

Universidad de las Ciencias Informáticas

Facultad 7



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

Implementación de un servidor WADO como un servicio web

Autor : Livan Concepción Valido

Tutores : Ing. Yoel González Mesa

Ing. Leodan Vega Izaguirre

La Habana, junio de 2011

“Año 53 de la Revolución”

DECLARACIÓN DE AUTORÍA

Declaro que soy el único autor de este trabajo y autorizo a la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Para que así conste firmo la presente a los < > días del mes de Mes Junio del año 2011.

Autor:

Livan Concepción Valido

Tutores:

Ing. Yoel González Mesa

Ing. Leodan Vega Izaguirre

DATOS DE CONTACTO

Tutores:

Tutor principal:

Ing. Yoel González Mesa:

Ingeniero en Ciencias Informáticas graduado del curso 2008-2009 en la Universidad de Ciencias Informáticas de la Habana. Especialista del Departamento de Software Médico Imagenológico (SWMI) del Centro de desarrollo de Sistemas de Informática Médica (CESIM). Posee Categoría Docente de Profesor Instructor Graduado. (ygmesa@uci.cu)

Co tutor

Ing. Leodan Vega Izaguirre

Graduado de Ingeniero en Ciencias Informáticas, egresado de la UCI en el año 2008, profesor instructor. Ha impartido las asignaturas de Matemática 1, Álgebra Lineal, Práctica Profesional, Gestión de Software y Sistemas de Base de Datos. Es profesor de la Facultad 7 y se desempeña actualmente como Jefe de proyecto y Especialista de Sistemas de Bases de Datos en el Departamento de Producción de Software Médico Imagenológico del Centro de Informática Médica de la Universidad de las Ciencias Informáticas. Miembro de la Sociedad Cubana de Informática Médica. (lizaguirre@uci.cu)

RESUMEN

La constante evolución y aparición de nuevas tecnologías hacen necesario el constante reajuste de los estándares que las regulan. El estándar DICOM (Digital Imaging and Communication in Medicine) en su edición del 2006 define un servidor WADO (Web Access of DICOM Object) como una aplicación web encargada de realizar todo el procesamiento necesario para la visualización de imágenes DICOM sobre la web.

El SWMI (Departamento de Software Médico Imagenológico) del CESIM (Centro Especializado en Informática Médica) tiene entre sus proyectos el desarrollo de un visor web de imágenes DICOM. Este visor se encuentra ya en un estado avanzado de desarrollo, para el funcionamiento del mismo se implementó una aplicación web que hace las funciones de un servidor WADO.

El presente trabajo de diploma tiene como objetivo mostrar los pasos a seguir para el desarrollo de un servidor WADO como un servicio web. Se exponen las tecnologías y herramientas utilizadas en correspondencia con la arquitectura propuesta por el Centro Especializado en Informática Médica, basadas en estándares abiertos para el desarrollo de la aplicación.

Con el desarrollo de la aplicación se dispondrá de un servicio web que funcione como un servidor WADO con el que se comunique el visor web de imágenes DICOM para la lectura y procesamiento de dichas imágenes, y así se facilitaría el trabajo de los profesionales de la salud lo cual deriva en un mayor bienestar de la población en general.

Palabras claves:

Digital Imaging and Communication in Medicine, Web Access of DICOM Object, Centro Especializado en Informática Médica, Departamento de Software Médico Imagenológico, DICOM, WADO, CESIM, SWMI,

TABLA DE CONTENIDOS

1	INTRODUCCIÓN	I
	CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA	5
1.1.	SISTEMAS PACS	5
1.1.1	Modalidades de los PACS	5
1.2.	ESTÁNDAR DICOM	6
1.3.	SERVIDOR WADO	6
1.4.	ANÁLISIS DE SOLUCIONES EXISTENTES	7
1.5.	TENDENCIAS Y TECNOLOGÍAS	11
1.5.1.	Internet	11
1.5.2.	Web Services	11
1.5.3.	Lenguajes de programación	13
1.6.	METODOLOGÍA DE DESARROLLO	14
1.6.1.	Rational Unified Process	15
1.7.	HERRAMIENTAS	15
1.7.1.	Enterprise Architect	16
1.7.2.	Visual Studio 2010	16
1.8.	PATRONES DE DISEÑO	17
1.8.1.	Patrones GOF	18
1.8.2.	Patrones Creacionales	19
1.8.3.	Singleton	19
2	CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA	20
2.1	MODELO DE DOMINIO	20
2.2	ESPECIFICACIONES DE LOS REQUISITOS DE SOFTWARE	22
2.2.1	Requisitos funcionales	22
2.2.2	Requisitos no funcionales	23
2.3	DEFINICIÓN DE LOS CASOS DE USO DEL SISTEMA	24
2.3.1	Actores del sistema	24
2.4	DIAGRAMAS DE CASOS DE USO DEL SISTEMA	25
2.4.1	Casos de uso expandidos	27

3	CAPÍTULO 3: DISEÑO DEL SISTEMA.....	33
3.1	ESTILO ARQUITECTÓNICO UTILIZADO	33
3.2	PATRONES UTILIZADOS.....	34
3.2.1	Singleton.....	34
3.3	DIAGRAMAS DE CLASES.....	35
3.3.1	Services	35
3.3.2	DICOMDomain.....	36
3.3.3	DataLayerAccess.....	37
3.3.4	ImageProccesing	38
3.4	DIAGRAMAS DE SECUENCIAS.....	39
4	CAPÍTULO 4: IMPLEMENTACIÓN.....	43
4.1	DIAGRAMA DE COMPONENTES.....	43
4.2	DIAGRAMA DE DESPLIEGUE.....	44
5	CONCLUSIONES.....	45
6	RECOMENDACIONES.....	46
7	REFERENCIAS BIBLIOGRÁFICAS.....	47
8	BIBLIOGRAFÍA.....	50
9	ANEXOS.....	53
9.1	CASO DE USO APLICAR BRILLO Y CONTRASTE	53
9.2	CASO DE USO RECUPERAR UN ESTUDIO DE IMÁGENES.....	54
9.3	CASO DE USO RECUPERAR UNA SERIE DE IMÁGENES.....	55
9.4	CASO DE USO EXPORTAR IMÁGENES DICOM A OTROS FORMATOS.....	56
10	GLOSARIO DE TÉRMINOS.....	58

1 INTRODUCCIÓN

En pleno siglo XXI y con una invasión de las Tecnologías de la Informática y las Comunicaciones (TIC) en todos los sectores, la demanda de soluciones informáticas se ha hecho cada vez más grande; la calidad y las poderosas prestaciones son unas de las características que figuran entre los deseos más comunes de los clientes, conscientes de que los métodos tradicionales son obsoletos.

Desde hace un par de décadas se ha reflejado el interés médico en diagnosticar y tratar enfermedades por medio de imágenes. La tecnología médica ha evolucionado rápidamente en la modalidad de técnicas de instrumentación y procesamiento para equipos de diagnóstico, entre las más conocidas se nombran: estudios de rayos X, radiografía computada, tomografía computada, resonancia magnética, ultrasonido y medicina nuclear. Por otro lado, la forma de visualizar y archivar las imágenes producidas por dichos equipos también se ha diversificado: impresión en placas, vídeo, cintas y discos, entre otras. [1]

La digitalización de imágenes médicas representa un gran beneficio para los médicos radiólogos, ya que pueden disponer de toda la información de las imágenes rápidamente en línea. Sin embargo, médicos remitentes, como traumatólogos, cirujanos ortopédicos, especialistas internistas etc., muchas veces no tienen acceso inmediato a las imágenes médicas necesarias, ya sea en hospitales o en la asistencia a pacientes ambulatorios.

Cabe señalar que el uso de imágenes en el campo médico abarca diversos contextos: desde un consultorio médico o dental, hasta las diferentes áreas de un hospital. Además, la necesidad de que esta información se encuentre compartida se ha convertido hoy en día en un punto crítico. La evolución de la tecnología de la informática y de las redes de comunicación ha conducido al surgimiento de los sistemas PACS. [2]

Los sistemas PACS (Picture Archiving and Communication Systems) ofrecen una alternativa en el manejo de imágenes digitales en forma eficiente a través de dispositivos conectados en una red, lo cual permite proveer servicios de almacenamiento, tratamiento y transferencia de información, para dar soporte a las áreas donde se genera un volumen importante de imágenes. Estos aparecen tardíamente en la historia de la computación, ya que antes de pensar en ello, había que desarrollar el hardware y los sistemas operativos gráficos que permitieran hacerlo. Por otro lado, los algoritmos y las técnicas de optimización que han tenido

que desarrollarse para el procesamiento digital de imágenes son muy sofisticados y elaborados. En la actualidad existen muchas aplicaciones de software que permiten el procesamiento digital de imágenes, muchos de estos utiliza técnicas o algoritmos que son bien conocidos por la comunidad que trabaja en ello, pero otros utilizan sus propias variantes o técnicas nuevas que están poco documentadas. [3]

En el intercambio información de las imágenes entre los diferentes elementos de los PACS no se establecía un formato definitivo y a medidas que surgieron sistemas que contenían flujo de imágenes aparecía mas extensiones de imágenes por lo que se decidió conformar como estándar de comunicación entre los PACS el estándar DICOM (Digital Imaging and Communication in Medicine) para establecer una comunicación viable entre sistemas heterogéneos, en especial para el área médica ,DICOM añade información que es la clave para administrar, distribuir y recuperar las imágenes correctamente. [4]

Con el surgimiento de algunas aplicaciones web que implementan el estándar DICOM se le ha agregado algunos componentes a dicho estándar como la vinculación con la Web para el manejo de imágenes mediante protocolos HTTP/HTTPS, a este componente se le conoce como WADO (Web Acces to DICOM Object)

WADO es la aplicación encargada de realizar todo el procesamiento necesario para la visualización de imágenes DICOM sobre la web. Recientemente un suplemento del estándar DICOM define los puntos necesarios para la implementación de un servidor WADO con la utilización de un servicio web. [5]

Entre las limitaciones de los servidores WADO se encuentran: Se adecua a la solución Web basado en navegador, pero no a la comunicación directa con el Servidor DICOM. La consulta URL base es fácil de escribir, pero no se adaptan para ser analizada. No hay manera fácil de ayudar al desarrollo de aplicaciones a través de WSDL .Una sola instancia SOP (No Hay Manera de Recuperar TODAS Las Imágenes de Una serie / Estudio en una sola llamada).

El Departamento de Software Médico Imagenológico (SWMI) del CESIM posee entre uno de sus proyectos el desarrollo de un visor web de imágenes DICOM. Este visor se encuentra ya en un estado avanzado de desarrollo, para el funcionamiento del mismo se implementó una aplicación web que hace las funciones de un servidor WADO. Sin embargo una de las principales desventajas de emplear un servidor WADO tradicional es que solo se puede obtener una imagen en cada petición al servidor, o sea que para obtener un

estudio o una serie completa deben hacerse tantas llamadas como imágenes posea el estudio o serie, entre otras limitaciones, no sucede así cuando se utiliza un servidor WADO/WS

Dada la situación anterior el **Problema a resolver** consiste en: ¿Cómo dar solución a las limitaciones del visor web del Departamento de Software Médico Imagenológico en el manejo y procesamiento de las imágenes DICOM?

El **objeto de estudio** enmarca a todas las aplicaciones Web compatibles con el estándar DICOM. El **campo de acción** se enfoca en el Visor Web de imágenes DICOM desarrollado en el departamento de SWMI

Par dar solución al problema antes mencionado se propone **como objetivo general**: Implementar un servicio web que contenga las funciones necesarias para mejorar el funcionamiento del servidor WADO en el Visor Web.

Las **tareas** propuestas se definen como:

- Realizar una valoración del estado del arte sobre el tema de la utilización de servicios web en la implementación de un servidor WADO.
- Analizar todo lo referente a la implementación de un servidor WADO en el estándar DICOM.
- Valorar la factibilidad para el departamento del empleo de un servicio web sobre una aplicación web para el desarrollo de un servidor WADO.
- Realizar una descripción minuciosa sobre las modificaciones del estándar DICOM referentes al empleo de servicios web para implementar un servidor WADO.
- Diseñar la arquitectura de un servidor basado en un servicio web.
- Implementar un servicio web que contenga las funcionalidades de un servidor WADO.
- Integrar el servicio web con el visor web de imágenes DICOM desarrollado por el departamento.
- Realizar pruebas de calidad al servidor.

La estructura del presente documento se compone de cuatro capítulos en los que se recoge la información obtenida desde el punto de vista investigativo y lo referente a la implementación de un servidor WADO como un servicio web.

CAPÍTULO 1. Fundamentación Teórica: Contiene los aspectos esenciales para entender el entorno del problema a resolver. Se muestran los antecedentes que dieron origen al problema en cuestión, se define el objeto de estudio, la situación problemática, el estado del arte de los sistemas informáticos en el Sistema Nacional de Salud o a nivel internacional sobre la implementación de un servidor WADO como un servicio web.

CAPÍTULO 2. Características del Sistema: Presenta los argumentos principales de los procesos vinculados al problema a resolver, como son la descripción de los conceptos fundamentales asociados al dominio del problema y la identificación de los requerimientos mediante las especificación de requerimientos funcionales y no funcionales.

CAPÍTULO 3. Diseño del Sistema: Muestra los elementos básicos del diseño del sistema, mediante la justificación del uso de patrones y los diagramas de clases del diseño, las descripciones de las clases más significativas, el diagrama de clases persistentes y el modelo de datos con la explicación de las tablas y atributos.

CAPÍTULO 4. Implementación del Sistema: Describe los componentes fundamentales de todo el proceso de implementación, y argumenta la justificación de la integración con otros módulos, los métodos, estándares de diseño, codificación y excepciones, así como la representación general del diagrama de despliegue.

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

En este capítulo se exponen los principales aspectos para entender el entorno del problema a resolver, se analizan los antecedentes que le dieron origen al mismo, se define el estado del arte de los sistemas de servidores WADO y sus precedentes en aplicaciones vinculadas a la salud, además de las tendencias y tecnologías utilizadas en la elaboración del sistema.

1.1. SISTEMAS PACS

Un sistema PACS (Picture Archiving Communication System) es un sistema de archivo y comunicación de imagen. Ofrecen una alternativa en el manejo de imágenes digitales en forma eficiente a través de dispositivos conectados en una red, donde se brindan servicios de almacenamiento, tratamiento y transferencia de información, para dar soporte a las áreas donde se genera un volumen importante de imágenes.

Estos aparecen tardíamente en la historia de la computación, ya que antes de pensar en ello, había que desarrollar el hardware y los sistemas operativos gráficos que permitieran hacerlo. Por otro lado, los algoritmos y las técnicas de optimización que han tenido que desarrollarse para el procesamiento digital de imágenes son muy sofisticados y elaborados. En la actualidad existen muchas aplicaciones de software que permiten el procesamiento digital de imágenes, muchos de estos utiliza técnicas o algoritmos que son bien conocidos por la comunidad que trabaja en ello, pero otros utilizan sus propias variantes o técnicas nuevas que están poco documentadas.[6]

Es un conjunto de equipos y modalidades digitales así como sistemas informáticos de gestión hospitalaria que es utilizado en el departamento de imagenología para mejorar la efectividad de los flujos de trabajos.

1.1.1 Modalidades de los PACS

- Salas de Rayos X
- Tomógrafos
- Eco cardiógrafo
- Angiógrafos

- Ultrasonido
- Medicina Nuclear
- Escáner de placas
- Broncoscopia
- Endoscopias
- Cirugías
- Estudios de Patología

1.2. ESTÁNDAR DICOM

Creado por ACR (American College of Radiology) y NEMA (National Electrical Manufacturers Association) es el estándar reconocido mundialmente para el intercambio de imágenes médicas, pensado para el manejo, almacenamiento, impresión y transmisión de imágenes médicas. Incluye la definición de un formato de fichero y de un protocolo de comunicación de red. El protocolo de comunicación es un protocolo de aplicación que usa TCP/IP para la comunicación entre sistemas. Los ficheros DICOM pueden intercambiarse entre dos entidades que tengan capacidad de recibir imágenes y datos de pacientes en formato DICOM. [8]

DICOM permite la integración de escáneres, servidores, estaciones de trabajo, impresoras y hardware de red de múltiples proveedores dentro de un sistema de almacenamiento y comunicación de imágenes. Las diferentes máquinas, servidores y estaciones de trabajo tienen una declaración de conformidad DICOM (conformance statements) que establece claramente las clases DICOM que soportan. DICOM ha sido adoptado ampliamente por hospitales y además hace incursión en pequeñas aplicación de oficinas de dentistas y de doctores. [7]

1.3. SERVIDOR WADO

WADO es un componente que se define para recuperar instancias DICOM que utiliza los protocolos http y https. Es la aplicación encargada de realizar todo el procesamiento necesario para la visualización de imágenes DICOM sobre la web. [8]

La norma específica WADO es un servicio basado en Web para acceder y presentar objetos persistentes DICOM, como las imágenes y los informes médicos de imagen. WADO se destina para la distribución de los

resultados y las imágenes para profesionales de la salud. Proporciona un mecanismo sencillo para acceder a un objeto DICOM persistente de las páginas HTML o documentos XML, a través de HTTP / HTTPS, y utiliza UID DICOM. [9]

El servidor WADO tiene muchas aplicaciones cuando se integra en los sistemas de información médica. WADO proporciona un acceso rápido y fiable a los informes y las imágenes, dentro de los entornos informáticos, este acceso es cada vez mas basado en tecnologías web.

Dentro de las aplicaciones de WADO en los sistemas informáticos estan:

- ✓ Hacer referencia a una imagen o un informe de un registro electrónico del paciente (EPR).
- ✓ Incluye referencias a las imágenes en un e-mail (segunda opinión, el hospital a la distribución y el médico, etc.).
- ✓ Facilitar el acceso a los informes, imágenes y formas de onda por los clientes Web fuera del hospital

1.4. ANÁLISIS DE SOLUCIONES EXISTENTES.

A continuación se nombran algunos de los sistemas de visualización de imágenes DICOM sobre la web y la utilización del servidor WADO para el acceso de las imágenes DICOM en las paginas HTML y XML mediante los protocolos HTTP/HTTPS.

DICOM Viewer – RemotEye.

RemotEye fue creado por Neológica, una empresa italiana que se dedica al desarrollo de software avanzado para las soluciones medicas. [10]

Es un visor DICOM de imágenes basado en web que permite mostrar los estudios DICOM que físicamente están alejadas con respecto al sitio de visión y que son accesibles a través de Internet o de redes de Intranet. Permite la recuperación de imágenes médicas creadas por las modalidades de hospital y se almacena en archivos de imagen digital, que muestra en diferentes formas y la realización de las operaciones de procesamiento avanzado de imágenes sobre ellos.

Se puede integrar fácilmente en un sistema basado en la Web compatibles con DICOM: se basa en el applet de Java y las tecnologías Java Web Start, y se ha diseñado específicamente para ser integrado en las aplicaciones Web de terceros.

RemotEye ahora también es compatible con un nuevo mecanismo de integración, basado en los estándares HTTP y XML. Este mecanismo es especialmente adecuado cuando RemotEye tendrá la configuración de la parte frontal de visión-end para un sistema DICOM back-end o de archivo. En este modo, RemotEye envía las consultas a los servicios de fondo a través del protocolo HTTP, con base en los parámetros de búsqueda introducidos por el usuario en la ventana de búsqueda.

Después de realizar una consulta sobre el almacenamiento de archivos DICOM, el servidor envía respuestas con formato XML para RemotEye, que contiene los pacientes se pongan en venta, estudios, series e imágenes. RemotEye ahora será capaz de recuperar y mostrar los estudios pertinentes y la serie, según lo solicitado por el usuario, incluso si utiliza técnicas de compresión. Con el fin de apoyar mejor este modo la nueva integración, RemotEye es ahora también capaz de ejecutar una aplicación Java Web Start, además de como un applet de Java.

RAIM (Server, Java, PC)

El RAIM fue creado por el Complejo Hospitalario de Albacete. Es un visualizador de imagen médica digital que funciona a través de Internet y mediante un navegador web se utiliza desde cualquier navegador de páginas web, ofrece la posibilidad de distribuir exploraciones radiológicas para todo el hospital y su área de influencia. La tecnología Java permite utilizarlo con cualquier ordenador y sistema operativo (Macintosh, Linux, Unix...). [10]

Su instalación y mantenimiento son muy sencillos: se instala en un solo ordenador, juntamente con el servicio RAIM, y puede ser utilizado por todos los ordenadores que tengan acceso a este. Trabaja de forma conjunta integrada con el resto de sistemas de información del hospital que gestionan la actividad y los flujos de trabajo.

Existen 3 modalidades de RAIM, cada una de ellas con una funcionalidad distinta y de uso exclusivo para facultativos, radiólogos e informáticos: [11]

RAIMServer.

Esta aplicación es para uso exclusivo del personal de informática. La aplicación RAIM Server es la herramienta final para poder almacenar y visualizar las placas radiológicas.

Esta aplicación recibe imágenes de distintos lugares para posteriormente almacenarlas en la base de datos del servidor donde se encuentra instalado Raim Server. [11]

RAIM PC

Esta aplicación es para uso exclusivo de radiólogos. RAIM PC es la aplicación para adquirir, almacenar temporalmente, visualizar y enviar imágenes radiológicas, tanto en un entorno de trabajo intrahospitalario como inter-hospitalario. RAIM PC gestiona de forma unificada todos los datos relativos al paciente: la información personal (datos de filiación), la información clínica y las imágenes correspondientes a un tipo de exploración radiológica determinada con su informe correspondiente.

RAIM PC posibilita la incorporación de nuevos pacientes, la modificación y borrado de pacientes de la base de datos, así como la generación del informe radiológico asociado a las imágenes, y su firma, de esta forma impiden modificaciones posteriores. [11]

RAIM Java

RAIM Java es la aplicación que utilizarán los facultativos de los distintos servicios y consultas para visualizar imágenes asociadas a los informes radiológicos. Las consultas de imágenes permiten su visualización una a una o en parrilla. Se facilitan una serie de herramientas para la manipulación de las imágenes: ampliación, reducción, rotación, espejo, etc.; también se puede cambiar su ventana de visualización (Windows level) de forma totalmente interactiva con el ratón. Se incluyen herramientas sencillas de grabación en CD pensadas para el archivo personal y puntual de casos de interés. [11]

CARACTERÍSTICAS

Soporte Web: Recepción de imágenes DICOM a través de la red, línea telefónica convencional, XDSI, etc, a partir de protocolo web (HTTP), Soporte de protocolos seguros (HTTPS, IPsec...). Visualización de imagen médica digital: Radiografía convencional, tomografía computarizada (TAC), resonancia magnética, medicina nuclear. 100% DICOM compatible. Compresión Lossless Lossy adaptable. Visualización de imagen de forma individual, una a una, o en serie (dos, cuatro, etc., imágenes a la vez). Cambio de ventana de

visualización (nivel de gris) con ventanas predefinidas. Ampliación/ reducción de imagen, rotación y giro especular.

Otras características: Disponible en diferentes idiomas (castellano, catalán e inglés) y con posibilidades de ser traducido a otros idioma. Manual de integración y desarrollo. Documento DICOM Conformance Statements.

Requisitos del sistema

Software: Especificaciones Java de SUN Microsystems. Funcionamiento muy ligado al servidor de archivo de imagen digital (en el caso del archivo RAIM Server, la integración es total).

Hardware: Independiente del sistema operativo. Navegadores recomendados: Internet Explorer 4.0 o Netscape 4.0 o superiores. Configuración mínima recomendada: Intel Pentium III, 500 MHz, 128 MB de RAM, disco duro 20 GB. Monitor y tarjeta de vídeo:

Utilización diagnóstica: se recomienda el uso de monitores especiales para el diagnóstico (alta resolución, alto brillo y contraste, etc.), juntamente con la tarjeta de video adecuada para obtener el máximo rendimiento de las prestaciones del monitor.

Utilización no diagnóstica: se recomienda un monitor estándar que soporte la resolución de las imágenes con la que se trabaja y tarjetas gráficas que permiten configuraciones "True Color".

Conexión recomendable: 100 Mbps.

LEADTOOLS PACS Imaging Developer Toolkit

Software creado por LEAD Technologies, Inc., compañía estadounidense que fabrica software privativos.

Dentro de las características que presenta el software se encuentran: Soporte para IPv6. Para el desarrollo de Windows y las aplicaciones web e incluye nativo de 32 y 64 bits y PACS binarios de imágenes médicas. Añade PACS y la tecnología de imágenes médicas a las solicitudes individuales y multiproceso, Desarrollo de herramientas y componentes para formularios Windows, formularios Web Forms, WPF, C y C + +. Incluye muchos de los proyectos generales y específicos para la proyección de imagen funcional, SCP y SCU con la fuente código para C, Visual C + + (MFC), VB.NET y C #.La tecnología de imágenes disponibles como servicios WCF. Proyección de imagen disponible la

tecnología como actividades de WF.

Ninguna de las herramientas anteriormente analizadas provee una solución factible al problema propuesto por diversas causas; son software privativo y muy costosos, no permiten en ningún caso la modificación de sus funciones para adaptarlo a la problemática planteada y no cumplen con la política de la soberanía tecnológica que se lleva a cabo en Cuba, por lo que se determina llevar a cabo un estudio de las tendencias y tecnologías actuales para desarrollara una herramienta que de solución al problema científico.

1.5. TENDENCIAS Y TECNOLOGÍAS

Los elementos tecnológicos como lenguaje, arquitectura, patrones, herramientas y metodologías, relacionados con las políticas anteriormente mencionadas y seleccionados para la construcción del sistema, forman parte de las orientaciones del CESIM para el desarrollo de aplicaciones médicas.

1.5.1. Internet

Internet es un método de interconexión descentralizada de redes de computadoras implementado en un conjunto de protocolos denominado TCP/IP .Garantiza que redes físicas heterogéneas funcionen como una red lógica única, de alcance mundial. Sus orígenes se remontan a 1969, cuando se estableció la primera conexión de computadoras, conocida como ARPANET, entre tres universidades en California y una en Utah, EE.UU.

Internet es un método de interconexión descentralizada de redes de computadoras implementado en un conjunto de protocolos denominado TCP/IP .Garantiza que redes físicas heterogéneas funcionen como una red lógica única, de alcance mundial. Sus orígenes se remontan a 1969, cuando se estableció la primera conexión de computadoras, conocida como ARPANET, entre tres universidades en California y una en Utah, EE.UU. [12]

1.5.2. Web Services

Los Web Services representan una posibilidad para que dos o más programas se comuniquen entre sí sin tener que mediar intervención humana. Se ha utilizado el término Web Services para cubrir lo que se conoce como servicios de aplicaciones provistos por la Web, como el acceso a una agenda de un grupo de trabajo, o a procesar la nómina en un ASP, sin embargo el verdadero impacto que tendrán las tecnologías se sentirá cuando se empiecen a usar profundamente los estándares en las conexiones entre aplicaciones en modelos

de comercio negocio a negocio.

En últimas lo que se quiere es que cualquiera que desee uno de los servicios pueda acceder a un código de programación y colocarlo en su sitio Web, que con la suficiente seguridad en el proceso, se podría brindar una extensión del sistema a terceros. Para que cualquier tercero se pueda comunicar con el sistema, en la forma no tradicional, se efectuarían modificaciones a los programas en nuestro sistema para amoldarse a cada uno de los clientes/proveedores queriéndose conectar. Sin embargo, con el advenimiento de la tendencia de los Web Services y las tecnologías que aporta, es factible ahora si efectuar programación en cada uno de los interesados en conectarse sin preocuparse por la marca de la tecnología, siempre y cuando observe los estándares.

A manera de ejemplo, un sitio de alquiler de vehículos puede querer que en todos los sitios Web de las aerolíneas se pueda alquilar un vehículo y a la aerolínea le interesará que la persona no tenga que salirse del sitio Web de la aerolínea para efectuar la transacción. Si la empresa que arrienda vehículos tuviera que hacer un programa para cada aerolínea, estaría en dificultades, sin embargo con los Web Services y sus tecnologías, con un solo desarrollo estarían disponibles para todas las aerolíneas que se quieran incorporar al plan. Con cada nueva tendencia aparecen también nuevas "sopas de letras" que denotan las tecnologías que la soportan.

Un Web Services es un programa de software que se puede comunicar con otra aplicación a través de la red donde utilizarían para la comunicación un set de protocolos estándar: SOAP, UDDI y WSDL. SOAP (Simple Object Access Protocol, Protocolo de acceso simple a objetos) es el protocolo estándar mediante el cual se define cómo se envía la información, típicamente en formato XML, entre aplicaciones a través de la red. Este protocolo reemplaza otros que prometían hacer lo mismo pero no tuvieron aceptación, como el CORBA y el DCOM. Soap se puede utilizar para crear conexiones entre las aplicaciones y estas conexiones se pueden describir usando WSDL, Web Services Description Language (lenguaje de descripción de servicios Web). Un desarrollador puede utilizar la descripción para diseñar una aplicación que se conecte a un servicio Web.

Cuando los servicios Web sean más sofisticados, una aplicación podrá utilizar una WSDL para auto configurarse y conectarse a otros servicios Web. Estos servicios Web deben ser encontrados en la red, y es ahí donde juega el UDDI, un conjunto de protocolos y conexiones que definan un repositorio de registro donde se pueden catalogar y buscar los servicios Web y su correspondiente descripción. En un futuro, los negocios o un agente de software automático buscarán primer en los registros UDDI para encontrar

proveedores. Soap, WSDL y UDDI son tecnologías independientes de plataforma que hacen uso extenso del XML, un lenguaje estándar para definir protocolos y codificar datos utilizados por las aplicaciones para comunicarse entre sí.

Aunque el Soap, WSDL y UDDI son con frecuencia referidos como estándares, todavía no tienen la bendición de una organización reconocida como la World Wide Web Consortium (W3C). Hoy Soap avanza en el W3C para ser revisado y se espera que sus definiciones sean incluidas dentro de una nueva especificación bajo el nombre Protocolo XML o XMLP. WSDL no está tan cerca, pero ya ha sido sometido al consorcio. UDDI es manejado por el UDDI Project, pero se espera que parte de esta especificación sea remitida al consorcio. Sin estándares ampliamente aceptados por todos, para lo que se requiere que sean independientes del proveedor, los Servicios Web pueden acompañar, en el tarro de la basura, a otros intentos para unificar la lógica de los negocios, como Corba, DCOM, EDI, Java Remote Method Invocation y Unix Remote Procedure Call. Solo los servicios Web han estado tan cerca de lograr el objetivo.

Tal vez solo se queda por explicar lo que se denomina una tendencia. Es una tendencia una dirección hacia la cual trabajan proveedores y consumidores de tecnología informática. Las tecnologías y protocolos explicados arriba, son simplemente eso, una tendencia, sobre la cual se debe reflexionar y saber que hacia allá va la tecnología y que cada vez se hace más difícil desarrollar aplicaciones en casa que puedan manejar todas estas tendencias, a menos que se seleccionen los estándares adecuados. Si no desarrolla sus aplicaciones, deberá cerciorarse que su proveedor también camina en este sentido. [13]

1.5.3. Lenguajes de programación

Se seleccionó como lenguaje de programación C# (pronunciado Sharp en inglés) es un lenguaje de programación orientado a objetos desarrollado y estandarizado por Microsoft como parte de su plataforma .NET, que después fue aprobado como un estándar por la ECMA e ISO. Su sintaxis básica deriva de C/C++ y utiliza el modelo de objetos de la plataforma.NET, similar al de Java aunque incluye mejoras derivadas de otros lenguajes (entre ellos Delphi). La creación del nombre del lenguaje, C # , proviene de dibujar dos signos positivos encima de los dos signos positivos de "C++", donde se quiere dar una imagen de salto evolutivo, del mismo modo que ocurrió con el paso de C a C++. C# , como parte de la plataforma.NET, está normalizado por ECMA desde diciembre de 2001 (C# Language Specification "Especificación del lenguaje C#"). El 7 de noviembre de 2005 salió la versión 2.0 del lenguaje, que incluía mejoras tales como tipos genéricos, métodos anónimos, iteradores, tipos parciales y tipos anulables. El 19 de noviembre de 2007

salió la versión 3.0 de C#, donde destacan entre las mejoras los tipos implícitos, tipos anónimos y LINQ (Language Integrated Query -consulta integrada en el lenguaje). [14]

Las principales características que definen al lenguaje C# son:

Sencillez de uso: C# elimina muchos elementos añadidos por otros lenguajes y que facilitan su uso y comprensión, como por ejemplo ficheros de cabecera, o ficheros fuentes IDL.

Modernidad: Al ser C# un lenguaje de última generación, incorpora elementos que se ha demostrado a lo largo del tiempo que son muy útiles para el programador, como tipos decimales o booleanos, así como una instrucción que permita recorrer colecciones con facilidad (instrucción foreach). Estos elementos hay que simularlos en otros lenguajes como C++ o Java.

Orientado a objetos: C# como lenguaje de última generación es orientado a objetos. Además, C# soporta todas las características del paradigma de la programación orientada a objetos, como son la encapsulación, la herencia y el polimorfismo.

Orientado a componentes: La propia sintaxis de C# incluye elementos propios del diseño de componentes que otros lenguajes tienen que simular.

Recolección de basura: Todo lenguaje incluido en la plataforma .NET tiene a su disposición el recolector de basura.

Eficiente: En C#, todo el código incluye numerosas restricciones para garantizar su seguridad, de manera que no permiten el uso de punteros.

Compatible: Para facilitar la migración de programadores de C++ o Java a C#, no sólo se mantiene una sintaxis muy similar a la de los dos anteriores lenguajes, sino que también ofrece la posibilidad de acceder a código nativo escrito como funciones sueltas no orientadas a objetos, tales como las DLL de la API de Win32. Se decide utilizar el lenguaje de programación C#, de la plataforma .NET, para generar los artefactos de tipo software, pues C# es simple, eficaz, orientado a objetos, permite desarrollar aplicaciones rápidamente y mantiene la expresividad y elegancia de los lenguajes de tipo C.

1.6. METODOLOGÍA DE DESARROLLO.

La metodología de desarrollo es un conjunto de procedimientos, técnicas, herramientas, y un soporte documental que ayuda a los desarrolladores a realizar nuevo software. Una metodología puede seguir uno o

varios modelos de ciclo de vida, es decir, el ciclo de vida indica qué es lo que hay que obtener a lo largo del desarrollo del proyecto pero no cómo hacerlo.

1.6.1. Rational Unified Process

Rational Unified Process (RUP) es una infraestructura flexible de desarrollo de software que proporciona prácticas recomendadas probadas y una arquitectura configurable [16]. Es un proceso de desarrollo de software que contiene un conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema de software. Más que un simple proceso, el proceso de desarrollo de software es un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas software, para diferentes áreas de aplicación, diferentes tipos de organizaciones, diferentes niveles de actitud y diferentes tamaños de proyecto. [17]

Las características principales de RUP son:

- **Dirigido por casos de uso:** Los casos de uso reflejan lo que los usuarios futuros necesitan, es una facilidad que el software debe proveer a sus usuarios. A partir de aquí los casos de uso guían el proceso de desarrollo incluyendo el diseño, la implementación y las pruebas del sistema.
- **Centrado en la arquitectura:** La arquitectura involucra los elementos más significativos del sistema. Surge de las necesidades de la empresa y se refleja en los casos de uso. También se ve influida por otros factores como las plataformas de software, los sistemas operativos, protocolos y requerimientos no funcionales. El modelo de arquitectura se representa a través de vistas en las que se incluyen los diagramas de Lenguaje Unificado de Modelado (Unified Modeling Language, UML).
- **Iterativo e incremental:** RUP divide el proceso en cuatro fases, que se desarrollan en iteraciones de las actividades principales básicas de cualquier proceso de desarrollo. Las iteraciones hacen referencia a pasos en los flujos de trabajo, y los incrementos, al crecimiento del producto. Están controladas y se ejecutan de una forma planificada. Se selecciona esta metodología de desarrollo porque provee un entorno configurable, basado en estándares, permite tener claro y accesible el proceso de desarrollo que se sigue, además de ser configurado según las necesidades de la organización y del proyecto. Utiliza mejores prácticas probadas en la industria y provee a cada participante con la parte del proceso que le compete directamente.

1.7. HERRAMIENTAS

Las herramientas de desarrollo son aquellas aplicaciones que tienen cierta importancia en el desarrollo de un programa. A continuación se describen las características principales de las herramientas Enterprise Architect, Visual Studio 2008 Team System.

1.7.1. Enterprise Architect

Enterprise Architect (EA) es una herramienta flexible, completa y potente de modelado en UML bajo plataforma Windows. Provee lo más nuevo en desarrollo de sistemas, administración de proyectos y análisis de negocio [18].

EA es una herramienta multi-usuario, con bases construidas sobre la especificación UML 2.0. Diseñada para ayudar a construir software robusto y fácil de mantener. Ofrece salida de documentación flexible y de alta calidad. Provee una generación poderosa de documentos y herramientas de reporte con un editor de plantilla completo y complejos de EA con la información que se necesita en el formato que su compañía o cliente demanda [19].

Una de las características más importantes es que EA soporta generación e ingeniería inversa de código fuente para muchos lenguajes conocidos, incluyendo C++, C#, Java, Delphi, VB.Net, Visual Basic y PHP. También es posible vincularla con Visual Studio.NET, permitiéndole modelar en EA y saltar directamente al código fuente en su editor de .NET. Se decide utilizar Enterprise Architect como herramienta CASE profesional, ya que soporta el ciclo de vida completo, combina la potencia de la última especificación UML 2.1 con un alto rendimiento y una interfaz intuitiva. Además brinda características como: la generación de código en visual C# 2.0; integración con Visual Studio; soporte para pruebas; usabilidad; velocidad; y otras.

1.7.2. Visual Studio 2010

.NET es la nueva tecnología desarrollada por Microsoft con los objetivos principales de mejorar los sistemas operativos y de obtener un entorno diseñado para el desarrollo y ejecución del software en forma de servicios que puedan ser accedidos a través de Internet de forma independiente al lenguaje de programación, sistema operativo y hardware utilizados tanto para desarrollarlos como para publicarlos.

Visual Studio .NET es la herramienta de desarrollo multilenguaje más completa para construir e integrar rápidamente aplicaciones y servicios Web XML (Extensible Markup Language, Lenguaje de Marcas Ampliable). Aumenta de un modo extraordinario la productividad de los desarrolladores y crea nuevas

oportunidades de negocio. En su diseño se han integrado a fondo los estándares y protocolos de Internet, como XML y SOAP (Simple Object Access Protocol), por lo que Visual Studio .NET simplifica considerablemente el ciclo de vida del desarrollo de aplicaciones. [20]

Visual Studio 2010 tiene un elevado aumento de la productividad al escribir aplicaciones dirigidas a la nueva versión de .NET Framework, esto incluye la ampliación de tipos de proyectos, la reducción de tareas mundanas y siempre en evolución y los aspectos de equipo orientados a la ingeniería de software. [21]

A las mejoras de desempeño, escalabilidad y seguridad con respecto a la versión anterior, se agregan entre otras, las siguientes novedades:

- **Mejora en las capacidades de Pruebas Unitarias:** son ejecutas más rápido independientemente de si lo hacen en el entorno IDE o desde la línea de comandos.
- **Visual Studio Tools for Office (VSTO):** integrado con Visual Studio 2010 es posible desarrollar rápidamente aplicaciones de calidad basadas en la interfaz de usuario de Office que personalicen la experiencia del usuario y mejoren su productividad en el uso de Word, Excel, PowerPoint, Outlook, Visio, InfoPath y Project.
- **LINQ (Language Integrated Query):** nuevo conjunto de herramientas diseñado para reducir la Complejidad del acceso a Base de Datos.
- **Soluciones multiplataforma:** Visual Studio 2010 ahora permite la creación de soluciones multiplataforma adaptadas para funcionar con las diferentes versiones de .Net Framework: 2.0. (Incluido con Visual Studio 2005), 3.0 (incluido en Windows Vista) y 4.0 (incluido con Visual Studio 2010). Se decide utilizar Visual Studio 2010 como IDE de desarrollo ya que ofrece la visión de las aplicaciones clientes inteligentes, al permitir a los desarrolladores con avanzadas herramientas de desarrollo, y otras características innovadoras, la creación de aplicaciones de manera rápida a través de diversas plataformas.

1.8. PATRONES DE DISEÑO

Los patrones de diseño son la base para la búsqueda de soluciones a problemas comunes en el desarrollo de software y otros ámbitos referentes al diseño de interacción o interfaces. Un patrón de diseño es una solución a un problema de diseño. Para que una solución sea considerada un patrón debe poseer ciertas

características. Una de ellas es que debe haber comprobado su efectividad en resolver problemas similares en ocasiones anteriores. Otra es que debe ser reusable, lo que significa que es aplicable a diferentes problemas de diseño en distintas circunstancias. [22]

Son soluciones a los problemas recurrentes de diseño de software a encontrar una y otra vez en el desarrollo de aplicaciones del mundo real. Los patrones se encuentran sobre el diseño y la interacción de los objetos, así como proporcionar una plataforma de comunicación en relación con soluciones elegantes y reutilizables a los problemas de programación más frecuentes. [23]

1.8.1. Patrones GOF

La Banda de los Cuatro (GoF) los patrones son considerados como el fundamento de todos los otros patrones. Se clasifican en tres grupos: la Creación, estructurales y de comportamiento. Aquí encontrarás información sobre estos patrones importantes.

Para darle una ventaja inicial, el código fuente C # se presenta en dos formas: «estructural» y «mundo real» código utiliza los nombres de tipo estructural tal como se definen en la definición de patrones y diagramas UML. El código del mundo real proporciona situaciones del mundo real de programación en la que pueden utilizar estos patrones. [24]

Objetivos de los patrones

Los patrones de diseño pretenden:

- Proporcionar catálogos de elementos reusables en el diseño de sistemas software.
- Evitar la reiteración en la búsqueda de soluciones a problemas ya conocidos y solucionados anteriormente.
- Formalizar un vocabulario común entre diseñadores.
- Estandarizar el modo en que se realiza el diseño.
- Facilitar el aprendizaje de las nuevas generaciones de diseñadores condensando conocimiento ya existente.

Asimismo, no pretenden:

- Imponer ciertas alternativas de diseño frente a otras.

- Eliminar la creatividad inherente al proceso de diseño.

No es obligatorio utilizar los patrones, solo es aconsejable en el caso de tener el mismo problema o similar que soluciona el patrón, siempre tienen en cuenta que en un caso particular puede no ser aplicable. Abusar o forzar el uso de los patrones puede ser un error.

1.8.2. Patrones Creacionales

Los patrones creacionales abstraen el proceso de instanciación y hacen que el sistema sea independiente de la creación de los objetos. Se utiliza de manera práctica el Singleton de este grupo de patrones creacionales.

1.8.3. Singleton

Este patrón asegura la creación de una única instancia de un objeto y sólo es posible el acceso global a esa única instancia.

Tiene dos características fundamentales:

1. Restricción de acceso al constructor: con esto se logra que sea imposible crear nuevas instancias. Solo la propia clase puede crear la instancia.

Mecanismo de acceso a la instancia: el acceso a la instancia única se hace a través de un único punto bien definido, que es gestionado por la propia clase y que puede ser accedido desde cualquier parte del código en principio.

CONCLUSIONES

El estudio del proceso de creación de un servidor WADO como un servicio web con las funcionalidades del servidor WADO creado para el estándar DICOM permitió la selección de las técnicas y algoritmos a utilizar en esta investigación; además quedó evidenciada la necesidad de implementar este componente de software. Se establecieron la metodología y herramientas para el modelado y desarrollo del mismo según sus características y se realizó el estudio de los diferentes tipos de patrones de diseño con el objetivo de brindar claridad y organización en la codificación.

2 CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

En este capítulo se presentan los argumentos principales de los procesos vinculados al problema a resolver, como son la descripción de los conceptos fundamentales asociados al dominio del problema y la identificación de los requerimientos mediante las especificaciones de requerimientos funcionales y no funcionales.

2.1 MODELO DE DOMINIO

RUP define en su primera fase de desarrollo la realización del modelo de negocio, con el objetivo de comprender el entorno del cliente y detectar las mejoras potenciales en los procesos de la organización. Cuando no es posible identificar claramente los procesos del negocio, RUP propone realizar un modelo de dominio, que es un subconjunto del modelo de negocio.

Un modelo de dominio captura los tipos más importantes de objetos en el contexto del sistema. Los objetos del dominio representan los conceptos que existen o eventos que ocurren en el entorno en que trabaja el sistema. El modelo de dominio se describe específicamente mediante diagramas de clases, mediante el lenguaje de modelado UML.

Como que los procesos de negocio no están claramente definidos, se describe el negocio a través del modelo de dominio. (Fig. 1)

DICOMDomain: Administra toda la información de las imágenes DICOM adjunta en su meta data, además de especificar el acceso a los ficheros (Estudio, Series, Imagen).

DICOMFileSystem: Brinda el acceso para cargar los ficheros (Estudio, Series, Imagen).

DataLayerAccess: Establece la conexión con la Base de datos del servidor de imágenes DICOM.

2.2 ESPECIFICACIONES DE LOS REQUISITOS DE SOFTWARE

Los requisitos funcionales y no funcionales muestran las capacidades o condiciones que el sistema debe cumplir y las propiedades o cualidades que el producto debe tener, los cuales en la fase de construcción deben ser posibles de probar o verificar.

2.2.1 Requisitos funcionales

Servidor WADO del sistema AlasPACSWebViewer

Nº	Funcionalidad	Descripción	Complejidad
Paquete RF 1 < Solicitudes y respuestas de imágenes, series y estudios DICOM>			
RF 1.1	Recuperar Imagen	1-Brindar servicios para recuperar imágenes DICOM imples y multi-frame.	Alta
RF 1.2	Recuperar una serie	Brinda el acceso para recuperar una series de imágenes dado el Id serie	Alta
RF 1.3	Recuperar un estudio	Brinda el acceso para recuperar un estudio de imágenes DICOM	
RF 1.4	Acceder al meta data DICOM	Brindar acceso a toda la información de la meta data adjunta en el fichero DICOM	Alta

Nº	Funcionalidad	Descripción	Complejidad
Paquete RF 3 < Servicios para procesar imágenes DICOM>			
RF 2.1			

	Aplicar brillo y contraste a la imagen	Brindar servicio para brillo y contraste a la imagen.	Alta
RF 2.2	Aplicar filtros de inversión de los niveles de grises.	Brindar servicio para filtros de inversión de los niveles de grises.	Alta
RF 2.3	Filtrar imágenes RGB a niveles de grises.	Brindar servicio para filtrar imágenes RGB a niveles de grises.	Alta
RF 2.4	Filtrar imágenes por filtros paso bajo y paso alto	Brindar servicio para filtrar imágenes por filtros paso bajo y paso alto con algunas variantes.	Alta
RF 2.5	Filtrar imágenes por filtros de suavizados	Brindar servicio para filtrar imágenes por filtros de suavizados.	Alta
RF 2.6	Aplicar Paleta de Colores	Brinda servicios para aplicarle paletas de colores sobre la imagen seleccionada	Alta
RF 2.7	Exportar imágenes DICOM a otros formatos	Brindar servicio para Exportar imágenes DICOM a otros formatos conocidos (jpg, bmp, tiff).	Alta
RF 2.8	Aplicar Espejos sobre imagen	Brinda servicios para aplicarle un espejo sobre la imagen según el eje especificado (Vertical , Horizontal)	Alta

2.2.2 Requisitos no funcionales

Seguridad

Velar por la integridad de los datos en el servicio, a través de la consistencia de la información

Rendimiento

El tiempo de respuesta y acción de los servicios deben ser el mínimo posible, y además deben prestar especial atención a los servicios de procesamiento. Se debe garantizar, por las características del software, el acceso y la respuesta en tiempo real.

Usabilidad

Permitir la conexión concurrente a los servicios. Posibilitar el incremento de usuarios concurrentes.

Software

Servidor HTTP Apache: Servidor HTTP de código abierto para plataformas Unix (BSD, GNU/Linux, etc.), Windows, Macintosh y otras, que implementa el protocolo HTTP y la noción de sitio virtual.

Hardware

El servidor web requiera las siguientes características técnicas, por el alto volumen de información a procesar en tiempo real; requiere de buenas prestaciones en procesador, memoria y redes. Para este se proponen las siguientes características:

Procesador: Mínimo Intel Xenón 5000 series o superior.

- Es de alta prioridad que el procesador disponga de altas frecuencias.
- Front Size Bus: de preferencia 1066 MHz o superior.
- Caché L2: recomendado 4 MB o superior.
- Frecuencia: preferentemente 3GHz o superior.
- Memoria: Preferentemente 8 GB de DIMM DDR2 800 MHz.
- Almacenamiento: Mínimo 300 GB de capacidad, SATA II o SCSI.

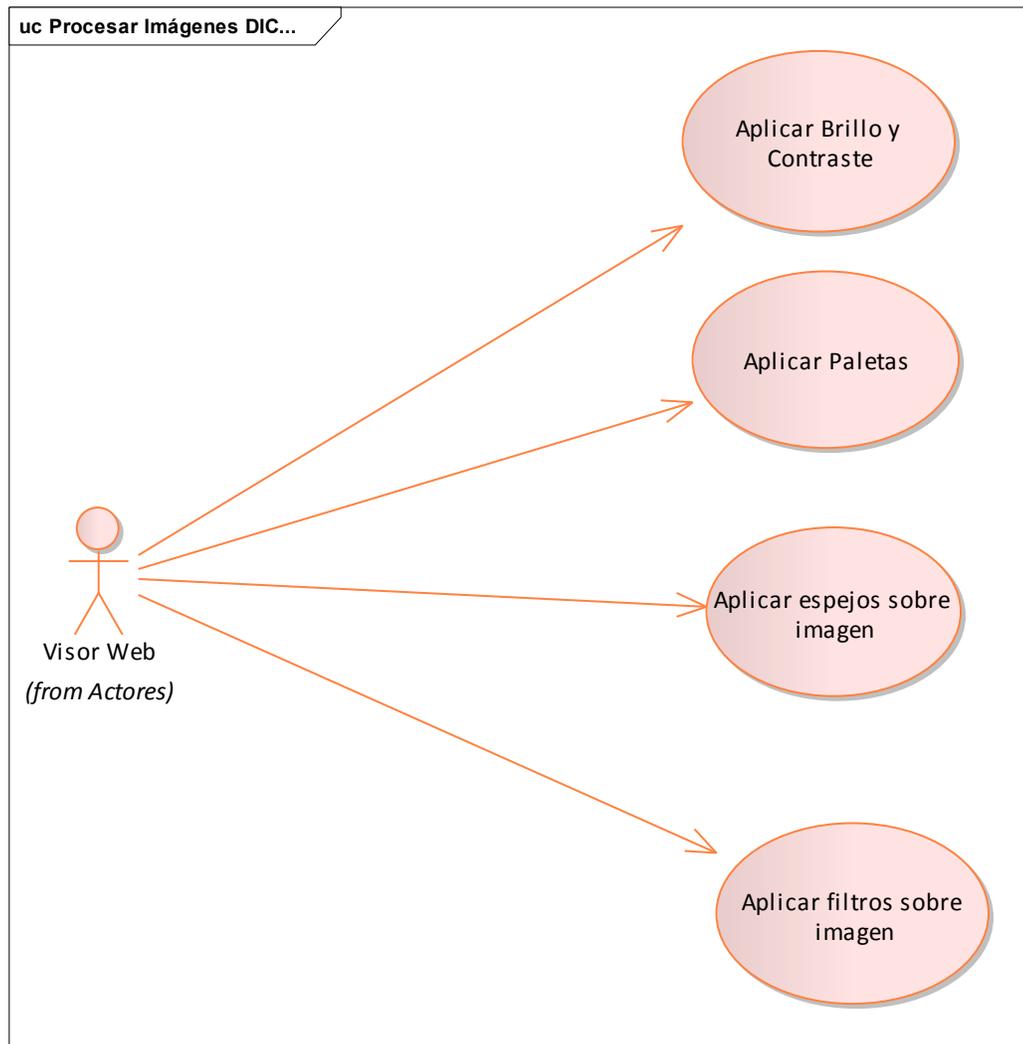
2.3 DEFINICIÓN DE LOS CASOS DE USO DEL SISTEMA

2.3.1 Actores del sistema

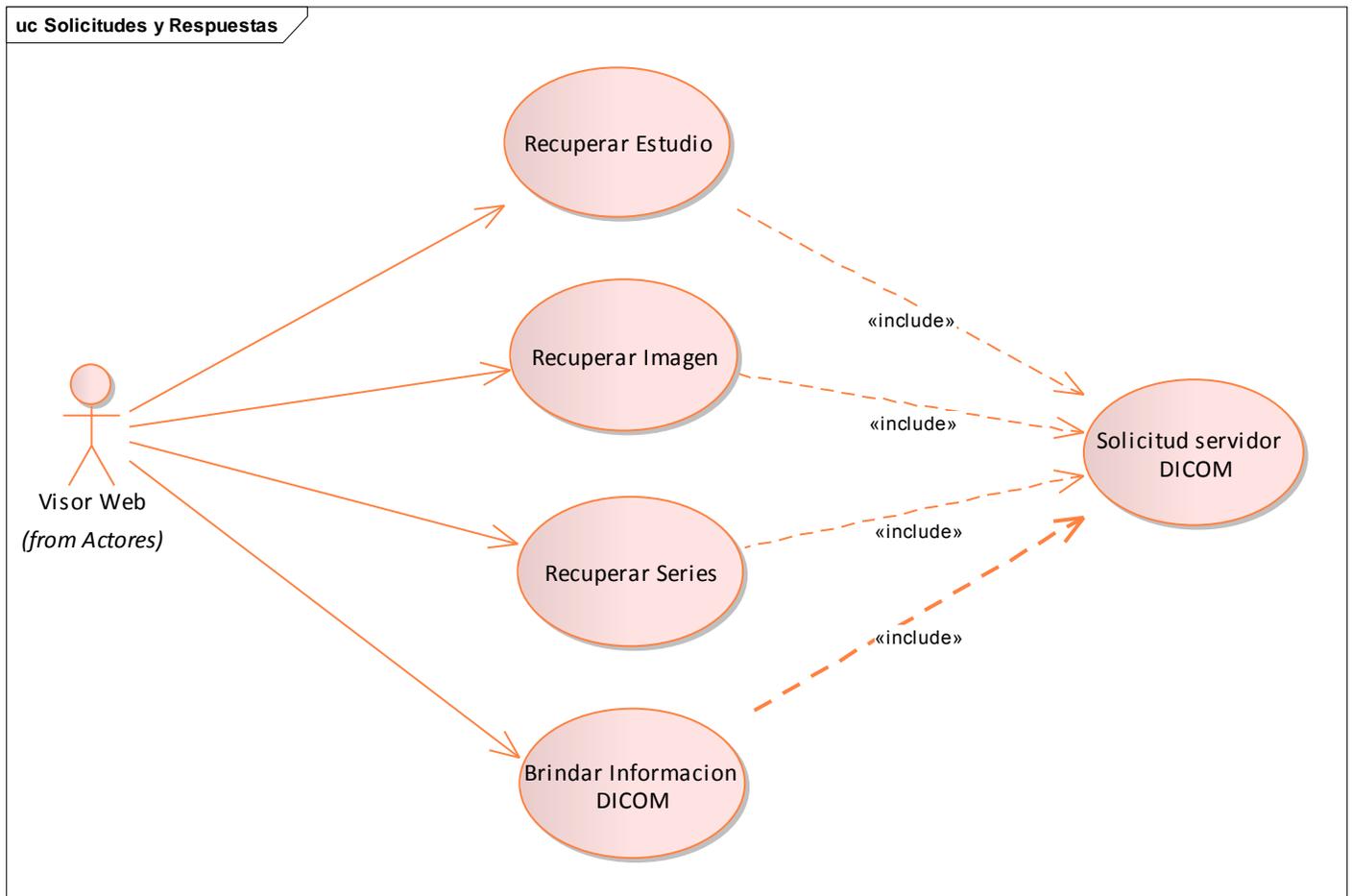
Actor	Descripción
-------	-------------

Visor Web	Sistema que interactúa con el servidor, realizando peticiones de funcionamiento a las imágenes generadas en los estudios.
Wado Server	Sistema encargado de procesar las imágenes generadas por la búsqueda.

2.4 DIAGRAMAS DE CASOS DE USO DEL SISTEMA



(Fig2.Diagrama de Casos de uso Procesar Imágenes DICOM)



(Fig2.Diagrama de Casos de uso Solicitudes y Respuestas)

2.4.1 Casos de uso expandidos

Los casos de uso expandidos son muy útiles para alcanzar un conocimiento más profundo de los procesos y de los requerimientos. Constituyen un documento narrativo que describe la secuencia de eventos de un actor que utiliza el sistema para completar un proceso.

CU Aplicar Filtros sobre imagen

Caso de Uso	Aplicar Filtros sobre imagen	
Objetivo	Su objetivo es lograr transformar con éxito la(s) imagen(es) aplicando el filtro especificado por el Visor	
Actores	Visor Web(Inicia)	
Resumen	El caso de uso inicia cuando el Visor envía al Servidor Wado la imagen que desea transformar y el filtro y el Servidor le muestra la imagen resultante.	
Complejidad	Media	
Prioridad	Auxiliar	
Referencias	RF 2.2, RF 2.3, RF 2.4, RF 2.5	
Precondiciones	Procesar imagen DICOM	
Poscondiciones	-	
Flujo de eventos		
Flujo básico < Aplicar filtros sobre imagen >		
<ol style="list-style-type: none"> 1. El Visor envía el filtro el cual se desea aplicar a la imagen visualizada. 2. El servidor realiza la operación y le envía al Visor la imagen con el filtro correspondiente. 3. Termina el caso de uso. 		
Relaciones	CU Incluidos	
	CU Extendidos	No aplicable
Requisitos funcionales	no	No aplicable
Asuntos pendientes		No aplicable

CU Recuperar imágenes DICOM

Caso de Uso	Recuperar Imágenes DICOM	
Objetivo	Su objetivo es lograr brindarle al visor una(s) imagen(es) con éxito para su visualización y procesamiento.	
Actores	Visor Web(Inicia)	
Resumen	El caso de uso inicia cuando el Visor envía el id de la(s) imagen(es) a cargar	
Complejidad	Media	
Prioridad	Auxiliar	
Referencias	RF 1.1	
Precondiciones	Solicitudes y Respuestas	
Poscondiciones	-	
Flujo de eventos		
Flujo básico < Recuperar Imágenes DICOM >		
1. El visor envía la id de las imagen(es).		
2. El Sistema le envía la(s) imagen(s) al visor por el protocolo HTTP.		
3. Termina el caso de uso.		
Relaciones	CU Incluidos	
	CU Extendidos	No aplicable
Requisitos funcionales	no	No aplicable
Asuntos pendientes	No aplicable	

CU Aplicar Espejos sobre Imagen

Caso de Uso	Aplicar Espejo sobre Imagen	
Objetivo	Su objetivo es lograr transformar con éxito la(s) imagen(es).	
Actores	Visor Web(Inicia)	
Resumen	El caso de uso inicia cuando el Visor envía la imagen que desea modificar, en qué eje aplicará el espejo y luego el Servidor le envía la imagen modificada.	
Complejidad	Media	
Prioridad	Auxiliar	
Referencias	RF 2.8	
Precondiciones	Procesar Imágenes DICOM	
Poscondiciones	-	
Flujo de eventos		
Flujo básico < Aplicar Espejo sobre Imagen >		
<ol style="list-style-type: none"> 1. El Visor envía al servidor la opción de Espejos que desee (horizontal o vertical). 2. El Servidor procesa y devuelve la imagen después de ser modificada. 3. Termina el caso de uso. 		
Relaciones	CU Incluidos	
	CU Extendidos	No aplicable
Requisitos funcionales	no	No aplicable
Asuntos pendientes	No aplicable	

CU Aplicar Paletas

Caso de Uso	Aplicar Paletas	
Objetivo	Su objetivo es lograr aplicarle con éxito la paleta especificada a una imagen.	
Actores	Visor Web(Inicia)	
Resumen	El caso de uso inicia cuando el Visor le envía el id de la imagen y el tipo de paleta que se le aplicará, al servidor, este realiza la operación sobre la imagen definida según la opción especificada por el Visor.	
Complejidad	Alta	
Prioridad	Crítico	
Referencias	RF 2.6	
Precondiciones		
Poscondiciones	-	
Flujo de eventos		
Flujo básico < Aplicar Paletas >		
	<ol style="list-style-type: none"> 1. El Visor realiza la petición de aplicar la paleta deseada sobre la imagen seleccionada 2. El Servidor realiza la operación y devuelve la imagen modificada. 3. Termina el caso de uso. 	
Relaciones	CU Incluidos	
	CU Extendidos	No aplicable
Requisitos funcionales	no	No aplicable
Asuntos pendientes		No aplicable

Se describieron algunos de los casos de usos más significativos de las funcionalidades que ofrece el servicio web, los demás casos de uso están descrito en los siguientes anexos.

- Aplicar Brillo y Contraste – [\(Ver Anexo 1\)](#).
- Recuperar un Estudio de imágenes – [\(Ver Anexo 2\)](#).
- Recuperar una Serie de Imágenes – [\(Ver Anexo 3\)](#).
- Exportar Imágenes DICOM a otros Formatos – [\(Ver Anexo 4\)](#).

CONCLUSIONES

En el presente capítulo fueron elaborados y descritos brevemente algunos de los artefactos propuestos por RUP para el desarrollo de software, tales como el modelo de dominio, definición de los requisitos funcionales y no funcionales, los actores, casos de uso, diagrama de casos de uso del sistema y la descripción textual de los mismos.

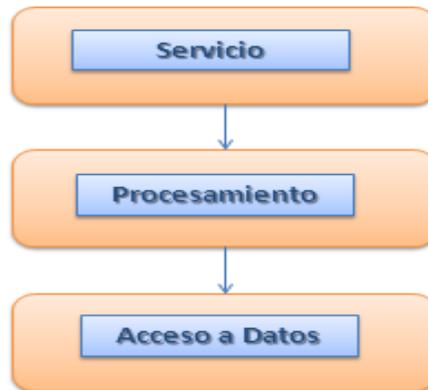
3 CAPÍTULO 3: DISEÑO DEL SISTEMA

En este Capítulo se muestran los elementos básicos del diseño del sistema, mediante la justificación del uso de patrones y los diagramas de clases del diseño, las descripciones de las clases más significativas, el diagrama de clases persistentes y el modelo de datos con la explicación de las tablas y atributos.

3.1 ESTILO ARQUITECTÓNICO UTILIZADO

La arquitectura definida por capas es un estilo de programación y su objetivo principal es la separación de la capa de presentación, capa de negocio y la capa de datos.

Ventaja principal: El desarrollo se puede llevar a cabo en varios niveles y permite distribuir el trabajo de creación de una aplicación por los mismos; cada grupo de trabajo está totalmente abstraído del resto de niveles, de forma que basta con conocer la API¹ que existe entre niveles. En el diseño de sistemas informáticos actuales se suele usar las arquitecturas multilineal o Programación por capas



(Fig5. Arquitectura WADO Services)

¹ API (Application Programming Interface) Es el conjunto de funciones y procedimientos o métodos que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción

3.2 PATRONES UTILIZADOS

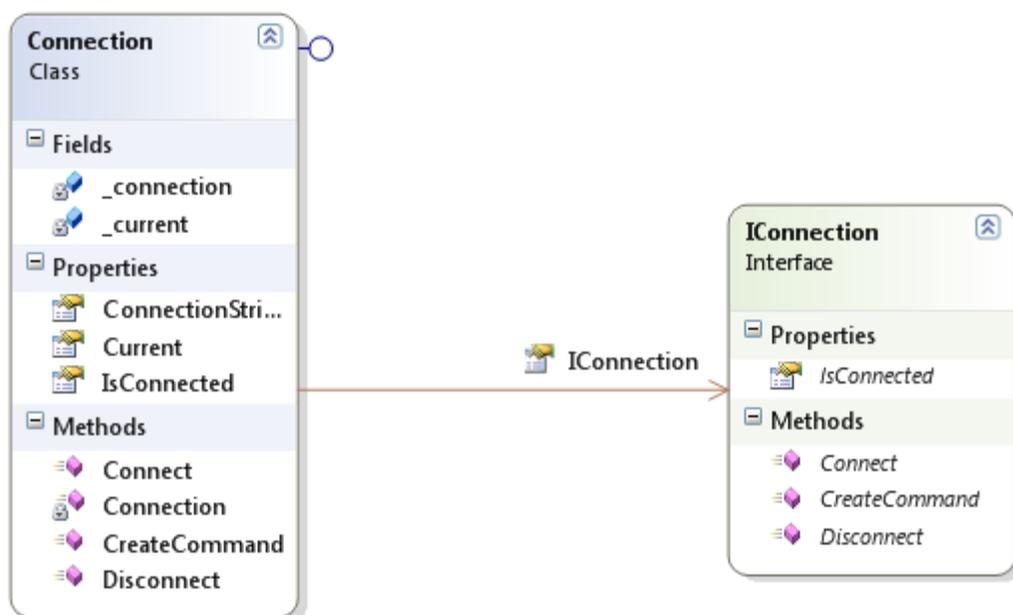
En el diseño del componente se utilizaron los siguientes patrones para dar solución a las diferentes problemáticas.

3.2.1 Singleton

Problema: Se necesita mantener de manera única la instancia de la clase Conexión

Solución: Para esto se aplica el patrón de Creación, a nivel de objetos SINGLETON.

Aplicación: Este patrón se utiliza cuando se realiza una conexión al servidor de imágenes DICOM para garantizar que sólo exista una instancia de este objeto.

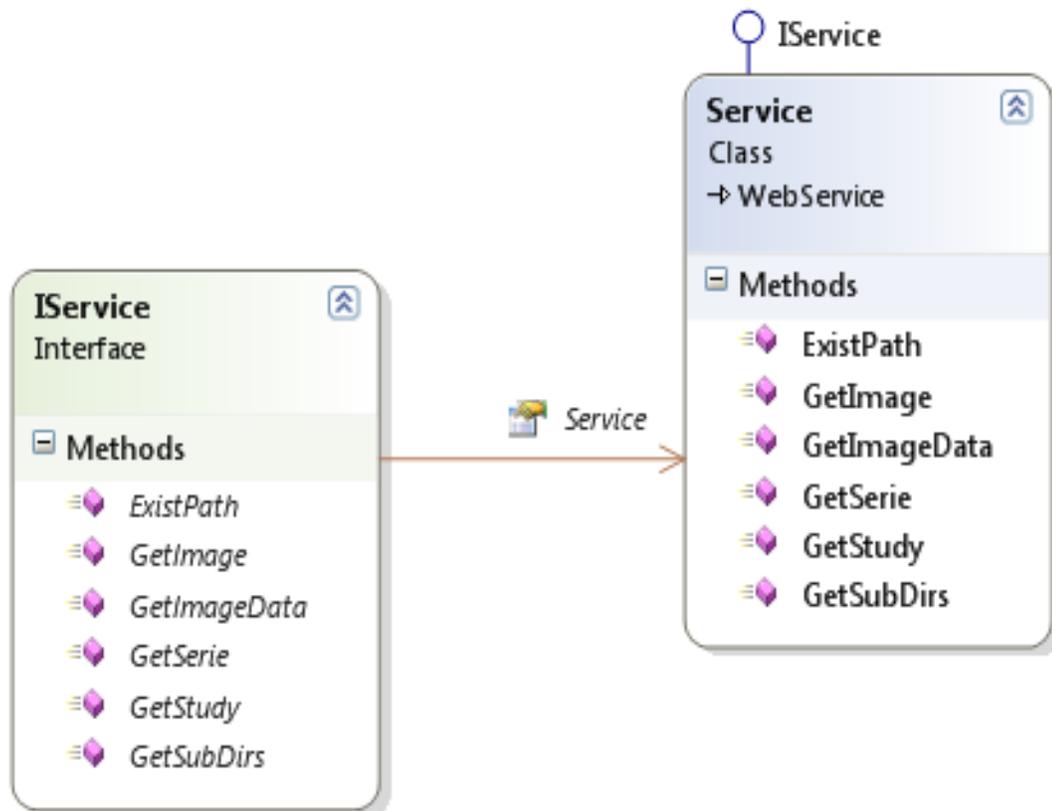


(Fig. 6 Patrón Singleton)

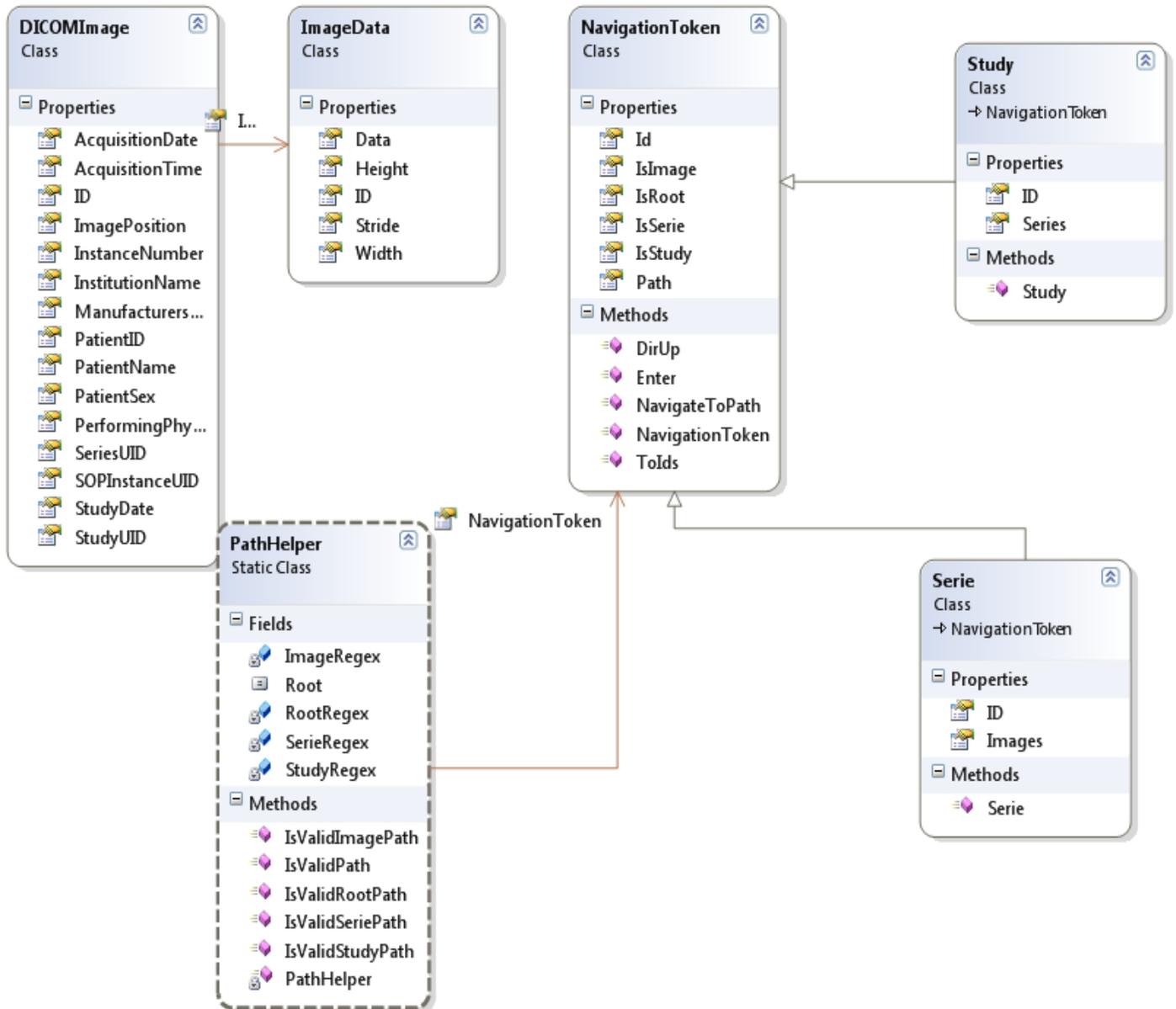
3.3 DIAGRAMAS DE CLASES

Un diagrama de clases del diseño describe gráficamente las especificaciones de las clases de software y de las interfaces de la aplicación. Además visualiza las relaciones entre las clases que se involucran en el sistema.

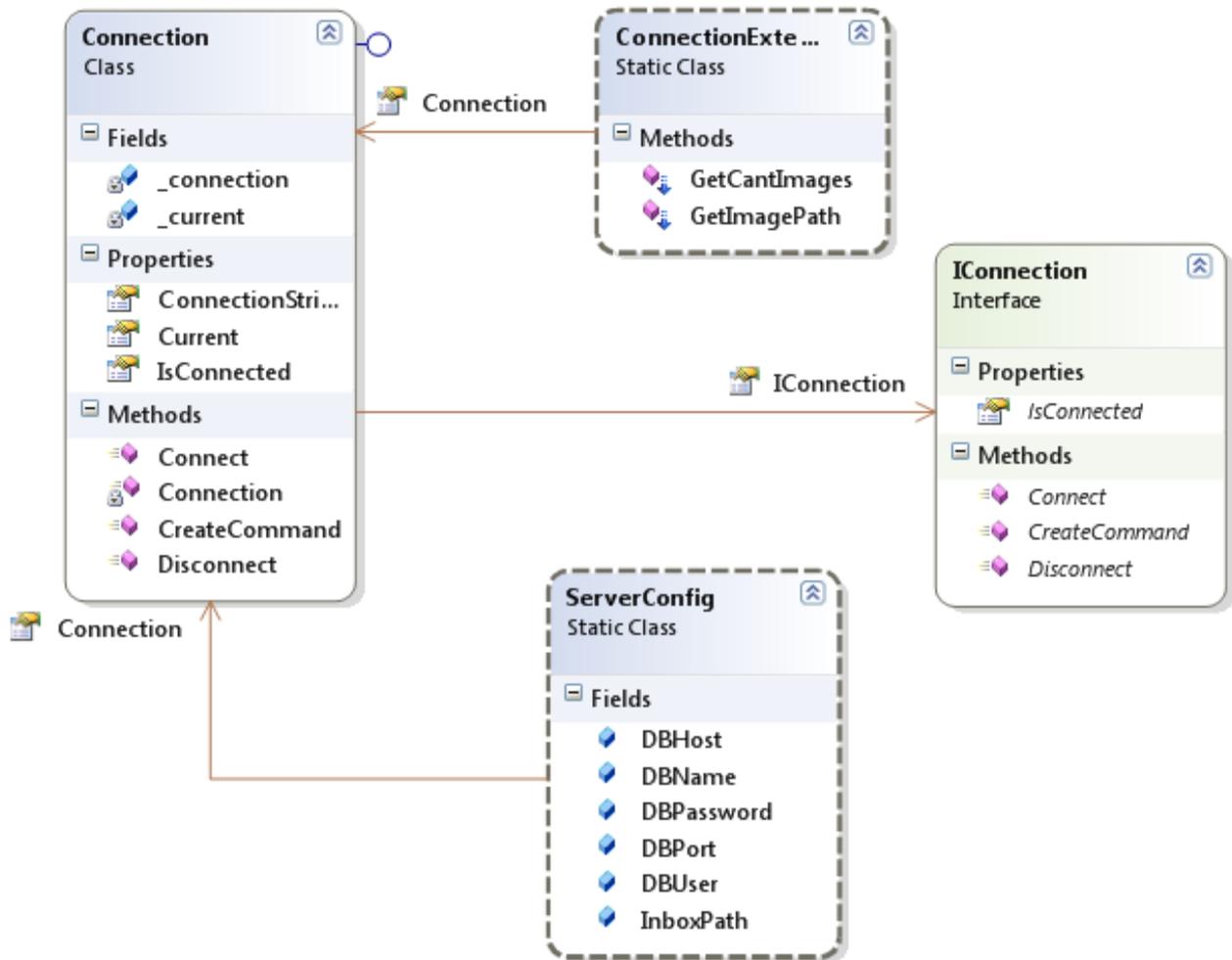
3.3.1 Services



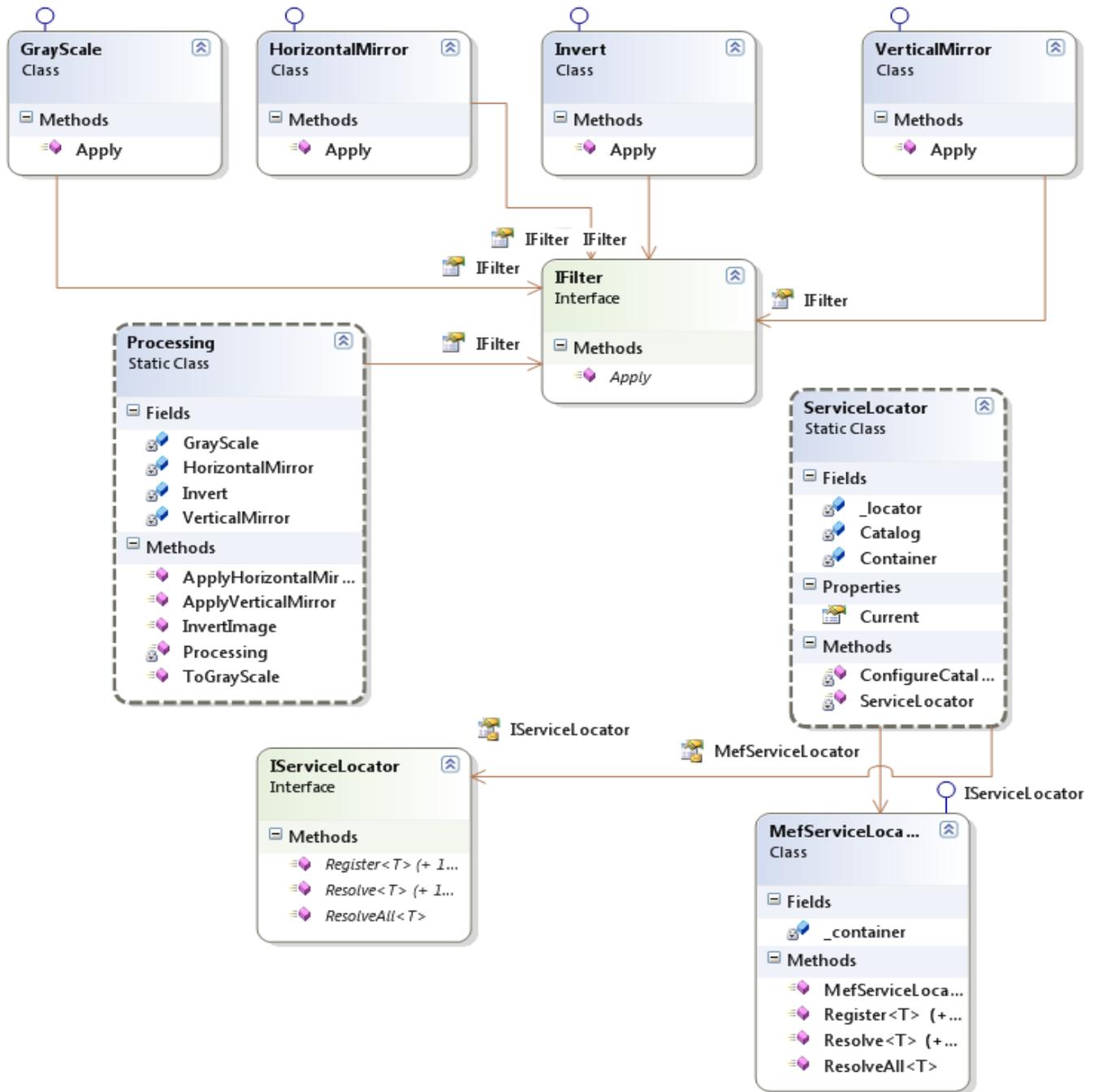
3.3.2 DICOMDomain



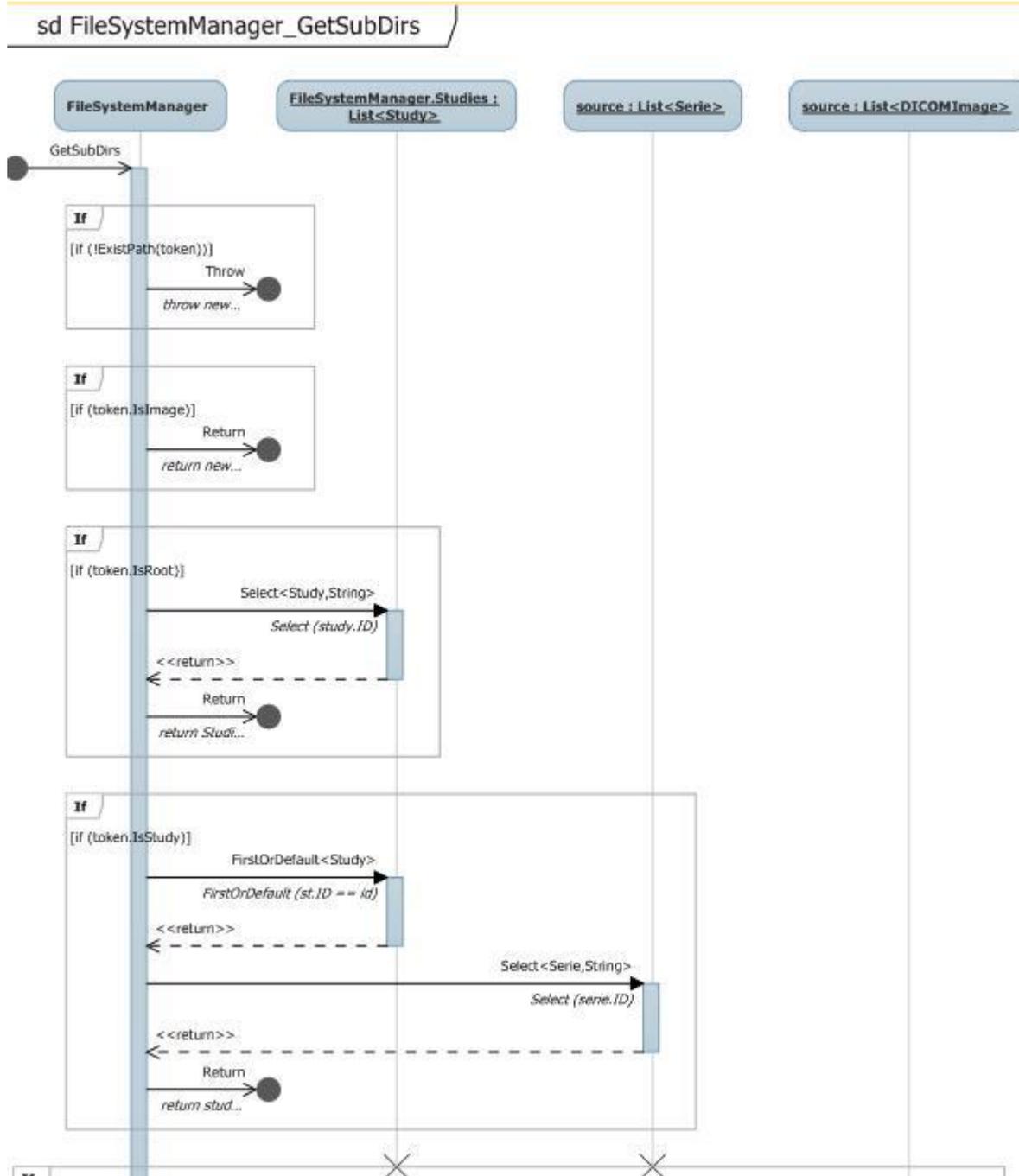
3.3.3 *DataLayerAcces*

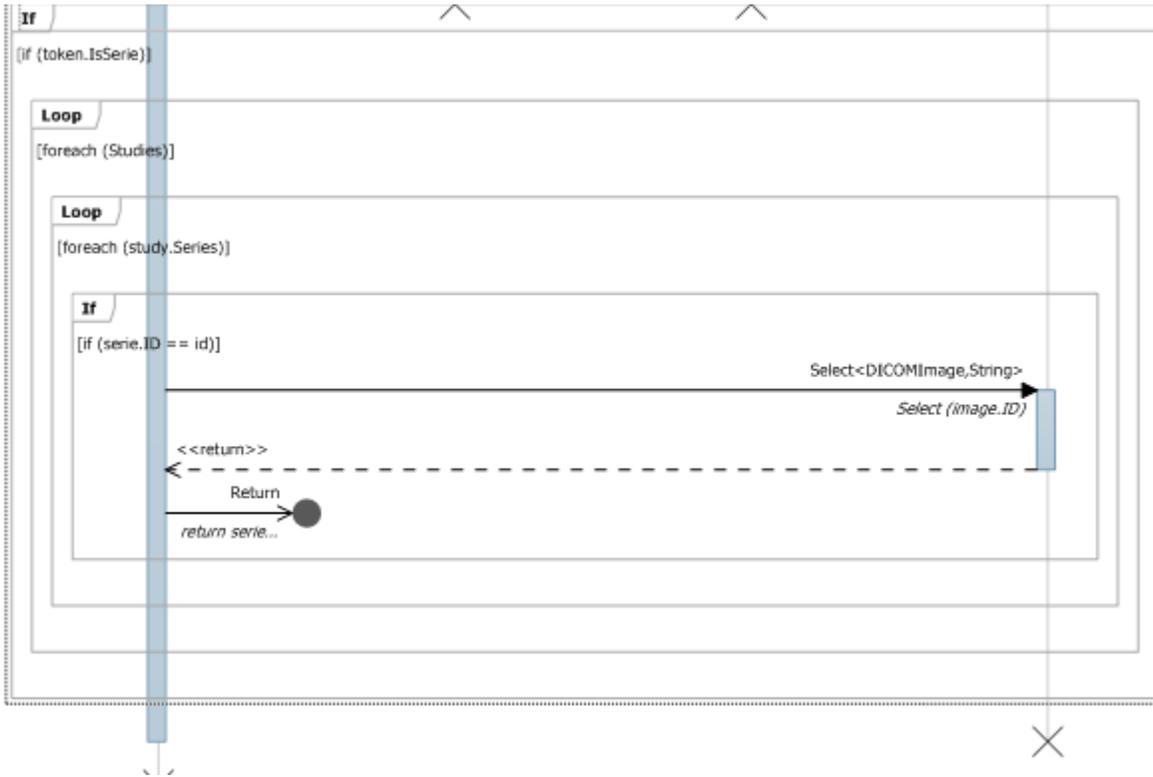


3.3.4 ImageProcessing

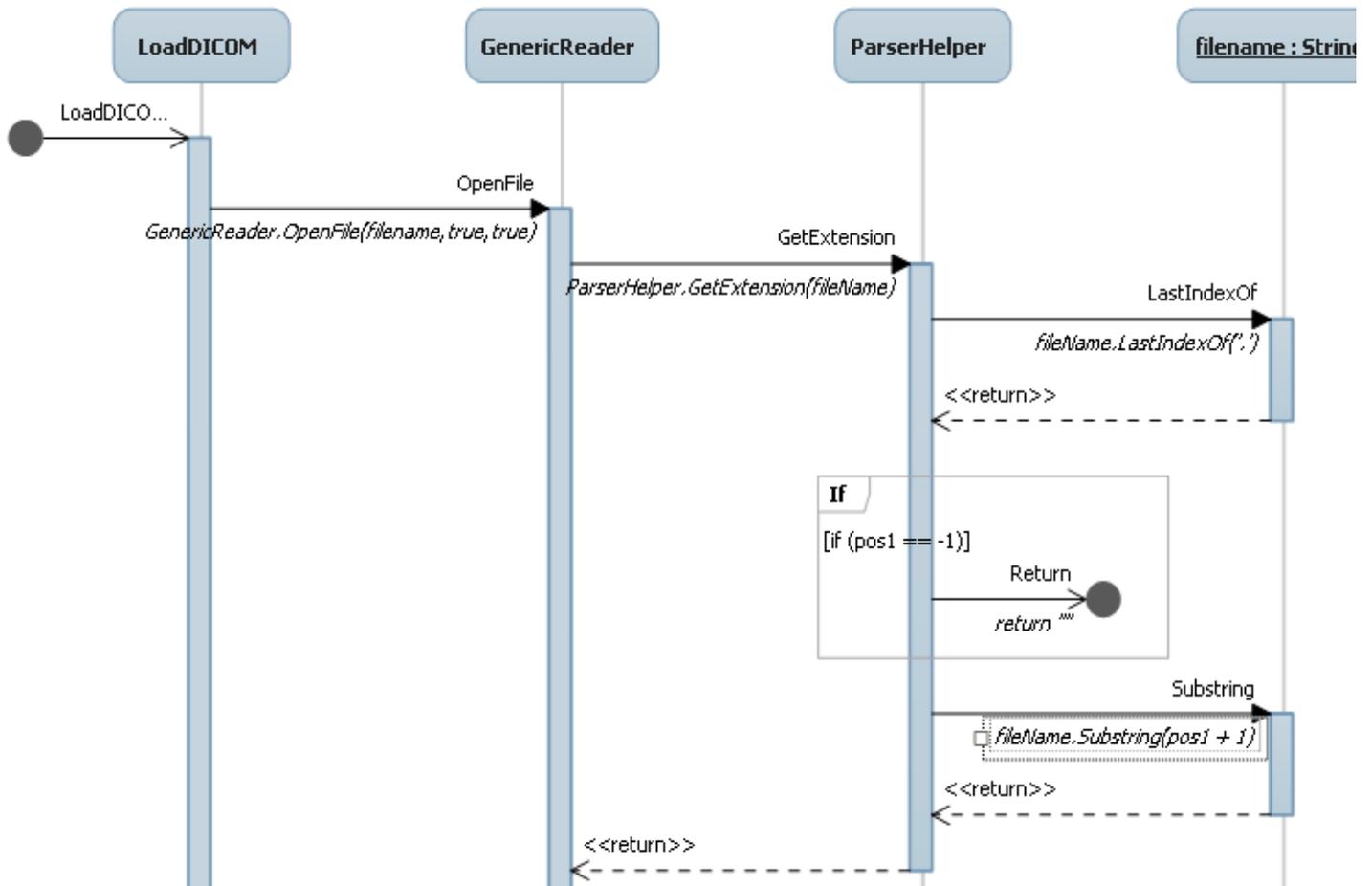


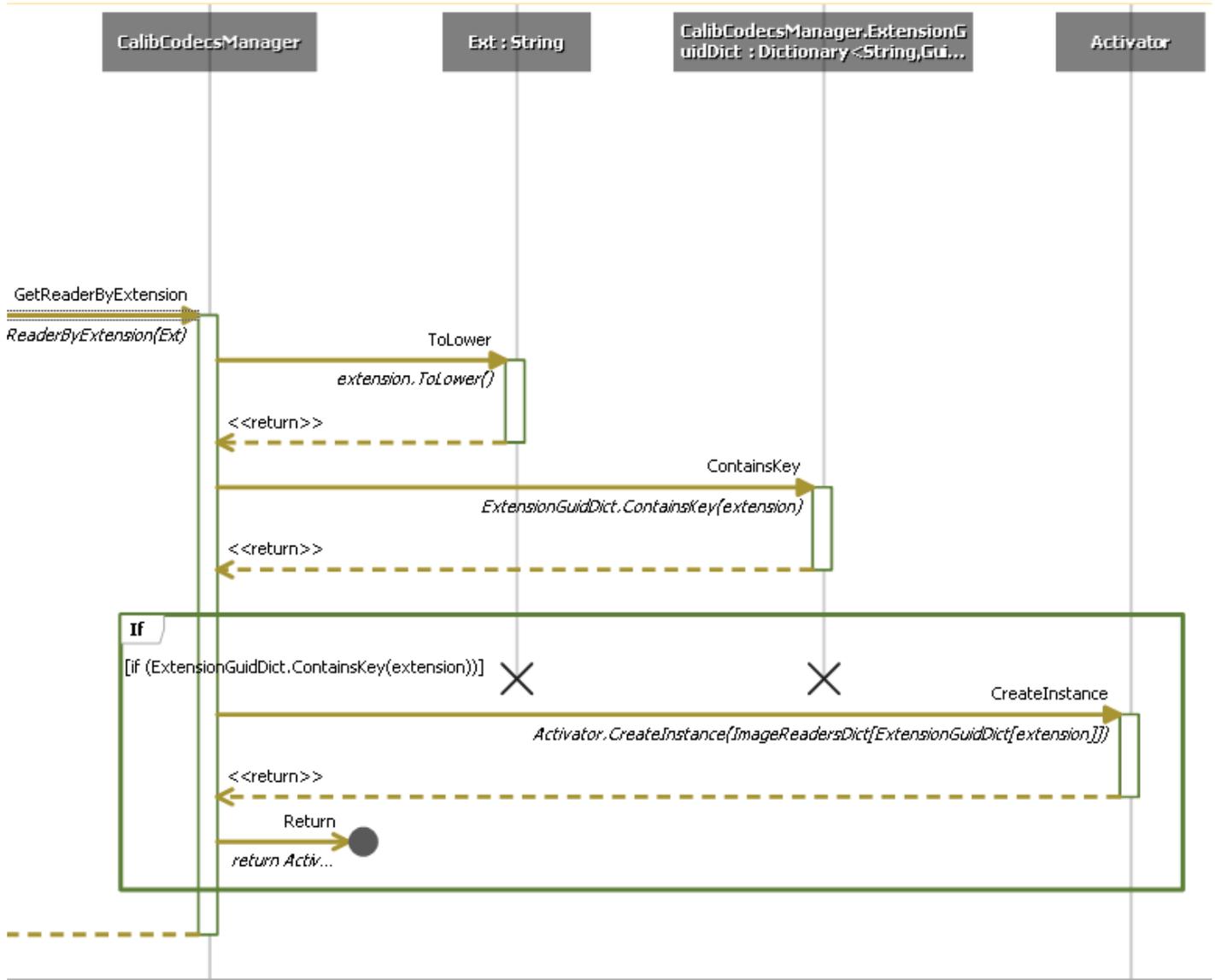
3.4 DIAGRAMAS DE SECUENCIAS





sd LoadDICOM_LoadDICOMImage





CONCLUSIONES

En este capítulo se presentó el diagrama de componentes para la representación en componentes físicos; además fueron descritos algunos fragmentos de la implementación del código del componente de software para brindar claridad y entendimiento a los desarrolladores que requieran hacer uso del mismo.

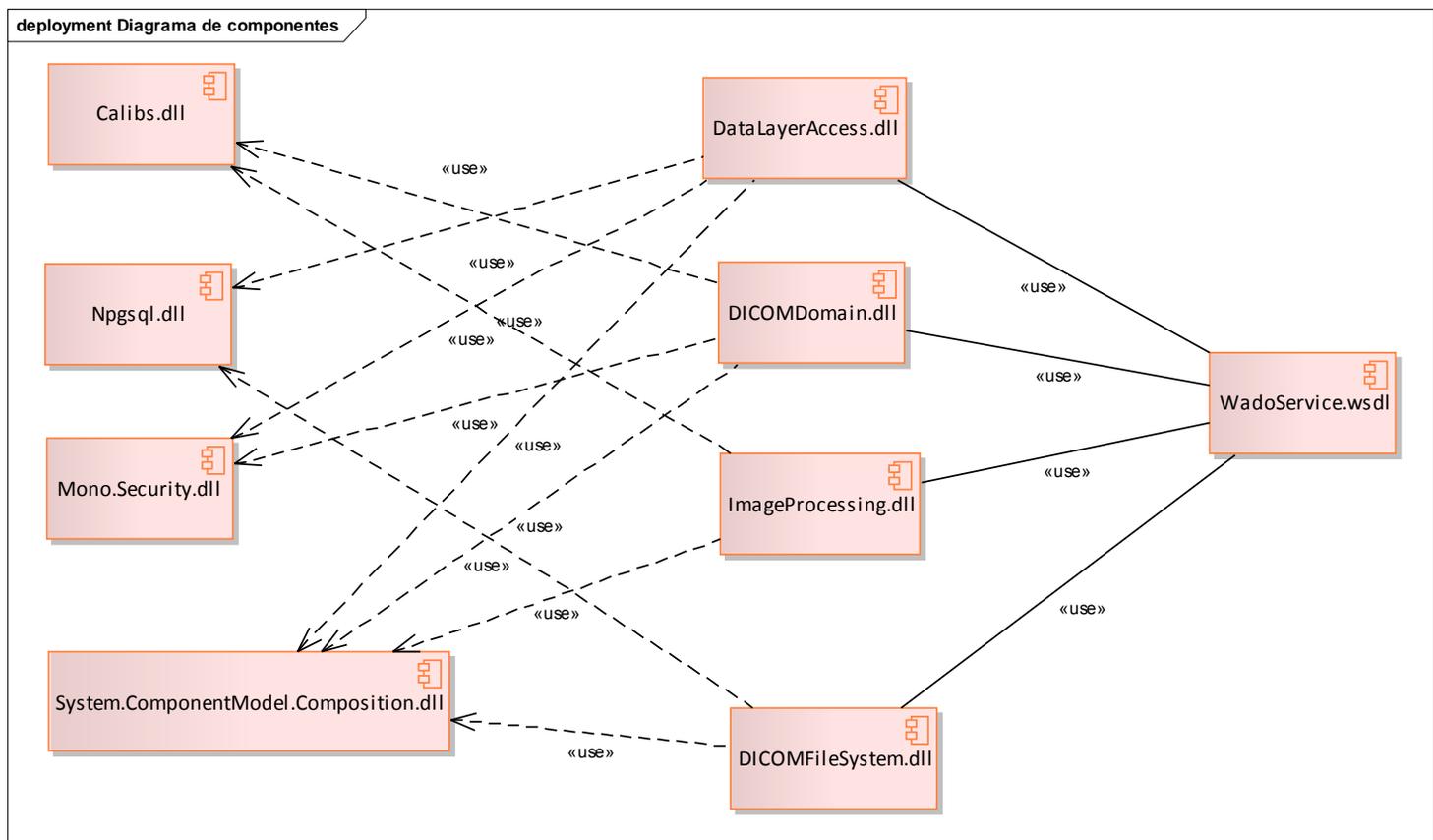
4 CAPÍTULO 4: IMPLEMENTACIÓN

El presente Capítulo describe los componentes fundamentales de todo el proceso de implementación, y argumenta la justificación de la integración con otros módulos, los métodos, estándares de diseño, codificación y excepciones, así como la representación general del diagrama de despliegue.

4.1 DIAGRAMA DE COMPONENTES

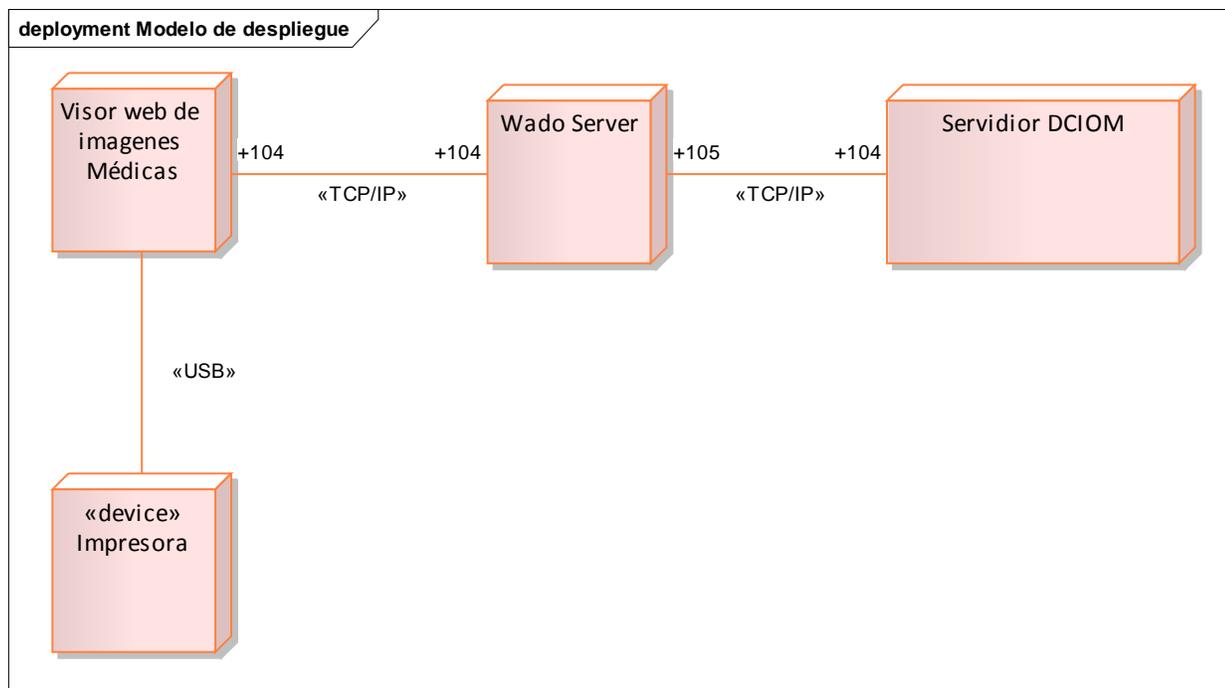
Un diagrama de componentes es utilizado para estructurar el modelo de implementación en términos de subsistemas y modelar la vista estática de un sistema, donde se describen sus elementos físicos y las relaciones entre estos. Representa cómo un sistema de software es dividido en componentes y muestra la organización y las dependencias lógicas entre estos.

En la figura muestra el diagrama de componentes del Servidor Wado como Web Services.



4.2 DIAGRAMA DE DESPLIEGUE.

Un Diagrama de Despliegue modela la arquitectura en tiempo de ejecución de un sistema. Esto muestra la configuración de los elementos de hardware (nodos) y muestra cómo los elementos y artefactos del software se trazan en esos nodos.



Conclusiones

En este capítulo se mostraron los elementos relacionados con la implementación del sistema y las relaciones que existen entre cada uno de ellos a través del diagrama de componentes. Además, se presentó el diagrama de despliegue.

5 CONCLUSIONES

El estudio del proceso de investigación permitió la selección de las técnicas y algoritmos a utilizar en la elaboración de la misma.

Durante el estudio sobre sistemas homólogos y el análisis de sus principales características quedó evidenciada la necesidad de implementar este componente de software.

Al realizarse un análisis minucioso sobre la implementación de un Web Services en el estándar DICOM se definió una arquitectura basada en el estilo arquitectónico n-capas.

La exitosa descripción de cada caso de uso dio lugar a una sencilla y rápida implementación de los mismos.

6 RECOMENDACIONES

Sobre el presente trabajo los autores recomiendan:

Implementar el componente Visor Web que consuma los servicios publicados por el servidor WADO.

Desplegar la solución en hospitales cubanos para su uso y aprovechamiento ya que representa gran utilidad para los distintos departamentos médicos que trabajan con un flujo de imágenes Dicom

7 REFERENCIAS BIBLIOGRÁFICAS

1. Wikipedia, La enciclopedia libre. [En línea] Fundación Wikimedia, Enero de 2001. [Citado el: 25 de Octubre de 2010.] Fundación Wikimedia. <http://es.wikipedia.org/wiki/DICOM>.
2. Ídem a la Referencia 1
3. **Wikipedia**. Wikipedia, La Enciclopedia Libre. [En línea] Wikipedia.org. [Citado el: 17 de Enero de 2011.] <http://es.wikipedia.org/wiki/OpenUP>. 15.
4. Ídem a la Referencia 3
5. Ídem a la Referencia 3
6. **Wikipedia**. Wikipedia, La enciclopedia Libre. [En línea] [Citado el: 14 de Enero de 2011.] http://es.wikipedia.org/wiki/C_Sharp. 14.
7. —. Wikipedia La Enciclopedia Libre. [En línea] 2001. [Citado el: 8 de Noviembre de 2010.] <http://es.wikipedia.org/wiki/DICOM>. 7.
8. Ídem a la Referencia 7
9. Ídem a la Referencia 7
10. —. Wikipedia - La Enciclopedia Libre. [En línea] Wiki.org, 2001. [Citado el: 20 de Noviembre de 2010.] http://es.wikipedia.org/wiki/Acceso_a_Internet. 12.
11. **T, José Camilo Daccach**. WIKILEARNING, Comunidades Wiki para aprender. *WIKILEARNING, Comunidades Wiki para aprender*. [En línea] 2007. [Citado el: 25 de Noviembre de 2010.] http://www.wikilearning.com/articulo/conectando_aplicaciones-web_services/16812-1. 13.
12. **NeoLogica**. NeoLogica - Hi - Tech Software Solutions. [En línea] NeoLogica.it, 2002. [Citado el: 11 de Noviembre de 2010.] <http://www.neologica.it/>. 9.
13. —. NeoLogica - Hi - Tech Software Solutions. [En línea] NeoLogica - Hi - Tech Software Solutions, 2002. [Citado el: 11 de Noviembre de 2010.] <http://eng.neologica.it/prodotti/remoteye.index>. 8.

14. **Javier Pascau, Jose A. Rodríguez-Maniega, Manuel Desco.** SOCIEDAD ESPAÑOLA DE INFORMÁTICA DE LA SALUD. *SOCIEDAD ESPAÑOLA DE INFORMÁTICA DE LA SALUD*. [En línea] 1997. [Citado el: 6 de Noviembre de 2010.] http://www.seis.es/seis/is/is45/IS45_99.pdf. 6.
15. **Innova, Grupo Soluciones.** Soluciones y Propuestas RATIONAL, Servicios, Capacitacion, Consultoria, Outsourcing Testing-Hardware IBM. [En línea] Grupo Soluciones Innova . [Citado el: 17 de Enero de 2011.] <http://www.rational.com.ar/herramientas/rup.html>. 16.
16. **IBM.** IBM Research. [En línea] IBM. [Citado el: 27 de Octubre de 2010.] <http://www.research.ibm.com/haifa/projects/software/wado/how.html>. 3.
17. **Coruña, Universidad de.** Facultad de informática de Coruña. [En línea] [Citado el: 25 de Octubre de 2010.] <http://www.fic.udc.es/files/asignaturas/58IB/DICOM-cas.pdf>. 2.
18. **Confluence, Atlassian.** dcm4che.org. [En línea] Atlassian Pty Ltd. [Citado el: 3 de Noviembre de 2010.] <http://www.dcm4che.org/confluence/display/ee2/WADO+Service>. 4.
19. **Albacete., Complejo Hospitalario de.** Complejo Universitario de Albacete - Radiología Digital. [En línea] Complejo Universitario de Albacete. [Citado el: 19 de Noviembre de 2010.] <http://www.chospab.es/ykonos/raim.htm>. 11.
20. **Albacete, Complejo Hospitalario Universitario de.** Complejo Hospitalario Universitario de Albacete. [En línea] Complejo Hospitalario Universitario de Albacete. [Citado el: 15 de Noviembre de 2010.] http://www.google.com/url?sa=t&source=web&cd=1&ved=0CBUQFjAA&url=http%3A%2F%2Fwww.chospab.es%2Fykonos%2Fpdf%2Fraim_java.pdf&rct=j&q=RAIM%20java&ei=X3YsTfaXl4Gdlgf5rOzBCw&usg=AFQjCNEJdSegDTmG4TBJV1tQtM6IDXMu0w&cad=rja.. 10.
- 21 - IBM Rational Unified Process – Grupo Soluciones Innova [En línea]2007 [Citado el: 20 de Enero de 2011] Disponible en: [7](#).
- 22 - Jacobson, Ivar; Booch, Grady; Rumbaugh, James. El Proceso Unificado de Desarrollo de Software. [Citado el: 20 de Enero de 2011.]
- 23 - APEXNET - Enterprise Architect. [En línea] 22 de Octubre de 2008 [Citado el: 20 de Enero de 2011] Disponible en: <http://www.apexnet.com.ar/index.php/product/viewProducts/24/sl=0>
- 24 - . Enterprise Architect - Herramienta de diseño UML. [En línea] 2007 [Citado el: 20 de Enero de 2011] Disponible en: <http://www.sparxsystems.com.ar/products/ea.html>

25 - Vitter,D; Templeman J. Visual Studio .NET, 2002. [Citado el: 20 de Enero de 2011]

26. - Powers, R; Snell, M. Microsoft Visual Studio 2008 UNLEASHED, 2008. [Citado el: 21 de Enero de 2011]

8 BIBLIOGRAFÍA

1. **Wikipedia.** Wikipedia, La Enciclopedia Libre. [En línea] Wikipedia.org. [Citado el: 17 de Enero de 2011.] <http://es.wikipedia.org/wiki/OpenUP>. 15.
2. **Wikipedia.** Wikipedia, La enciclopedia Libre. [En línea] [Citado el: 14 de Enero de 2011.] http://es.wikipedia.org/wiki/C_Sharp. 14.
3. —. Wikipedia La Enciclopedia Libre. [En línea] 2001. [Citado el: 8 de Noviembre de 2010.] <http://es.wikipedia.org/wiki/DICOM>. 7.
4. —. Wikipedia - La Enciclopedia Libre. [En línea] Wiki.org, 2001. [Citado el: 20 de Noviembre de 2010.] http://es.wikipedia.org/wiki/Acceso_a_Internet. 12.
5. **T, José Camilo Daccach.** WIKILEARNING, Comunidades Wiki para aprender. *WIKILEARNING, Comunidades Wiki para aprender*. [En línea] 2007. [Citado el: 25 de Noviembre de 2010.] http://www.wikilearning.com/articulo/conectando_aplicaciones-web_services/16812-1. 13.
6. **NeoLogica.** NeoLogica - Hi - Tech Software Solutions. [En línea] NeoLogica.it, 2002. [Citado el: 11 de Noviembre de 2010.] <http://www.neologica.it/>. 9.
7. —. NeoLogica - Hi - Tech Software Solutions. [En línea] NeoLogica - Hi - Tech Software Solutions, 2002. [Citado el: 11 de Noviembre de 2010.] <http://eng.neologica.it/prodotti/remoteye.index>. 8.
8. **Javier Pascau, Jose A. Rodríguez-Maniega, Manuel Desco.** SOCIEDAD ESPAÑOLA DE INFORMÁTICA DE LA SALUD. *SOCIEDAD ESPAÑOLA DE INFORMÁTICA DE LA SALUD*. [En línea] 1997. [Citado el: 6 de Noviembre de 2010.] http://www.seis.es/seis/is/is45/IS45_99.pdf. 6.
9. **Innova, Grupo Soluciones.** Soluciones y Propuestas RATIONAL, Servicios, Capacitacion, Consultoria, Outsourcing Testing-Hardware IBM. [En línea] Grupo Soluciones Innova . [Citado el: 17 de Enero de 2011.] <http://www.rational.com.ar/herramientas/rup.html>. 16.
10. **IBM.** IBM Research. [En línea] IBM. [Citado el: 27 de Octubre de 2010.] <http://www.research.ibm.com/haifa/projects/software/wado/how.html>. 3.
11. **Coruña, Universidad de.** Facultad de informática de Coruña. [En línea] [Citado el: 25 de Octubre de 2010.] <http://www.fic.udc.es/files/asignaturas/58IB/DICOM-cas.pdf>. 2.

12. **Confluence, Atlassian.** dcm4che.org. [En línea] Atlassian Pty Ltd. [Citado el: 3 de Noviembre de 2010.] <http://www.dcm4che.org/confluence/display/ee2/WADO+Service>. 4.
13. **Albacete., Complejo Hospitalario de.** Complejo Universitario de Albacete - Radiología Digital. [En línea] Complejo Universitario de Albacete. [Citado el: 19 de Noviembre de 2010.] <http://www.chospab.es/ykonos/raim.htm>. 11.
14. **Albacete, Complejo Hospitalario Universitario de.** Complejo Hospitalario Universitario de Albacete. [En línea] Complejo Hospitalario Universitario de Albacete. [Citado el: 15 de Noviembre de 2010.] http://www.google.com/url?sa=t&source=web&cd=1&ved=0CBUQFjAA&url=http%3A%2F%2Fwww.chospab.es%2Fykonos%2Fpdf%2Fraim_java.pdf&rct=j&q=RAIM%20java&ei=X3YsTfaXI4Gdlgf5rOzBCw&usg=AFQjCNEJdSegDTmG4TBJV1tQtM6IDXMu0w&cad=rja.. 10.
15. Wikipedia, La enciclopedia libre. [En línea] Fundación Wikimedia, Enero de 2001. [Citado el: 25 de Octubre de 2010.] Fundación Wikimedia. <http://es.wikipedia.org/wiki/DICOM>.
- 17- Bankman Isaac N. Handbook of Medical Imaging. Processing and Analysis. Estados Unidos, Academic Press, 2000. 901 páginas. [Citado el: 16 de Octubre de 2010].
- 18- Adams R, Bischof L. Seeded region growing, Patter Recognition, 1994, Vol. 16(no.6), 641-647. [En línea] [Citado el: 16 de Octubre de 2010] Disponible en: <http://portal.acm.org/citation.cfm?id=628615>
- 19- Patrones del “Gang of Four” - Unidad Docente de Ingeniería del Software Facultad de informática - Universidad Politécnica de Madrid
- 20- TESIS DE GRADO EN INGENIERÍA EN INFORMÁTICA TESISTA: Diego Fernando Montaldo dmontal@fi.uba.ar DIRECTOR : Profesor Ing. Guillermo Pantaleo gpantaleo@fi.uba.ar “Patrones de Diseño de Arquitecturas de Software Enterprise” (Noviembre 2005) [En línea]: <http://www.fi.uba.ar/~dmontal/> [Citado el: 18 Octubre de 2010].
- 21- Digital Imaging and Communications in Medicine (DICOM) Part 18: Web Access to DICOM Persistent Objects (WADO) [Citado el: 5 de Octubre de 2010].
- 22-[Citado]: 25 de Octubre de 2010
<http://es.wikipedia.org/wiki/DICOM>
- 23– [Citado]: 25 de Octubre de 2010

<http://www.fic.udc.es/files/asignaturas/58IB/DICOM-cas.pdf>

24– [Citado]: 27 de Octubre de 2010

<http://www.research.ibm.com/haifa/projects/software/wado/how.html>

25– [Citado]: 6 de Noviembre de 2010

<http://www.dcm4che.org/confluence/display/ee2/WADO+Service>

26– [Citado]: 8 de Noviembre de 2010

http://www.seis.es/seis/is/is45/IS45_99.pdf

27– [Citado]: 11 de Noviembre de 2010

<http://es.wikipedia.org/wiki/DICOM>

28 – [Citado]: 15 de Noviembre de 2010

<http://eng.neologica.it/prodotti/remoteye.index>

29 – [Citado]: 15 de Noviembre de 2010

<http://www.neologica.it/>

30 – [Citado]: 19 de Noviembre de 2010

[http://www.google.com/url?sa=t&source=web&cd=1&ved=0CBUQFjAA&url=http%3A%2F%2Fwww.chospab.es%2Fykonos%2Fpdf%2Ffrim_java.pdf&rct=j&q=RAIM%20java&ei=X3YsTfaXI4Gdlgf5rOzBCw&usg=AFQjCNEJdSegDTmG4TBJV1tQtM6IDXMu0w&cad=rja.](http://www.google.com/url?sa=t&source=web&cd=1&ved=0CBUQFjAA&url=http%3A%2F%2Fwww.chospab.es%2Fykonos%2Fpdf%2Ffrim_java.pdf&rct=j&q=RAIM%20java&ei=X3YsTfaXI4Gdlgf5rOzBCw&usg=AFQjCNEJdSegDTmG4TBJV1tQtM6IDXMu0w&cad=rja)

31 – [Citado]: 20 de Noviembre de 2010

9 ANEXOS

9.1 CASO DE USO APLICAR BRILLO Y CONTRASTE

Caso de Uso	Aplicar Brillo y Contraste	
Objetivo	Su objetivo es lograr transformar la imagen modificándole a la misma los valores de ancho y centro.	
Actores	Visor Web(Inicia)	
Resumen	El caso de uso inicia cuando el Visor realiza la petición al servidor sobre que imagen se realizara la opción de brillo y contraste, este realiza la operación sobre la imagen definida según la opción especificada por el Visor.	
Complejidad	Alta	
Prioridad	Crítico	
Referencias	RF 2.1	
Precondiciones		
Poscondiciones	-	
Flujo de eventos		
Flujo básico < Aplicar Brillo y Contraste >		
	<ol style="list-style-type: none"> 1. El Visor realiza la petición de aplicar brillo y contraste sobre la imagen seleccionada 2. El Servidor realiza la operación y devuelve la imagen modificada. 3. Termina el caso de uso. 	
Relaciones	CU Incluidos	
	CU Extendidos	No aplicable
Requisitos	no	No aplicable

funcionales	
Asuntos pendientes	No aplicable

9.2 CASO DE USO RECUPERAR UN ESTUDIO DE IMÁGENES

Caso de Uso	Recuperar un estudio de imágenes	
Objetivo	Su objetivo es lograr cargar un estudio completo de imágenes.	
Actores	Visor Web(Inicia)	
Resumen	El caso de uso inicia cuando el Visor realiza la petición al servidor sobre el estudio que desea cargar, este realiza la operación definida por el Visor.	
Complejidad	Alta	
Prioridad	Crítico	
Referencias	RF 1.3	
Precondiciones		
Poscondiciones	-	
Flujo de eventos		
Flujo básico < Recuperar un estudio de imágenes >		
	<ol style="list-style-type: none"> 1. El Visor realiza la petición de recupera un estudio de imágenes. 2. El Servidor realiza la operación y devuelve el estudio especificado. 3. Termina el caso de uso. 	
Relaciones	CU Incluidos	
	CU Extendidos	No aplicable

Requisitos funcionales	no	No aplicable
Asuntos pendientes		No aplicable

9.3 CASO DE USO RECUPERAR UNA SERIE DE IMÁGENES

Caso de Uso	Recuperar una serie de imágenes	
Objetivo	Su objetivo es lograr cargar una serie completa de imágenes.	
Actores	Visor Web(Inicia)	
Resumen	El caso de uso inicia cuando el Visor realiza la petición al servidor sobre la serie que desea cargar, este realiza la operación definida por el Visor.	
Complejidad	Alta	
Prioridad	Crítico	
Referencias	RF 1.2	
Precondiciones		
Poscondiciones	-	
Flujo de eventos		
Flujo básico < Recuperar un estudio de imágenes >		
	<ol style="list-style-type: none"> 1. El Visor realiza la petición de recupera una serie de imágenes. 2. El Servidor realiza la operación y devuelve la serie especificada. 3. Termina el caso de uso. 	
Relaciones	CU Incluidos	
	CU Extendidos	No aplicable

Requisitos funcionales	no	No aplicable
Asuntos pendientes		No aplicable

9.4 CASO DE USO EXPORTAR IMÁGENES DICOM A OTROS FORMATOS

Caso de Uso	Exportar imágenes DICOM a otros formatos.	
Objetivo	Su objetivo es guardar imágenes DICOM en otros formatos compatibles.	
Actores	Visor Web(Inicia)	
Resumen	El caso de uso inicia cuando el Visor realiza la petición al servidor sobre la serie que desea guardar y el formato en que desea guardarla, este realiza la operación definida por el Visor.	
Complejidad	Alta	
Prioridad	Crítico	
Referencias	RF 1.3	
Precondiciones		
Poscondiciones	-	
Flujo de eventos		
Flujo básico < Exportar imágenes DICOM a otros formatos>		
	<ol style="list-style-type: none"> 1. El Visor realiza la petición de guardar una imagen en el formato deseado. 2. El Servidor realiza la operación. 3. Termina el caso de uso. 	
Relaciones	CU Incluidos	

	CU Extendidos	No aplicable
Requisitos funcionales	no	No aplicable
Asuntos pendientes		No aplicable

10 GLOSARIO DE TÉRMINOS