAL MANUFACTURERS, 2009. Digital Imaging and Communications in Medicine (DICOM). *Part 12: Storage Functions and Media Formats for Data Interchange* [en línea]. 1300 N. 17th Street Rosslyn, Virginia 22209 USA: s.n., Disponible en:
ftp://medical.nema.org/medical/Dicom/2007/07_12pu.pdf.
- NATIONAL ELECTRICAL MANUFACTURERS ASSOCIATION, 2009a. Digital Imaging and Communications in Medicine (DICOM). *Part 10: Media Storage and File Format for Media Interchange* [en línea]. 1300 N. 17th Street Rosslyn, Virginia 22209 USA: s.n., Disponible en:
http://dicom.nema.org/Dicom/2011/11_10pu.pdf.
- NATIONAL ELECTRICAL MANUFACTURERS ASSOCIATION, 2009b. Digital Imaging and Communications in Medicine (DICOM). *Part 11: Media Storage Application Profiles* [en línea]. 1300 N. 17th Street Rosslyn, Virginia 22209 USA: s.n., Disponible en:
ftp://medical.nema.org/medical/Dicom/2007/07_11pu.pdf.
- NEMA, 2013a. 6.2 Value Representation (VR). [en línea]. Disponible en:
http://dicom.nema.org/dicom/2013/output/chtml/part05/sect_6.2.html.
- NEMA, 2013b. 7 DICOM File Format. [en línea]. Disponible en:
http://dicom.nema.org/medical/dicom/current/output/chtml/part10/chapter_7.html.
- NEMA, 2013c. 7 The Data Set. [en línea]. Disponible en:
http://dicom.nema.org/dicom/2013/output/chtml/part05/chapter_7.html#sect_7.1.2.
- NEMA, 2013d. PS3.10. [en línea]. Disponible en:
<http://dicom.nema.org/medical/dicom/current/output/html/part10.html>.

- NEMA, DICOM, 2013a. 8.3 File Management Roles and Services. [en línea]. Disponible en: http://dicom.nema.org/dicom/2013/output/chtml/part10/sect_8.3.html.
- NEMA, DICOM, 2013b. 8 DICOM File Service. [en línea]. Disponible en: http://dicom.nema.org/dicom/2013/output/chtml/part10/chapter_8.html.
- NEMA, DICOM, 2013c. F.2.2.2 Example of a DICOMDIR File Structure. [en línea]. Disponible en: http://dicom.nema.org/medical/dicom/current/output/chtml/part03/sect_F.2.2.2.html.
- NEMA, DICOM, 2013d. G.2.1.2 File Set Reader. [en línea]. Disponible en: http://dicom.nema.org/medical/dicom/2017b/output/chtml/part11/sect_G.2.1.2.html.
- NEMA, DICOM, 2013e. I.2 Clinical Context. [en línea]. Disponible en: http://dicom.nema.org/medical/dicom/2018d/output/chtml/part11/sect_I.2.html.
- OSIRIX, 2010. OsiriX DICOM Viewer | The world famous medical imaging viewer. [en línea]. Disponible en: <https://www.osirix-viewer.com/>.
- OWENS, Michael, 2006. Introducing SQLite. En: , *The Definitive Guide to SQLite* [en línea]. Berkeley, CA: Apress, pp. 1-16. ISBN 978-1-4302-0172-4. Disponible en: https://doi.org/10.1007/978-1-4302-0172-4_1.
- OWENS, Mike, 2006. *The Definitive Guide to SQLite*. S.I.: Apress. ISBN 978-1-4302-0172-4.
- PACSLINK, 2005. Pacslink Eco (Dicomizador). En: Library Catalog: www.mipacs.com.ar, *miPacs* [en línea]. Disponible en: <https://www.mipacs.com.ar/productos/pacslink-eco-dicomizador/>.
- PIQUERAS, C.G., 2019. *DISOCIACIÓN/ANONIMIZACIÓN DE LOS DATOS DE SALUD* [en línea]. 2019. S.I.: s.n. Disponible en: <https://dialnet.unirioja.es/descarga/articulo/3016229.pdf>.
- PIQUERAS, J., CARREÑO, J.C. y LUCAYA, J., 1993. *Sistemas de Archivo y Comunicación de Imagen en Radiología*. [en línea]. Disponible en: <http://www.pediatricrad.info/wo/textes/pacsteor.htm>.
- PRESSMAN, R.S., 2010. *Ingeniería de Software un enfoque práctico*. Séptima Edición. Prolongación Paseo de la Reforma 1015, Torre A Piso 17, Colonia Desarrollo Santa Fe, Delegación Álvaro Obregón C.P. 01376, México, D. F. Miembro de la Cámara Nacional de la Industria Editorial Mexicana, Reg. Núm. 736: McGRAW-HILL INTERAMERICANA EDITORES, S.A. DE C.V.
- PROGRAMA TEÓRICO, 2012. *Radiología y medicina física*. 2012. S.I.: Facultad de medicina guía docente licenciatura de medicina.
- PROYECTO DESARROLLO DE LA PLATAFORMA PARA LA GESTIÓN DE LA INFORMACIÓN IMAGENOLÓGICA (PACS-RIS), 2017a. *Descripcion_de_requisitos_por_proceso_Visor*. 2017. S.I.: Universidad de las Ciencias Informáticas.
- PROYECTO DESARROLLO DE LA PLATAFORMA PARA LA GESTIÓN DE LA INFORMACIÓN IMAGENOLÓGICA (PACS-RIS), 2017b. *Especificación de requisitos de software*. 2017. S.I.: Universidad de las Ciencias Informáticas.
- RISING, L., 1998. *The Patterns Handbook: Techniques, Strategies, and Applications*. S.I.: Cambridge University Press. ISBN 978-0-521-64818-9.

- SÁNCHEZ, 2015. *Metodología de desarrollo para la Actividad productiva de la UCI*. 2015. S.l.: s.n.
- SCHMIDT, G., 2008. *Ecografía: De la imagen al diagnóstico*. S.l.: Ed. Médica Panamericana. ISBN 978-84-9835-154-5.
- SPARX SYSTEMS, 2018. Características de Enterprise Architect herramienta de modelado UML. [en línea]. Disponible en: <http://www.sparxsystems.com.ar/products/ea/features.php>.
- TERRYGLEE, 2010. Overview of Visual Studio. En: Library Catalog: docs.microsoft.com [en línea]. Disponible en: <https://docs.microsoft.com/en-us/visualstudio/get-started/visual-studio-ide>.
- VERGARA E, M., SEPÚLVEDA R, G. y VEGA T, D., 2006. TECNICA RADIOGRAFICA EN RADIOGRAFIA COMPUTADA. *Revista chilena de radiología*, vol. 12, no. 4, pp. 153-156. ISSN 0717-9308. DOI 10.4067/S0717-93082006000400003.
- VILCHES, C., 2018. Biblioguias: Gestión de datos de investigación: Anonimización de los datos. En: Library Catalog: biblioguias.cepal.org [en línea]. Disponible en: <https://biblioguias.cepal.org/c.php?g=495473&p=4961125>.
- VITREA, 2016. *VitreaCore* [en línea]. 2016. S.l.: VitalU. Disponible en: https://vitrea.med.usc.edu/vital/help/es/pdf/es_vcUG.pdf.
- WD BIDGOOD y SC HORII, 1992. Introduction to the ACR-NEMA DICOM standard. | RadioGraphics. [en línea]. Disponible en: <https://pubs.rsna.org/doi/abs/10.1148/radiographics.12.2.1561424>.

Anexos

Anexo 1. Property *CantImageCopy*.

```
public int CantImageCopy
{
    get
    {
        for (int i = 0; i < _PacientesCargados.Count; i++)
        {
            _imagenes += Pacientes[i].CantImage;
        }
        return _imagenes;
    }
    set { _imagenes = value; NotifyPropertyChanged("ImagenesToCopy"); }
}
```

Figura 18. Fragmento de código de la property *CantImageCopy*.

Anexo 2. Vista principal del PACSBurner.

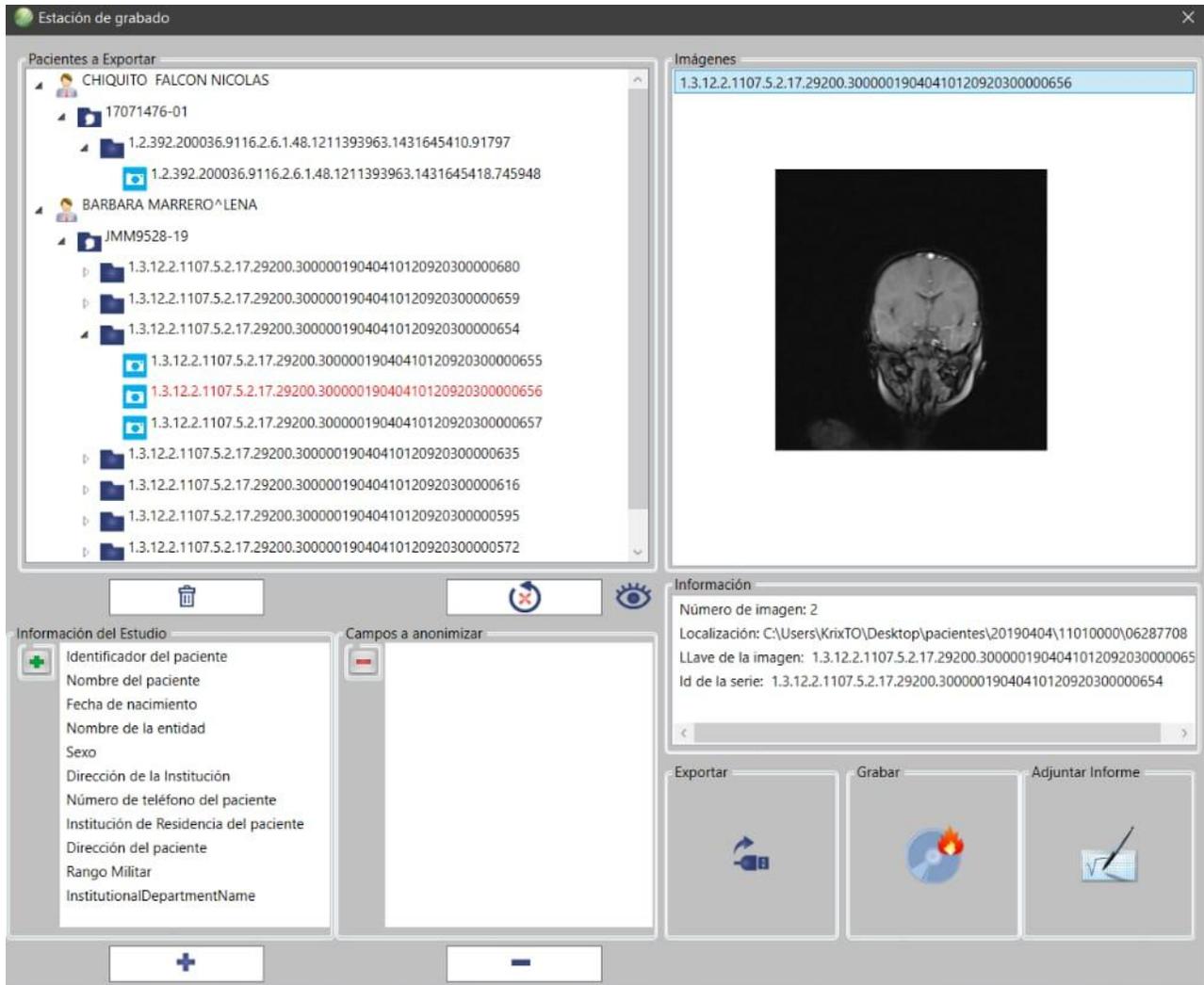


Figura 19. Vista de la ventana principal de la herramienta PACSBurner.

Anexo 3. Vista en el visor Lite de los datos antes de ser anonimizados.

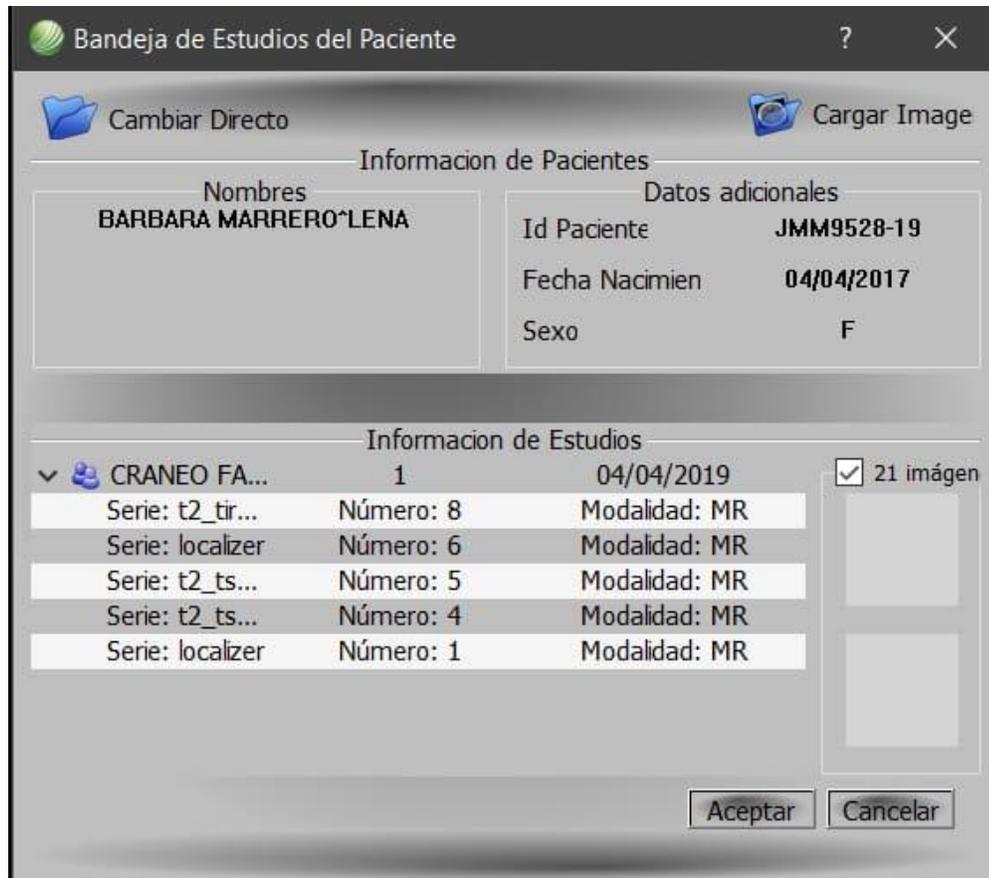


Figura 20.Datos de un estudio radiológico observado en un visor Lite.

Anexo 4. Ventana que se muestra para validar que los datos han sido exportados correctamente.



Figura 21.Exportado correctamente.

Anexo 5. Vista en el Explorador de Archivos del estudio copiado a la ruta seleccionada.

equipo > Escritorio > PacientesExportados >

Nombre	Fecha de modificación	Tipo	Tamaño
iconengines	26/08/2020 2:34	Carpeta de archivos	
imageformats	26/08/2020 2:34	Carpeta de archivos	
Images	26/08/2020 2:34	Carpeta de archivos	
platforms	26/08/2020 2:34	Carpeta de archivos	
DICOMDIR	26/08/2020 2:34	Archivo	38 KB
icudt53.dll	26/02/2020 6:26	Extensión de la ap...	21.061 KB
icuin53.dll	26/02/2020 6:26	Extensión de la ap...	3.671 KB
icuuc53.dll	26/02/2020 6:26	Extensión de la ap...	2.045 KB
libgcc_s_dw2-1.dll	26/02/2020 6:26	Extensión de la ap...	118 KB
libstdc++-6.dll	26/02/2020 6:26	Extensión de la ap...	1.003 KB
libwinpthread-1.dll	26/02/2020 6:26	Extensión de la ap...	48 KB
qt_ca.qm	26/02/2020 6:26	Archivo QM	153 KB
qt_cs.qm	26/02/2020 6:26	Archivo QM	167 KB
qt_de.qm	26/02/2020 6:26	Archivo QM	189 KB
qt_fi.qm	26/02/2020 6:26	Archivo QM	177 KB
qt_hu.qm	26/02/2020 6:26	Archivo QM	176 KB
qt_it.qm	26/02/2020 6:26	Archivo QM	182 KB
qt_ja.qm	26/02/2020 6:26	Archivo QM	151 KB
qt_ru.qm	26/02/2020 6:26	Archivo QM	177 KB
qt_sk.qm	26/02/2020 6:26	Archivo QM	176 KB
qt_uk.qm	26/02/2020 6:26	Archivo QM	177 KB
Qt5Core.dll	26/02/2020 6:26	Extensión de la ap...	4.717 KB
Qt5Gui.dll	26/02/2020 6:26	Extensión de la ap...	5.097 KB
Qt5Svg.dll	26/02/2020 6:26	Extensión de la ap...	322 KB
Qt5Widgets.dll	26/02/2020 6:26	Extensión de la ap...	6.321 KB
XAVIAViewerLite.exe	26/02/2020 6:26	Aplicación	1.788 KB

Figura 22. Estudio copiado en la ruta seleccionada.

Anexo 6. Ventana que se muestra para validar que los datos han sido grabados correctamente.



Figura 23. Grabado correctamente.

Anexo 7. Ventana Grabar al finalizar el grabado.



Figura 24. Ventana Grabar al finalizar el grabado.

Anexo 8. Vista en el Explorador de Archivos en el CD, del estudio grabado.

equipo > (E:) Unidad de DVD RW - alasPACSViewer >

Nombre	Fecha de modificación	Tipo	Tamaño
✓ Archivos actualmente en el disco (26)			
iconengines	25/08/2020 5:25	Carpeta de archivos	
imageformats	25/08/2020 5:25	Carpeta de archivos	
Images	25/08/2020 5:25	Carpeta de archivos	
platforms	25/08/2020 5:25	Carpeta de archivos	
DICOMDIR	25/08/2020 5:25	Archivo	15 KB
icudt53.dll	26/02/2020 6:26	Extensión de la ap...	21.061 KB
icuin53.dll	26/02/2020 6:26	Extensión de la ap...	3.671 KB
icuuc53.dll	26/02/2020 6:26	Extensión de la ap...	2.045 KB
libgcc_s_dw2-1.dll	26/02/2020 6:26	Extensión de la ap...	118 KB
libstdc++-6.dll	26/02/2020 6:26	Extensión de la ap...	1.003 KB
libwinpthread-1.dll	26/02/2020 6:26	Extensión de la ap...	48 KB
qt_ca.qm	26/02/2020 6:26	Archivo QM	153 KB
qt_cs.qm	26/02/2020 6:26	Archivo QM	167 KB
qt_de.qm	26/02/2020 6:26	Archivo QM	189 KB
qt_fi.qm	26/02/2020 6:26	Archivo QM	177 KB
qt_hu.qm	26/02/2020 6:26	Archivo QM	176 KB
qt_it.qm	26/02/2020 6:26	Archivo QM	182 KB
qt_ja.qm	26/02/2020 6:26	Archivo QM	151 KB
qt_ru.qm	26/02/2020 6:26	Archivo QM	177 KB
qt_sk.qm	26/02/2020 6:26	Archivo QM	176 KB
qt_uk.qm	26/02/2020 6:26	Archivo QM	177 KB
Qt5Core.dll	26/02/2020 6:26	Extensión de la ap...	4.717 KB
Qt5Gui.dll	26/02/2020 6:26	Extensión de la ap...	5.097 KB
Qt5Svg.dll	26/02/2020 6:26	Extensión de la ap...	322 KB
Qt5Widgets.dll	26/02/2020 6:26	Extensión de la ap...	6.321 KB
XAVIAViewerLite.exe	26/02/2020 6:26	Aplicación	1.788 KB

Figura 25. Estudio copiado en el CD.

Anexo 9.RF1 Grabar CD/DVD extraído del Expediente de proyecto Desarrollo de XAVIA PACS-RIS.

Descripción textual	Permite grabar en un CD o DVD los estudios deseados. Las imágenes son grabadas con un visor ligero que permite a otro médico que no posea el sistema visualizar estas imágenes en formato DICOM.
Actores	Especialista (Inicia).
Precondiciones	El paciente, estudio o serie ha sido seleccionado.
Flujo de eventos	
Flujo básico Quemar estudios	
1.	El Especialista selecciona el/los paciente(s), estudio(s) o serie(s) a quemar en la pestaña "Pacientes locales" de la Bandeja de casos.
2.	El Especialista selecciona una de las siguientes opciones: <ul style="list-style-type: none"> • La opción "Quemar" en la sección "Exportar" de la pestaña "Pacientes locales" en la Bandeja de casos. • La opción "Quemar" del menú contextual que aparece al darle clic derecho sobre los elementos seleccionados.
3.	El Sistema copia las imágenes de los estudios o de la series que se van a grabar en el disco.
4.	El Sistema lee los datos del disco que hay en la bandeja del grabador.
5.	El Sistema muestra la ventana "Estación de Quemado" con la información del disco la velocidad de escritura que permite el disco y los botones "Grabar disco". También permite "Borrar disco" (Ver Flujo básico Borrar Disco) y "Expulsar bandeja" (Ver Flujo básico Expulsar Bandeja).
6.	El Especialista selecciona la opción "Grabar disco".
7.	El Sistema graba el disco mostrando el porcentaje de grabación.
8.	El Sistema expulsa la bandeja de disco.
9.	El Sistema muestra un mensaje informando el fin de grabación.
10.	El Especialista selecciona la opción "Aceptar" o la "X" en la esquina superior derecha.
11.	El Sistema cierra la ventana.
Pos-condiciones	
1.	Se grabó el disco con las imágenes seleccionadas.
Flujos alternativos	

Figura 26.RF1 Grabar CD/DVD.

Anexo 10.RF2 Exportar a dispositivo extraído del Expediente de proyecto Desarrollo de XAVIA PACS-RIS.

Descripción textual	Permite enviar los estudios a un dispositivo externo junto con un visor ligero.
Actores	Especialista (Inicia).
Precondiciones	Se ha obtenido al menos un estudio del servidor de imágenes y se tiene abierta la Bandeja de casos.
Flujo de eventos	
Flujo básico Exportar a dispositivo	
1.	El Especialista selecciona el/los paciente(s), estudio(s) o serie(s) a quemar en la pestaña "Pacientes locales" de la Bandeja de casos.
2.	El Especialista selecciona una de las siguientes opciones: <ul style="list-style-type: none"> • La opción "Exportar a dispositivo" en la sección "Exportar" de la pestaña "Pacientes locales" en la Bandeja de casos. • La opción "Exportar a dispositivo" del menú contextual que aparece al darle clic derecho sobre los elementos seleccionados.
3.	El Sistema muestra una ventana para buscar el destino hacia dónde va a exportar la selección.
4.	El Especialista selecciona la carpeta destino.
5.	El Sistema copia las imágenes de los estudios o de la series que se van a exportar hacia el destino seleccionado junto con un visor ligero.
Pos-condiciones	
1.	Se copiaron en el destino las imágenes seleccionadas.
Flujos alternativos	
Flujo alternativo 5a Exportar a dispositivo sin espacio suficiente.	
1	El Sistema muestra el mensaje "No hay suficiente espacio en disco".

Figura 27. RF2 Exportar a dispositivo.