

Universidad de las Ciencias Informáticas

Facultad 7



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

**Comunicación entre los sistemas sanitarios en un entorno
hospitalario en el área de radiología**

Autores: Susana Hernández Claro

José Enrique Valdés López

Tutores: MSc. Reinier Alonso González

Ing. Raymundo Rodríguez García

La Habana, 2011

“Año 53 de la Revolución”

"El mundo camina hacia la era electrónica... Todo indica que esta ciencia se constituirá en algo así como una medida del desarrollo; quien la domine será un país de vanguardia. Vamos a volcar nuestros esfuerzos en este sentido con audacia revolucionaria"

Ernesto Che Guevara.



Tutor: MSc. Reinier Alonso González

Graduado de Ingeniería en Ciencias Informáticas en la Universidad de Ciencias Informáticas (UCI), año 2007 con categoría de Instructor. Ha impartido las asignaturas de Introducción a la programación, Programación I y Arquitectura de Software para la Salud.

Correo electrónico: ralonso@uci.cu

Tutor: Ing. Raymundo Rodríguez García

Profesor en adiestramiento. Graduado de Ingeniero en Ciencias Informáticas, egresado de la UCI en el 2010. Se desempeña actualmente como Administrador de la configuración del departamento de Tecnologías, Integración y Estándares del Centro de Informática Médica (CESIM), en la UCI.

Correo electrónico: rrdiguez@uci.cu



AGRADECIMIENTOS

*A la Universidad de las Ciencias Informáticas por hacernos participe de este proyecto de la revolución, así como al multidisciplinario claustro de profesores que lo integran.
A nuestros tutores: Raymundo Rodríguez García y Reinier Alonso González por su paciencia y colaboración.*

A todas las personas que han estado a mi lado en el transcurso de estos cinco años, los profesores que me han hecho una mejor persona y mis amistades, que hemos compartido momentos malos y buenos, alegrías y tristezas, pero al final siempre hemos estado juntos.

A mi familia, que los quiero mucho y siempre han estado ahí para mí, mis primas y primos que somos más que eso y nos queremos cantidad.

A mi familia postiza que siempre van a prevalecer en mi corazón porque me han dado un lugar en el suyo, mamá Martha, papá Angelito, Angélica y Yasneidy.

A mi amigo Marquito que más que amigo es un hermano, porque siempre ha estado ahí para mí, en cualquier ocasión siempre está dispuesto a todo para complacerme, te quiero.

A mi abuelita del alma Gina que aunque no se encuentra conmigo físicamente sé que siempre me está cuidando y dando todo su apoyo.

A mi hermano por ser un ejemplo de esfuerzo y dedicación y enseñarme que para lograr algo solo tienes que proponértelo.

A mis padres por ser las personas que me dieron la vida y pusieron todo su empeño para que pudiera vivirla al máximo.

A mi papá por ser el mejor hombre del mundo, y brindarme todo su amor, cariño y dedicación.

Y muy especialmente a la persona que ha estado conmigo en todo momento, la persona que cuando he reído ha reído conmigo, cuando he llorado ha llorado conmigo, cuando he necesitado de ella siempre ha

estado ahí para mí, aunque no se lo diga, ella lo adivina, a la persona que le debo cada pedazo de mi ser, mi vida entera, mi Madre.

Susana

A mi madre por sus eternos consejos y su infinita paciencia, por ser especial y la más linda del mundo.

*A mis abuelos, mi hermana, mi tío, mis primos, garrote, mi padre y a toda mi familia.
A Tony, Luna, Carlos Ernesto, Ronny, Pardo, el Nano, por ser más que mis amigos, por ser mis hermanos.*

*A mis compañeros de aula, de apartamento y de incontables historias a lo largo de estos 5 años.
A Ray por su infinita paciencia y por estar siempre disponible para la realización de este trabajo.
A todos los que de una manera u otra han influido en mi formación profesional.*

José Enrique

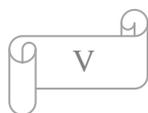
DEDICATORIA

Le dedico esta tesis y todo lo que soy a mi mami, quien ha sido mi soporte, guía y ejemplo. Por ser incondicional conmigo, por aguantarme todas mis malcriadeces, por darme las fuerzas cuando ya no tenía, por enseñarme que siempre se puede. Por ser la mejor madre del mundo y lo que más amo.

Susana

A mi mamá, por su cariño, su confianza, su apoyo, por ser siempre mi guía, sin ella esto no hubiera sido posible, a mis abuelos, por sus enseñanzas, consejos y su eterno amor, por formarme en el hombre que soy, a mi hermana por ser única, por estar siempre cuando la necesito, constantemente ha sido un ejemplo para mí, a mi linda sobrina por su eterna sonrisa. A mi tío que es mi padre, por todo el apoyo que me ha dado toda la vida, a Garrote por sus eternos consejos para la vida y correr siempre a mi lado en los momentos más difíciles. A toda mi familia, mis primos, más que primos son mis hermanos. A mi hermosa novia, por todo el apoyo que me ha dado, por estar conmigo en las buenas y en las malas durante todos estos años, por ser especial.

José Enrique



RESUMEN

En aras de informatizar los procesos en los departamentos de radiología se ha desarrollado el Sistema de Información Radiológica (RIS), el cual debe coexistir con los sistemas que facilitan, independientemente, la gestión de la información hospitalaria (HIS) y el trabajo con imágenes médicas digitales (PACS) en un entorno hospitalario. Cualquier servicio radiológico que desee mejorar su flujo de trabajo, optimizando tiempo, recursos y disponibilidad de la imagen adquirida, necesitará un HIS/RIS /PACS integrado.

El presente trabajo tiene como objetivo desarrollar una aplicación para la integración entre tres de los principales servicios necesarios para la gestión de la información médica de los pacientes. Durante el mismo, se realizó un estudio del estado del arte de los estándares de integración de sistemas sanitarios, protocolos y librerías desarrolladas con este objetivo para este sector.

Para la correcta elaboración del sistema se utilizó como metodología RUP para guiar el trabajo del equipo de desarrollo. Entre las herramientas escogidas figuran para el entorno de desarrollo .Net Framework v3.5, como IDE Visual Studio Team System 2008, para el modelado UML, Enterprise Architect v7.5 y como lenguaje de programación C# v3.0. Se describirá el diseño del sistema y los procesos del negocio, así como las características y funcionalidades que tendrá el sistema. Se explicará además la arquitectura utilizada para el desarrollo de la aplicación y se describen temas relacionados con la implementación del sistema, como las estrategias y los estándares de codificación utilizados.

Palabras claves: Sistema de Información Radiológica, HIS, PACS.

ÍNDICE

INTRODUCCIÓN 1

Capítulo 1: Fundamentación Teórica 7

 1.1 Sistemas de Almacenamiento, Transmisión y Visualización de Imágenes Médicas (PACS). 7

 1.1.1 Componentes de un PACS 8

 1.2 Sistemas de Información Radiológica (RIS) 9

 1.3 Sistemas de Información Hospitalaria (HIS) 10

 1.4 Interacción entre los sistemas HIS-RIS-PACS 10

 1.5 Integración de Empresas Sanitarias (por sus siglas en inglés IHE) 11

 1.6 Estándar HL7. 12

 1.6.1 MLLP (Minimun Lower Layer Protocol)..... 14

 1.6.1.1 ACK (Acknowledgement en español acuse de recibo) en MLLP. 14

 1.7 Estado del Arte..... 15

 1.7.1 Internacional..... 15

 1.7.2 Nacional 19

 1.8 Metodología, Tecnologías y Herramientas a utilizar. 20

CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA, Analisis y diseño de la solución...... 24

 2.1 Objeto de estudio 24

 2.1.1 Definición del problema 24

 2.2 Propuesta del sistema 24

 2.3 Modelo de Dominio 24

 2.4 Especificación de requerimientos de software. 26

 2.4.1 Requerimientos funcionales. 27

 2.5 Requerimientos no funcionales 30

 2.5.1 Usabilidad 30

 2.5.2 Soporte 30

 2.5.3 Hardware 31

 2.5.4 Software 31

 2.6 Especificación de Casos de Uso 31

 2.6.1 Definición de los actores. 31

2.6.2 Diagrama de Paquete: Administración de Pacientes (ADT).....	32
2.6.3 Diagrama de Paquete: Aceptación (ACK).	32
2.6.4 Diagrama de paquete: Programación de citas o estudios no solicitados. (SIU)	33
2.6.5 Diagrama de Paquete: Programación de citas o estudios y sus respuestas. (SRM/SRR)	34
2.6.6 Diagrama de Paquete: Solicitud de programación de citas o estudios y sus respuestas. (SQM/SQR).....	34
2.6.7 Diagrama de Paquete: Resultados. (ORU).....	35
2.6.8 Diagrama de Paquete: Protocolo MLLP.	35
2.6.9 Descripción Textual de los Casos de Uso.	35
2.7 Arquitectura en Capas:.....	37
2.8 Arquitectura Orientada a Servicios (SOA):	37
2.9 Patrones GRASP	38
2.10 Clases del Diseño	38
2.10.1 Descripción de las clases del diseño	38
2.11 Diagramas de clases del diseño.....	43
2.12 Diagramas de Secuencia	43
CAPITULO 3: IMPLEMENTACIÓN Y PRUEBA.....	44
3.1 Modelo de implementación.....	44
3.1.1 Diagrama de despliegue.....	44
3.1.2 Diagrama de componentes.....	45
3.2 Tratamiento de errores	46
3.3 Estrategias de codificación. Estándares y estilos a utilizar.	47
CONCLUSIONES	51
RECOMENDACIONES.....	52
REFERENCIAS BIBLIOGRÁFICAS.....	53
BIBLIOGRAFÍA.....	57
GLOSARIO DE TÉRMINOS	60

INTRODUCCIÓN

Los avances en las tecnologías de la información y las comunicaciones (TIC) vienen ocupando un lugar clave en el desarrollo de la sociedad. Mundialmente las Ciencias Informáticas han representado un papel esencial en esto, gracias a ella se evidencian grandes logros en la mejora de las distintas esferas de la vida diaria. Uno de los campos de la actividad humana que se ha visto beneficiado por los avances en las tecnologías es el área de la salud.

En la medicina se cuenta con diversas ramas para el tratamiento específico de las enfermedades, cada una de ellas ha sido sofisticada a medida que el proceso de informatización de la salud va creciendo. El área de imagenología es de gran importancia en cuanto al diagnóstico de cualquier enfermedad, puesto que es mucho más fácil descubrir que es lo que le afecta al paciente si se obtiene una imagen que describa con claridad los problemas que padece. Para esto se desarrollan los Sistemas de Almacenamiento y Transmisión de Imágenes (por sus siglas en inglés PACS).

Los PACS se encargan del almacenamiento, transmisión y visualización de las imágenes médicas digitales, así contribuyen al proceso de diagnóstico para los pacientes en el área de imagenología. Esta tecnología está conformada por equipos de adquisición de imágenes, estaciones de visualización, servidores de almacenamiento y la infraestructura de red para comunicar las distintas partes. Además, se encuentran los Sistemas de Información Radiológica (por sus siglas en inglés RIS) que comprende los procesos administrativos del departamento de radiología, incluyen gestión de citas, admisión y alta de pacientes, reportes de diagnósticos, ubicación y seguimiento de las placas radiográficas e información general de los pacientes de radiología. (1)

Actualmente en las organizaciones existe la generación masiva de información médica, cada día los procesos de registro, seguimiento y tratamiento del paciente deben mejorarse, innovarse y apoyarse en tecnologías para hacer más eficiente y eficaz las actividades rutinarias del hospital, centro de salud o clínica. Así surgen los Sistema de Información Hospitalaria (por sus siglas en inglés HIS) el cual es un sistema de información orientado a satisfacer las necesidades de generación de información, para almacenar, procesar y reinterpretar datos médicos y administrativos de cualquier institución hospitalaria. (2)

Con el desarrollo de estos sistemas se hizo necesaria la estandarización de la información para lograr una mejor fluidez en cuanto al manejo de la misma. Para esto aparecieron varios estándares como HL7 (Health Level Seven), DICOM (Digital Imaging and Communication in Medicine), entre otros, los cuales facilitaron la comunicación de forma efectiva.

DICOM 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. Por la necesidad de estándares de interoperabilidad que mejoraran la atención en la salud surge HL7. Se desarrolla con el objetivo de optimizar el flujo de trabajo, reducir la ambigüedad y mejorar la transferencia de conocimientos entre los prestadores de servicios de salud, organismos gubernamentales, la comunidad de proveedores, compañeros SDO acrónimo inglés de Standards Development Organization (Organización de Desarrollo de Estándares) y los pacientes. HL7 no es más que un conjunto de estándares para el intercambio electrónico de información clínica entre distintos tipos de sistemas de información médica independiente. (3).

La iniciativa I.H.E., acrónimo inglés de Integrating the Healthcare Enterprise (Integración de la Empresa/Organización de salud) nace como esfuerzo común de la R.S.N.A. (Radiological Society of North America) y la H.I.M.S.S. (Healthcare Information and Management Systems Society), a la que se unió posteriormente el A.C.C. (American College of Cardiology). Dicho proyecto tiene la intención de dar respuesta a las dificultades que tienen los diferentes sistemas de información clínico-asistenciales para comunicarse entre sí, aun cuando estos sean conformes a estándares de la categoría de DICOM y HL7. Surge para optimizar la eficiencia de estos sistemas, que necesitan comunicarse de tal forma que los usuarios finales tengan la información que necesitan para tomar decisiones cuando y donde lo necesiten. IHE no es un estándar como tal, sino que brinda soluciones en estándares ya existentes. (4)

Durante los últimos años un grupo de instituciones cubanas y el propio Ministerio de Salud Pública (MINSAP) han desarrollado sistemas encaminados a lograr la informatización de los servicios de la salud. En todos los casos el objetivo ha sido proveer al Sistema Nacional de Salud de información confiable, consistente y oportuna para la toma de decisiones y el mejoramiento de los procesos médicos asistenciales, garantizan de esta manera el incremento en la calidad y seguridad de la atención médica a

la población. Para la realización de esto se han desarrollado varios sistemas que facilitan el tratamiento con la información médica del paciente.

Los hospitales cuentan con equipamiento y médicos en sus diferentes áreas siendo el departamento de imagenología uno de los que se han beneficiado con el proceso de informatización, pues se cuenta con diversos equipos que brindan un diagnóstico más detallado del estado de salud, tales como resonancias magnéticas, tomografía, densitómetros óseos, entre muchos otros.

La Universidad de las Ciencias Informáticas (UCI) ha desempeñado un papel importante en cuanto a la informatización de la salud, debido a que ha desarrollado diversas aplicaciones como apoyo al desarrollo de la misma y la sofisticación de sus medios. Ejemplo de ello es el desarrollo del sistema alas PACS el cual permite el almacenamiento, transmisión y visualización de las imágenes médicas digitales, el sistema alas HIS que es el encargado de gestionar toda la información hospitalaria y alas RIS que comprende la parte administrativa del departamento de radiología en las instituciones sanitarias.

Estos sistemas se encuentran instalados actualmente en varios hospitales de Cuba y en la República Bolivariana de Venezuela. Las características de su desarrollo actual traen como consecuencia que el intercambio de información resulte incompatible, debido a que no existe una correcta integración entre cada uno de ellos. La integración que existe entre los sistemas alas PACS y alas RIS no implementa estándares internacionales como HL7. En ocasiones la comunicación entre los softwares instalados en los hospitales no es eficiente puesto que el mensaje que debe llegar no es bien recibido y esto proporciona que el diagnóstico y la correcta evaluación del paciente estén dañados, trayendo consigo una serie de consecuencias. Siendo alguna de ellas:

- Pérdidas de la información.
- No es entendido por parte de otro sistema instalado en la misma institución, los mensajes de un sistema con la información de un paciente, debido a que no existe una correcta integración de los mismos.
- Es imposible vender el sistema alas RIS como un producto independiente, debido a su naturaleza de complementar al sistema alas PACS, lo cual representa una desventaja desde el punto de vista comercial.

- Los sistemas alas RIS y alas PACS tienen fuertes dependencias, por esta razón impiden que se puedan integrar con otras soluciones

Es de vital importancia que entre los sistemas que coexisten en un entorno hospitalario exista una comunicación eficiente y segura para lograr una mejor atención al paciente y un aprovechamiento óptimo de los recursos que aquí se emplean.

Luego de analizar la situación antes expuesta queda definido el siguiente **problema a resolver**: ¿Cómo mejorar la comunicación entre los sistemas PACS, RIS, HIS utilizando los perfiles de integración IHE del dominio de Radiología?

Con vistas a dar solución al problema planteado se definió como **objeto de estudio**: La comunicación entre los sistemas sanitarios mediante servicios web; a raíz de lo cual el **campo de acción** se enmarca en la comunicación en el entorno de radiología que permita la integración de los sistemas PACS, HIS y RIS.

Siendo el **objetivo general** de la investigación mejorar la comunicación entre los sistemas PACS, RIS y HIS utilizando los perfiles de integración de IHE del dominio de radiología.

Para lograr este objetivo se plantean los siguientes **objetivos específicos**:

- Evaluar los perfiles de integración de IHE posibles a utilizar
- Diseñar los diferentes modelos para la solución de la integración.
- Implementar servicio web que logre la comunicación mediante mensajes entre los diferentes sistemas sanitarios.

Para dar cumplimiento al objetivo general, se plantean las siguientes **tareas de investigación**:

1. Realizar un análisis crítico y valorativo de las tendencias, técnicas, tecnologías, plataformas, librerías, metodologías y herramientas usadas en la actualidad en el sector sanitario.
 - 1.1. Estudiar interoperabilidad y servicios web desarrollados.
 - 1.2. Evaluar las tendencias en cuanto al desarrollo de servicios web para la comunicación entre sistemas dentro de instituciones hospitalarias.
 - 1.3. Evaluar las tendencias en cuanto al desarrollo de software libre

- 1.4. Estudiar los perfiles de integración.
- 1.5. Definir los distintos perfiles a utilizar con sus actores y transacciones.
- 1.6. Definir que capítulos de los estándares DICOM y HL7 serán utilizados en la solución propuesta.
2. Generar los artefactos correspondientes a los flujos de trabajo propuestos por la metodología seleccionada, sirviendo de base a los desarrolladores.
3. Analizar los procesos de negocio asociados a la integración de los sistemas alas HIS, alas PACS y alas RIS en el área de Radiología logrando un modelo único como guía para la implementación del sistema.
4. Implementar el sistema informático siguiendo lo establecido en la Especificación de Requisitos de Software.
 - 4.1. Identificar los requisitos de software para el desarrollo de la solución propuesta.
 - 4.2. Definir una arquitectura para la solución propuesta.
 - 4.3. Diseñar la solución propuesta.
 - 4.4. Implementar un servicio web para la creación de los flujos de mensajes.
 - 4.5. Implementar las transacciones priorizadas.

Beneficios

La comunicación entre los sistemas sanitarios mediante un servicio web que se encargará de proveer funcionalidades para la generación y envío de mensajes, cumpliendo estos con lo establecido por el estándar HL7 v2.3.1. Cualquier servicio radiológico que desee mejorar su flujo de trabajo, optimizando tiempo, recursos y disponibilidad de la imagen adquirida, necesitará un HIS/RIS /PACS integrado.

El documento consta con una estructura en capítulos que muestra todos los detalles de la investigación, así como también una descripción de los pasos seguidos para la realización del servicio web que integra los sistemas alas PACS, alas HIS y alas RIS.

➤ Capítulo 1

Fundamentación teórica: Se realiza el estudio teórico de la investigación. Muestra el estado del arte de los estándares de integración de sistemas sanitarios más utilizados en la actualidad, y se

profundiza en el estudio acerca de protocolos y librerías desarrolladas para el sector sanitario. Se estudian las principales tendencias, soluciones y experiencias en la integración.

➤ **Capítulo 2**

Características del sistema, análisis y diseño de la solución: Se hace referencia a las tecnologías necesarias que se utilizarán para la correcta elaboración del sistema. Se describen las herramientas empleadas para el desarrollo del servicio web. Describe la construcción del componente, a través del diseño, enfocado a cómo el sistema cumple sus objetivos, al tener en cuenta los requisitos funcionales y no funcionales. Se representan los procesos del negocio mediante un modelo de dominio, así como las características y funcionalidades que tendrá el sistema a partir de los requerimientos funcionales y no funcionales del software a desarrollar. Se explica además la arquitectura utilizada, así como el empleo de patrones de diseño.

➤ **Capítulo 3**

Implementación y prueba: Se describen temas relacionados con la implementación del sistema, como son los diferentes componentes que lo conforman y su distribución física. Se muestran las estrategias y los estándares de codificación utilizados.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

El presente capítulo contiene los resultados del estudio de estándares como HL7, además del protocolo MLLP (Minimum Lower Layer Protocol) perteneciente a este estándar. Se exponen los diversos sistemas de información médica existentes en la actualidad y cómo están integrados. Contiene los resultados del análisis de las principales herramientas y tecnologías actuales usadas para el desarrollo del sistema, así como la metodología de desarrollo.

1.1 Sistemas de Almacenamiento, Transmisión y Visualización de Imágenes Médicas (PACS).

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, permiten proveer servicios de almacenamiento, tratamiento y transferencia de información, para dar soporte a las áreas donde se genera un volumen importante de imágenes. (5)

Un PACS incluye entre sus componentes dispositivos para la adquisición de imágenes, unidades de almacenamiento, estaciones de visualización, computadoras y bases de datos. Estos componentes están integrados a través de una red de comunicación, imponen a los productores de software nuevos retos en temas de velocidad de transmisión de la información y capacidad de almacenamiento, véase figura 1.

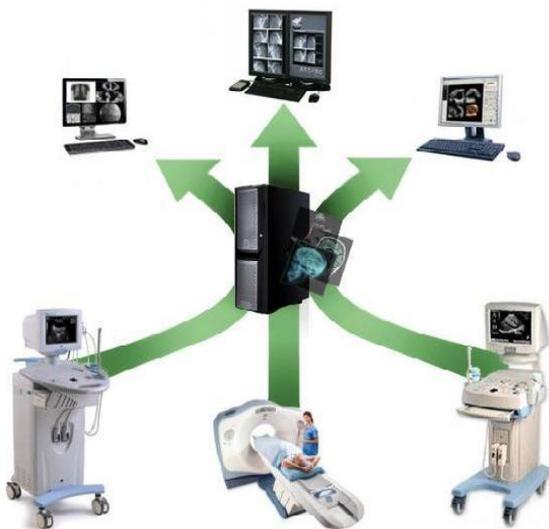


Figura 1. Componentes de un PACS

La tarea fundamental de un PACS es mantener el flujo de las imágenes adquiridas en las diferentes modalidades. El PACS es el responsable de la adquisición de imágenes con alta resolución, vela por tener una gran capacidad de almacenamiento para dar soporte a la actividad imagenológica. Otras capacidades sería la posibilidad de centralizar el almacenamiento y diagnosticar en cualquier computadora que se encuentre conectada a la red de imágenes. En esta se pueden observar estudios de distintas modalidades, siempre y cuando posea un visor de imágenes

médicas. (6)

Estos sistemas permiten que los especialistas no tengan que estar presentes en la captura de imágenes y estas puedan ser vistas desde cualquier estación de trabajo del hospital para su diagnóstico, ya sea individual o simultáneamente, gracias a que estas imágenes se almacenan en un servidor central y están disponibles en todo momento.

1.1.1 Componentes de un PACS

Sistemas de Adquisición: Equipos que se encargan de la obtención de imágenes médicas. Se agrupan en dos modalidades generales atendiendo a como generan las imágenes, que son imágenes digitales o analógicas. En el primer caso aparecen los tomógrafos y los resonadores, así como en el segundo caso los equipos convencionales de ultrasonido y endoscopia; estos últimos requieren posteriormente de digitalizadores. (7)

Servidor de Imágenes: Es un componente importante del PACS, por el gran volumen de información que deben admitir y sobre todo por la necesidad de acceso a las imágenes almacenadas, cada vez más frecuente (estudios multimodales y estudios seriados). Es el encargado de manejar las imágenes y su información asociada de manera eficiente. Incluye mecanismos de seguridad, automatización de tareas, administración de recursos y creación de copias de respaldo. Es conveniente que estén dotados de una estructura jerárquica, con dos niveles de acceso:

- Acceso inmediato (u online).
- Archivo histórico o a largo plazo, masivo y lento que debe mantenerse durante el plazo de tiempo legal (5-7 años) (8)

Estaciones de visualización: Cuenta con equipos especializados de visualización de imágenes, que cumplen características específicas para mejorar su evaluación, y facilitar un mejor diagnóstico. Los equipos de visualización son de alta resolución, deben contar con software especializado para el tratamiento y manipulación de las imágenes, y dar un espacio para poder guardar las evaluaciones y análisis dados por los médicos. El PACS, a su vez, debe ser capaz de transmitir con facilidad las imágenes a múltiples monitores para su manejo y control. (9)

Infraestructura de red: Este componente conecta las partes mencionadas anteriormente a través de una red LAN o WAN y los dispositivos utilizados, proporcionan así una eficaz comunicación entre ellos. Es imprescindible para el correcto funcionamiento de los PACS.

1.2 Sistemas de Información Radiológica (RIS)

Debido a la complejidad que encierra el flujo de trabajo de un departamento de radiología y con el objetivo de maximizar las posibilidades que brindan los PACS, los llamados Sistemas de Información Radiológica (RIS), creados inicialmente como Sistemas de Información en Salud especializados en la gestión de la información de los departamentos de radiología, se convirtieron en el complemento por excelencia de los sistemas PACS.

Un RIS es un sistema informático diseñado para soportar el flujo operacional y el análisis del negocio dentro del departamento imagenológico. Tiene la responsabilidad de gestionar la actividad clínica y administrativa del departamento, manejar la información demográfica de los pacientes, programar las citas y la entrega de reportes de diagnóstico, entre otras funciones.

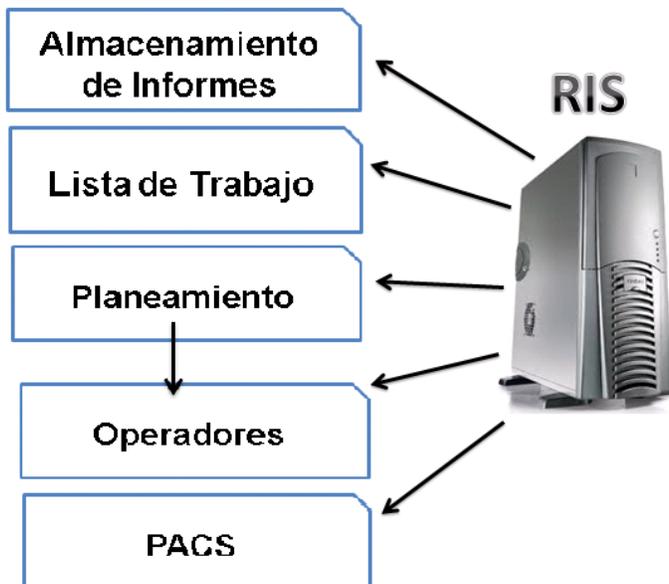


Figura 2. RIS

Entre sistemas RIS y PACS debe establecerse una comunicación bidireccional en la que cada sistema interactúe y se apropie de información. Para esto fueron definidos los estándares HL7 para la integración de los sistemas de salud y DICOM para soluciones imagenológicas. La integración de los sistemas PACS-RIS con soluciones de información hospitalarias posibilita un incremento de la calidad en la atención al paciente. Esto viene determinado porque el especialista a la hora de tomar decisiones tendría a su disposición además de la información radiológica del paciente, toda aquella generada en otras áreas de la institución,

véase figura 2. (10)

1.3 Sistemas de Información Hospitalaria (HIS)

En el dominio de la informática un sistema de información es una aplicación utilizada para obtener, almacenar, procesar, administrar y transmitir datos. El mismo procesa los datos obtenidos y genera nuevos datos e información para satisfacer las necesidades de gestión y control de un dominio específico. En los hospitales estos sistemas se nombran Sistemas de Información Hospitalaria (HIS) y con su implantación mejoran el flujo de la información y los servicios en los centros hospitalarios.

Un HIS consta de varios módulos especializados que informatizan las actividades de captura y procesamiento de los datos en cierta área del hospital. El Sistema de Información Hospitalaria al igual que el HIS tiene como objetivo la informatización de los procesos intrahospitalarios. Su arquitectura es flexible y robusta, basada en tecnologías maduras y estándares para el desarrollo de aplicaciones web. Fue desarrollado al utilizar herramientas con soporte a la administración de procesos del negocio (BPM), informatiza el flujo normal de la información en cualquier hospital, pero de haber cambios en el flujo de algún proceso el cambio es fácilmente configurable. (11)

1.4 Interacción entre los sistemas HIS-RIS-PACS

La interacción con el RIS es fundamental para el mejor aprovechamiento de las capacidades del PACS. El RIS proporcionará al PACS toda la información necesaria en el área de radiología, a su vez el PACS notificará al RIS que el estudio ha sido realizado y completado, para posteriormente proporcionar al radiólogo las imágenes de la exploración realizada, de forma que éste pueda elaborar el informe correspondiente en el RIS. Una vez finalizado este, el RIS envía una copia al PACS y la notificación de que el informe ha sido realizado.

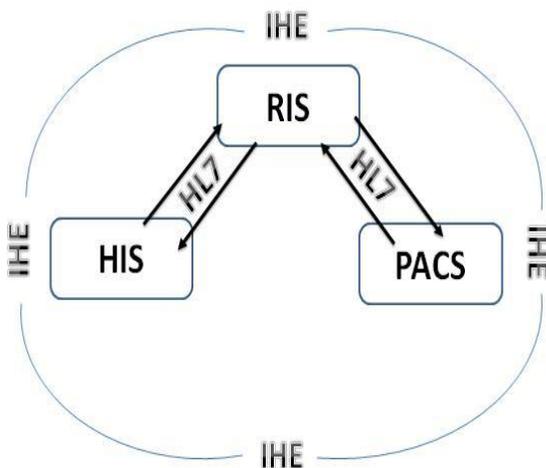


Fig 3. Integración HIS - RIS - PACS.

Un RIS es como la ficha del rompecabezas que integra las otras dentro de la gestión hospitalaria. Se comunica con el Sistema de Información Hospitalaria (HIS), y con todas las modalidades de exploración que están produciendo imágenes en formato DICOM, para enviarla a cada imagen información que la identifique, garantizando la unicidad del paciente y de los exámenes del mismo, véase figura 3.(12)

1.5 Integración de Empresas Sanitarias (por sus siglas en inglés IHE)

Integrating the Healthcare Enterprise, que se abrevia como IHE y que podría traducirse al castellano como Integrando las Empresas Sanitarias, es una iniciativa de profesionales de la sanidad y empresas proveedoras de equipos y software médicos, cuyo objetivo es mejorar la comunicación entre los sistemas de información que se utilizan en la atención al paciente. IHE proporciona una metodología práctica que asegura la interoperabilidad entre Sistemas de Información Sanitarios (13)

Las partes que intervienen en el proceso sanitario (tanto asistencial, gestor o planificador) “sufren” de sistemas de información computacionales que no comparten la información eficazmente. Como se refleja se hace necesario para ello el uso de lenguajes comunes o estándares de comunicación médica, para que dichos sistemas sean interoperables. Pero dichos estándares no garantizan la integración sin problemas, encontrándose los usuarios y fabricantes habitualmente con serias dificultades para integrar los diferentes sistemas de información.

La iniciativa IHE trata de establecer qué estándares son recomendables utilizar en una circunstancia específica, construyendo lo que se llaman los “Technical Frameworks”. Estos marcos de trabajo son un grupo detallado de documentos que guían a los desarrolladores de sistemas de información y los integradores para poder implementar las capacidades que se definen en la iniciativa. Bajo este, los profesionales médicos y la industria sanitaria llegan a un consenso de un modelo de integración que crezca anualmente, añadiendo progresivamente más y más transacciones entre un grupo mayor de sistemas (14).

IHE define Perfiles de Integración que utilizan estándares ya existentes para la integración de sistemas de manera que proporcionen una interoperabilidad efectiva y un flujo de trabajo eficiente. IHE permite alcanzar el nivel de integración exigible en la era de la historia clínica electrónica. Cada Perfil de Integración IHE describe una necesidad clínica de integración de sistemas y la solución para llevarla a cabo. Define los componentes funcionales, los cuales reciben el nombre de Actores IHE, y especifica con el mayor grado de detalle posible las transacciones que cada Actor deberá llevar a cabo, basadas siempre en estándares como el de Digital Imaging and Communication in Medicine (DICOM) y Health Level 7 (HL7). (15)

IHE hace que el uso de las tecnologías de la información avanzadas ayude en gran medida al personal sanitario a la hora de mejorar la calidad y eficiencia de la atención sanitaria, aumenta la seguridad del paciente al garantizar la integridad de la información médica, reduce el tiempo empleado en la solución de problemas tales como, la pérdida de datos y la aparición de estudios no correspondientes. De esta forma, se aprovecha el tiempo del personal, además de que proporciona al personal sanitario información bien estructurada sobre el paciente de modo que la toma de decisiones médicas se base en la mejor información posible.(16)

1.6 Estándar HL7.

HL7 es una organización fundada en 1987 para desarrollar estándares de intercambio electrónico de información clínica de diferentes tipos entre sistemas de informática médica independientes. Es un protocolo que define el formato de las "transacciones" entre diferentes componentes, de forma que dos sistemas completamente independientes puedan comunicarse entre sí, simplificando la integración de información entre sistemas médicos. Debe su nombre a que fue concebido como estándar para la capa 7 (Nivel de aplicaciones) del modelo OSI (Interconexión de Sistemas Abiertos), donde la unidad de información es el mensaje. Por ello es relativamente independiente del tipo de conexión física y protocolo de comunicación usado; se ocupa exclusivamente del proceso de dar formato a los datos para convertirlos en mensajes que cualquier aplicación que cumpla la norma, pueda entender.

Su uso como estándar de comunicaciones permite que logre una independencia de los sistemas en una plataforma de hardware o de software y facilitando la creación de interfaces entre cada sistema. Define transacciones para los distintos dominios del negocio y conforma un método para el intercambio de mensajes entre computadoras en el área de salud, por ejemplo: transmisión de órdenes de prestación, observaciones clínicas, datos clínicos (incluyendo resultados de estudios); admisión, transferencia o alta de pacientes; información demográfica o para facturación; reportes, consultas e incluso para mantenimiento de tablas maestras entre sistemas. En otras palabras, HL7 es un estándar para mensajes que facilita las comunicaciones entre aplicaciones en el campo de la salud (17).

El protocolo HL7 consta de una serie de mensajes que representan los datos, resultantes de la ocurrencia de un evento iniciador (trigger event) en el sistema emisor. Así, cuando ocurre un evento (ej.:

cuando un paciente es transferido desde una cama a otra) el programa aplicativo del sistema emisor produce una transacción HL7 que contiene el resultado de dicho evento disparador. (18)

A partir del año 2000, la organización HL7 cuenta con un proceso para definir una serie de herramientas de interoperabilidad (mensajes, documentos electrónicos, reglas, modelos de referencia), esto ha dado origen a varios estándares que facilitan los procesos de intercambio de información de salud. Debido a ello, hoy en día, se habla de Estándares HL7. Algunos de estos estándares son:

- Mensajería HL7 Versión 2: Estándar de mensajería para el intercambio electrónico de datos de salud.
- Mensajería HL7 Versión 3: Estándar de mensajería para el intercambio electrónico de datos de salud basada en el RIM (Reference Information Model).
- CDA HL7: (Clinical Document Architecture) Estándar de arquitectura de documentos clínicos electrónicos.
- SPL HL7: (Structured Product Labeling) Estándar electrónico de etiquetado de medicamentos.
- HL7 Medical Records: Estándar de administración de Registros Médicos.
- GELLO: Estándar para la expresión de reglas de soporte de decisiones clínicas.
- Arden Syntax: Es estándar sintáctico (if then) para compartir reglas de conocimiento clínico.
- CCOW: Es un estándar framework para compartir contexto entre aplicaciones.

El estándar de la versión 3 tiene la capacidad para apoyar los workflows del healthcare. El desarrollo de esta comenzó alrededor 1995, dando por resultado una publicación estándar inicial en 2005. En comparación con la versión 2, se basa en una metodología formal, el HDF (Entorno de desarrollo de aplicaciones en salud) y principios orientados al objeto. Como parte de la versión 3, el comité técnico del vocabulario HL7 desarrolló los métodos que permiten que las especificaciones HL7 dibujen sobre códigos y vocabularios de una variedad de fuentes. El trabajo del vocabulario V3 asegura que los sistemas que ponen las especificaciones en ejecución HL7 tienen una comprensión inequívoca de los

dominios de las fuentes del código y del valor de código que están utilizando. En más de 40 países se utiliza este importante estándar para la comunicación de los sistemas sanitarios. (19)

1.6.1 MLLP (Minimun Lower Layer Protocol)

El objetivo del protocolo MLLP es el de proveer una interface entre una aplicación HL7 y el protocolo. Esta característica, junto a su gran base implantada en el ámbito sanitario, han sido las condiciones por las que se ha escogido el protocolo. Los caracteres de control que fija el protocolo para transmitir un mensaje, son los siguientes: ver figura 4.

- Marca de inicio de mensaje: VT (hexadecimal 0x0B)
- Marca fin de mensaje: FS (hexadecimal 0x1C)
- Marca de separación: CR (hexadecimal 0x0D)

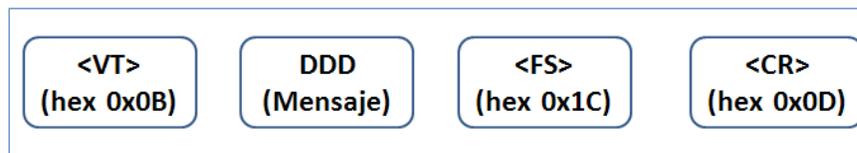


Figura 4. Estructura del mensaje utilizando MLLP

1.6.1.1 ACK (Acknowledgement en español acuse de recibo) en MLLP.

Dado que MLLP no es un protocolo que incluya ningún mecanismo de rechazo de mensaje o de notificación de aceptación, esta es necesario realizarla a través de un mensaje ACK de aceptación HL7. Todas las interacciones deberán ser contestadas por un ACK de este tipo. Las excepciones a esta norma son las consultas o solicitudes, que definen su propio ACK de contestación con una carga útil a nivel de aplicación.

El ACK-HL7 será contestado a través de la misma conexión (socket) de forma relativamente inmediata (con un timeout máximo de 5 segundos). La no recepción de un ACK en este intervalo se considerará como fallo de transmisión y por tanto ésta deberá intentarse más adelante. El ACK de HL7 permite indicar si el mensaje ha sido o no procesado correctamente. (20)

1.7 Estado del arte.

1.7.1 Internacional

A continuación se muestran algunas tendencias actuales de la integración de los sistemas sanitarios a nivel internacional de diferentes compañías.

ALMA3D

El producto Alma3D es un potente visor radiológico, especialmente diseñado para trabajar integrado en los sistemas de información de los centros médicos y radiológicos. Estos sistemas informáticos médicos, habitualmente incluyen, entre otros, los siguientes componentes: Sistema de información de gestión (RIS/HIS), modalidades radiológicas, servidor de imágenes PACS, visualizador de imágenes. (21)

La comunicación con los sistemas HIS/RIS se puede realizar por diferentes mecanismos, entre los que cabe destacar:

- Integración por mensajes XML.
- Integración por medio de la librería dinámica A3dConnect.dll.
- Integración mediante la aplicación A3dConnect.exe.
- Desarrollo a medida de componentes de software específicos.

La plataforma ALMA es una tecnología consolidada con más de 2400 licencias instaladas en el mercado. El software ha sido integrado con más de 10 fabricantes de PACS y HIS/RIS. Actualmente, la empresa se encuentra en un período de fuerte expansión a nivel mundial estableciendo acuerdos de colaboración con los principales líderes OEM (Original Equipment Manufacturer) y distribuidores que se han unido a Alma3D para ofrecer un valor real a sus clientes. (22)

Desde 1996, los socios fundadores de ALMA, crearon una sociedad con el objetivo de desarrollar un software innovador con herramientas para el diagnóstico de imágenes médicas, así como para la planificación, navegación y simulación quirúrgica. ALMA continúa diseñando tecnología de vanguardia y

soluciones de servicio impulsados por las necesidades reales de los usuarios y lograr niveles óptimos de productividad y eficacia. Cumple con las normas internacionales de la Industria tales como DICOM e IHE. (23)

SUEIIDISS

Es la Sociedad Uruguaya de Estandarización Intercambio e Integración de Datos e Información en Sistemas de Salud. Existen diferentes perfiles designados por IHE para resolver diferentes dominios de problemas, asimismo la SUEIIDISS maneja principalmente el estándar HL7 v3 y otros (ejemplo identificación de personas), con los cuales busca obtener un conjunto coherente y abarcador de recomendaciones para utilizar con éxito en sistemas de salud. Acompañando este desarrollo del conocimiento que se consolida en el seno de la sociedad, es que se seleccionan el conjunto de estándares a probar conjuntamente con tecnologías que permiten la integración a nivel real de los sistemas de los que se tiene conocimiento por parte de los posibles participantes. (24)

Para las pruebas anteriores algunos de los estándares seleccionados eran el estándar CDA release 2 de HL7 versión 3, como tipo de documento a intercambiar, en cuanto al tema de tecnologías se consolidó la utilización en las dos ediciones de Web Services como mecanismos de comunicación. Se realizaron pruebas utilizando certificados digitales y TLS en base a las recomendaciones que se tenía a nivel general para los estándares de comunicaciones de la propia SUEIIDISS, así como se recomienda en el perfil ATNA (Audit Trail and Node Authentication) uno de los tomados en cuenta para las pruebas. (25)

Quedo claro que la comunicación utilizando Web Services, era un estándar fácilmente adoptable por parte de los participantes dado que en todas las plataformas de desarrollo se consiguió su implementación sin dificultades. Como resultado más destacable: es el contar con una interface probada con diferentes instituciones para realizar intercambios de información, así como también el testeado de las implementaciones de los estándares (ej. CDA) y su posterior lectura por parte de otra institución. Algunas de las plataformas en las que se realizaron pruebas fueron .NET. (26)

Galileo

NoemaLife es un Grupo ítalo-alemán cuya misión es proveer soluciones software diseñadas para mejorar los procesos clínicos. Ha desarrollado un nuevo concepto en soluciones que le permite a las

organizaciones sanitarias afrontar exitosamente los cambios que actualmente están caracterizando al Sistema Sanitario. (27)

Galileo ha sido específicamente diseñado para asegurar una correcta integración entre los diferentes componentes y, al mismo tiempo, una arquitectura abierta, que permita una flexible y fácil integración con los sistemas de terceros. Este objetivo ha sido alcanzado gracias a una Service - Oriented Architecture (SOA) de última generación, un modelo de datos basado en el protocolo estándar HL7 y un poderoso sistema de mensajería y transcodificación de datos. La sencilla integración entre los diferentes componentes de Galileo es garantizada por la eHealth Application Platform, que provee un conjunto de componentes de infraestructura comunes a todas las aplicaciones. Tales componentes se ponen a disposición de las aplicaciones de Gaileo a través de una capa de Servicios Web y API's. (28)

La integración con sistemas de terceros se lleva a cabo mediante un potente gateway que permite la transcodificación y transmisión de los datos en los protocolos comúnmente utilizados: HL7, DICOM. Gracias a su potente capacidad de integración, Galileo ha sido implementado y testeado en el Connectathon en 16 diferentes perfiles IHE desde 2003. Galileo es nuevo, pero sin embargo ya es ampliamente utilizado por más de cien Hospitales de Europa y América Latina. (29)

Integración de los Sistemas HIS-PACS-RIS del Hospital Son Llätzer en España.

La integración de estos sistemas en el Hospital Son Llätzer se ha basado desde su apertura en el desarrollo de un lenguaje unificado entre HP (Hewlett Packard) y GE (General Electric) utilizando el estándar HL7 y siguiendo pautas en la actualidad de IHE. La mensajería HL7 entre los sistemas HIS de HP y RIS de G.E. es de 5.000 mensajes diarios y de 35.000 entre todos los sistemas. Dependiendo si el mensaje es para datos demográficos, citas o resultados se envían tipos diferentes con campos y segmentos distintos. Con la integración mediante HL7, han conseguido las peticiones informatizadas sin apoyo papel, la citación desde el HIS (hospital y atención primaria) de la totalidad de las pruebas radiológicas a tiempo real. (30)

La eliminación de todos los subprocesos manuales asociados al producto final como la gestión de impresión de placas, búsqueda de estudios previos, tiempo de informe y entrega de resultados con HIS/RIS/PACS integrados es lo que ha hecho que la efectividad aumente. Estos sistemas mantienen bases de datos unificadas y comparten la información entre los distintos sistemas aunque manteniendo la

independencia funcional de cada aplicación local, respetando todos los desarrollos existentes y futuros. (31)

InterSystems Corporation

InterSystems Corporation es el líder mundial en software para la Sanidad Conectada. Con su sede central en Cambridge. Esta corporación desarrollo la plataforma de integración InterSystems Ensemble®, entre sus múltiples funcionalidades se encuentra la interconexión entre aplicaciones departamentales. Asimismo, hace posible que establezca la interoperabilidad con organismos externos, como los Servicios Centrales del Instituto de Gestión Sanitaria, los Centros de Atención Primaria u otros Centros Sanitarios. (32)

El hospital universitario de Ceuta utiliza esta plataforma con el objetivo es implementar una Historia Clínica Unificada que permita, por un lado, consolidar toda la información asistencial asociada a cada paciente y, por otro, facilitar a los profesionales el uso de una sola herramienta de trabajo integrada. El proyecto se está desarrollando en dos fases: la primera centrada en que todas las aplicaciones gestionen los mismos datos demográficos de los pacientes y, la segunda, para la solicitud de peticiones de estudios, ordenes médicas y prescripciones, con la correspondiente recepción de resultados. (33)

La integración de los sistemas y aplicaciones del Hospital Universitario de Ceuta se efectuará mediante los estándares de intercambio de información clínica HL7 y DICOM, abriendo la posibilidad de que cualquier tipo de aplicación y sistema pueda ser integrado, ya sea clínicos (incluyendo la imagen clínica no radiológica) o económico - financieros y de gestión, sin que ello suponga esfuerzos extra en las implementaciones a efectuar. Las aplicaciones y sistemas que no cuenten con estos estándares dispondrán de adaptadores específicos, que permitirán su total integración en la plataforma. Esta infraestructura facilitará que se puedan realizar transformaciones entre mensajes HL7 de diferentes versiones, y entre formatos específicos de las aplicaciones que no dispongan de este estándar. (34)

Kodak Carestream RIS & Kodak Carestream PACS

KODAK CARESTREAM RIS está respaldada por la experiencia de Carestream Health, que se basa en más de 100 años ofreciendo tecnología de imágenes médicas, así como más de 100 instalaciones de RIS satisfactorias en todo el mundo. CARESTREAM RIS es el núcleo de información de la plataforma de

soluciones para radiología de KODAK CARESTREAM. Si se combina KODAK CARESTREAM RIS con KODAK CARESTREAM PACS, se obtiene un escritorio integrado destinado a mejorar la productividad y la eficacia del diagnóstico. Los radiólogos pueden consultar de forma inmediata el historial clínico completo de los pacientes para ofrecer una interpretación rápida y precisa de los exámenes. Esto es posible debido a que Carestream Health posee un motor único de integración basado en estándares como HL7 el cual posibilita que cualquiera de sus productos tanto el RIS como el PACS se integren entre ellos como con cualquier otro fabricante. (35)

Las imágenes se pueden mostrar automáticamente en los monitores de diagnóstico de la estación de trabajo, a lo que se une un monitor RIS adicional para ver la información textual del paciente. Mediante un clic se accede a un conjunto completo de herramientas, los datos de los exámenes, los datos clínicos (como los informes de laboratorio), los historiales de los pacientes y el sistema de dictado mediante voz. (36)

La solicitud de exámenes resulta rápida, flexible y precisa con este sistema. Los médicos que realizan peticiones pueden introducir los pedidos directamente en el sistema RIS a través de un HIS (sistema de información del hospital) HL-7 o mediante comunicaciones web seguras, reduciendo de este modo las tareas de introducción y las posibilidades de error. Los pedidos también se pueden introducir manualmente o escanear a partir de documentos en papel en la recepción de radiología. (37)

Estos productos ofrecen diferentes soluciones para los problemas de integración que existen en los sistemas sanitarios. En el mercado, el valor de las licencias para la utilización de estas plataformas de integración es muy elevado, lo cual constituye un problema para su obtención.

1.7.2 Nacional

Las aplicaciones para el trabajo con imágenes médicas no son algo nuevo en nuestro país, es destacable la existencia de sistemas como el PATRIS y el IMAGIS, desarrollados en la década de los 90 en los laboratorios de EISISOFT (Centro de Robótica y Software con sede en La Habana) y el Centro de Biofísica Médica de la Universidad de Oriente, respectivamente. El PATRIS, aplicación pionera en el campo de las imágenes médicas, fue utilizado en varios de los principales hospitales cubanos, pero quedó obsoleto al no poder responder al exigente desarrollo de la radiología. El IMAGIS está compuesto por tres

módulos: transmisión, almacenamiento y visualización, se encuentra instalado en varias instituciones hospitalarias del país.

Se debe mencionar el trabajo que realiza la Universidad de las Ciencias Informáticas que cuenta con una facultad dedicada al desarrollo de software para la salud. Existe además una línea de investigación dedicada al procesamiento digital de imágenes y señales con más de 6 años de experiencia en el desarrollo de software imagenológico (38). Entre los sistemas que se han desarrollado se encuentran: las PACS, las RIS y las HIS.

Existe una solución de integración entre los sistemas las RIS y las PACS, la misma se basa mediante consultas SQL y servicios web. En el primer caso, el sistema las RIS realiza consultas a la base de datos del servidor de imágenes del sistema las PACS con el objetivo de encontrar pacientes que previamente se les programó una cita, si los encuentra establece el estado de dicha cita como realizada. En el segundo caso el sistema las RIS expone un servicio web encargado de la gestión de peticiones provenientes del módulo Reportador del sistema las PACS. Estas peticiones pueden ser: admitir informe, actualizar lista de trabajo, autenticar, entre otros.

Según lo planteado anteriormente se puede llegar a la conclusión de que ambos sistemas tienen fuertes dependencias impidiendo que se puedan integrar con otras soluciones, dado que no implementan estándares internacionales tales como HL7. Además, es imposible vender el sistema las RIS como un producto aparte, debido a su naturaleza de complementar al sistema las PACS, lo cual representa una desventaja desde el punto de vista comercial.

1.8 Metodología, Tecnologías y Herramientas a utilizar.

El desarrollo de los equipos médicos y las tendencias actuales de evolución de los mismos están ligados a las Tecnologías de la Informática y las Comunicaciones (TICS), por lo que se hace necesario un estudio detallado de las metodologías, tecnologías y herramientas más usadas en el desarrollo de aplicaciones informáticas para la salud.

RUP

El Proceso Unificado de Rational, habitualmente resumido como RUP, es un proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado (UML), constituye la metodología estándar más

utilizada para el análisis, implementación y documentación de sistemas orientados a objetos a nivel mundial. Brinda una información detallada de las etapas, ciclos, flujos de trabajo y artefactos en el desarrollo de un software. Es un proceso iterativo e incremental. (Innova, Grupo Soluciones, 2007). (39)

El ciclo de vida RUP es una implementación del desarrollo en espiral. Fue creado ensamblando los elementos en secuencias semi-ordenadas. El ciclo de vida organiza las tareas en fases e iteraciones. Esta metodología divide el proceso en cuatro fases, dentro de las cuales se realizan varias iteraciones en número variable según el proyecto y en las que se hace un mayor o menor hincapié en las distintas actividades. (40)

Define qué, cuándo y quién debe realizar una tarea especificada. Organiza el desarrollo para poder hacer planificaciones y definir hitos que posibiliten controlar el progreso del proceso de desarrollo. (41)

.NET Framework 3.5

.Net Framework es un proyecto de la Microsoft para crear una infraestructura de desarrollo o plataforma sobre la que se reúne todo un conjunto de lenguajes y servicios que permiten el desarrollo de aplicaciones de una manera más rápida. Mediante esta herramienta se ofrece un entorno de ejecución altamente distribuido, que permite crear aplicaciones robustas y escalables. Esta nueva tecnología es una solución que se convierte en uno de los productos principales en la oferta de Microsoft, y pretende ser utilizada por la mayoría de las aplicaciones creadas para la plataforma Windows. (42)

C# 3.0

C# es un lenguaje de programación orientado a objetos que fue creado y estandarizado por Microsoft de manera independiente para desarrollar aplicaciones sobre la plataforma .NET. Logra capturar los mejores rasgos de los lenguajes de programación más populares que le precedieron, combinándolo en uno. Se valora el uso de este lenguaje debido a su gran flexibilidad, sencillez y seguridad. (43)

Además, permite trabajar tanto a bajo nivel con el uso de punteros como a un alto nivel, de manera que sea más descriptivo y cercano al lenguaje natural, permite acceder a las características avanzadas de la plataforma sobre la que se trabajará, crear código muy eficiente en aquellos puntos de la aplicación que son críticos y acceder a las interfaces de programación de aplicaciones existentes. Este lenguaje de

programación surge en el año 2000 con su primera versión (1.0), seguida de las versiones 1.1 en el 2003, 2.0 en el año 2005, 3.0 en el 2007, 3.5 en el año 2008 y la versión 4.0 en el año 2010. (44)

Visual Studio Team System 2008

Visual Studio es un Entorno de Desarrollo Integrado –IDE, por sus siglas en inglés– desarrollado por Microsoft. Soporta varios lenguajes de programación tales como Visual C++, Visual C#, Visual J# y Visual Basic.NET, así como tecnologías, dentro de las que se encuentran ASP.NET, Windows Presentation Foundation (WPF), Windows Communication Foundation (WCF) y Windows Workflow Foundation (WWF). Este permite a los desarrolladores crear aplicaciones de escritorio, aplicaciones web, así como servicios web en cualquier entorno que soporte la plataforma .NET. (45)

Visual Studio Team System es una variante de este popular IDE, está dirigida a grupos de desarrolladores, programadores y arquitectos de sistemas que trabajen de forma conjunta en un mismo entorno. Por ello, la principal diferencia con las otras versiones del producto radica en que las herramientas de colaboración, tales como las dedicadas a la gestión de la configuración, modelado, pruebas, entre otras, ofrecen un elevado nivel de integración. (46)

Enterprise Architect 7.5

Enterprise Architect (EA) es una herramienta para modelar y gestionar información compleja, diseñar y visualizar software, o construir y desplegar diversos sistemas, es 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. EA abarca el ciclo de vida completo del desarrollo de software. (47)

La edición corporativa está pensada para grandes equipos de desarrollo. Soporta todas las funcionalidades de las versiones Desktop y Professional, así como también la habilidad de conectarse a SQL Server, MySQL, Oracle 9i and 10g, PostgreSQL, MSDE, Adaptive Server Anywhere y MS Access backends como repositorio compartido.

En este capítulo se realizó el estudio de estándares como DICOM, HL7 y los perfiles de integración IHE, posibilita así lograr un mejor entendimiento de las soluciones factibles que ofrecen para la integración entre softwares sanitarios. Se estudiaron algunas plataformas y motores para la integración de los sistemas de información clínicos-asistenciales a nivel nacional e internacional quedando demostrado que

los mismos presentan varias dificultades. Según las restricciones de la aplicación y la política tecnológica de la universidad, se definieron las herramientas, lenguaje de programación y tecnologías a utilizar en la implementación.

CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA, ANÁLISIS Y DISEÑO DE LA SOLUCIÓN.

En el presente capítulo se describe la problemática existente, así como las características del componente a desarrollar. Debido a la poca definición de los procesos del negocio y la ausencia de un cliente se manifiestan los principales conceptos tratados y las relaciones existentes entre ellos mediante un Modelo de Dominio o Modelo Conceptual. Además, se plantean los requerimientos que el sistema debe cumplir, los casos de uso y los actores del mismo.

2.1 Objeto de estudio

2.1.1 Definición del problema

Actualmente se han implantado tres sistemas en algunos hospitales en el área de radiología, siendo estos; alas HIS, alas RIS y alas PACS. Cada uno tiene una funcionalidad diferente pero la finalidad de todos ellos se reduce en: mejorar la organización del centro y la atención al paciente.

2.2 Propuesta del sistema

Se propone realizar un servicio web para la comunicación eficiente y segura entre los sistemas alas HIS, alas RIS y alas PACS, basándose en los lineamientos IHE y los estándares DICOM y HL7, esto permitirá el envío de mensajes entre sistema cuando se admite un paciente y todo lo que conlleva la admisión del mismo, la notificación de mensajes para la programación de citas o estudios y sus respuestas, notificación de mensajes de aceptación entre otros.

2.3 Modelo de Dominio

Cuando en el entorno del negocio los procesos tienen un bajo nivel de estructuración y no existen flujos de información interconectados y bien definidos, se genera la imposibilidad de llevar a cabo un Modelo de Negocio, haciéndose necesario la realización de un Modelo de Dominio. El Modelo de Dominio, permite mostrar los principales conceptos que se manejan para el dominio del sistema en desarrollo. El diagrama del Modelo de Dominio se presenta en forma de diagrama de clases donde figuran los principales conceptos y roles del sistema analizado. (48)

Conceptos Fundamentales:

Cliente: Establecer la conexión con otras aplicaciones a través de la red y negociar el envío de los mensajes pertinentes (enviar el mensaje y recibir la respectiva respuesta). Su principal objetivo es que el mensaje llegue a su destino con la estructura requerida.

Servidor: Recibir las conexiones de otras entidades de aplicación y los respectivos mensajes que estas envían. Su principal objetivo es el de verificar la correcta estructura del mensaje y que el mismo pase a la capa superior.

Controladora de Capa: Controla el funcionamiento dentro de la capa. Dirige las funciones de las clases servidor y cliente según el momento en que se requiera su utilización. Dirige las peticiones provenientes de la capa superior.

Generadora de mensajes: Se encargaría de generar el mensaje pertinente según el evento disparador y la información que se quiera transmitir. Esta información sería proporcionada por el sistema que consume la librería o una capa superior encargada de esta negociación con los sistemas.

Mensaje: Clase básica a partir de la cual heredarían cada mensaje. Puede también ser una interfaz para obligar a que todos los tipos de mensajes implementasen las mismas características de ser necesario.

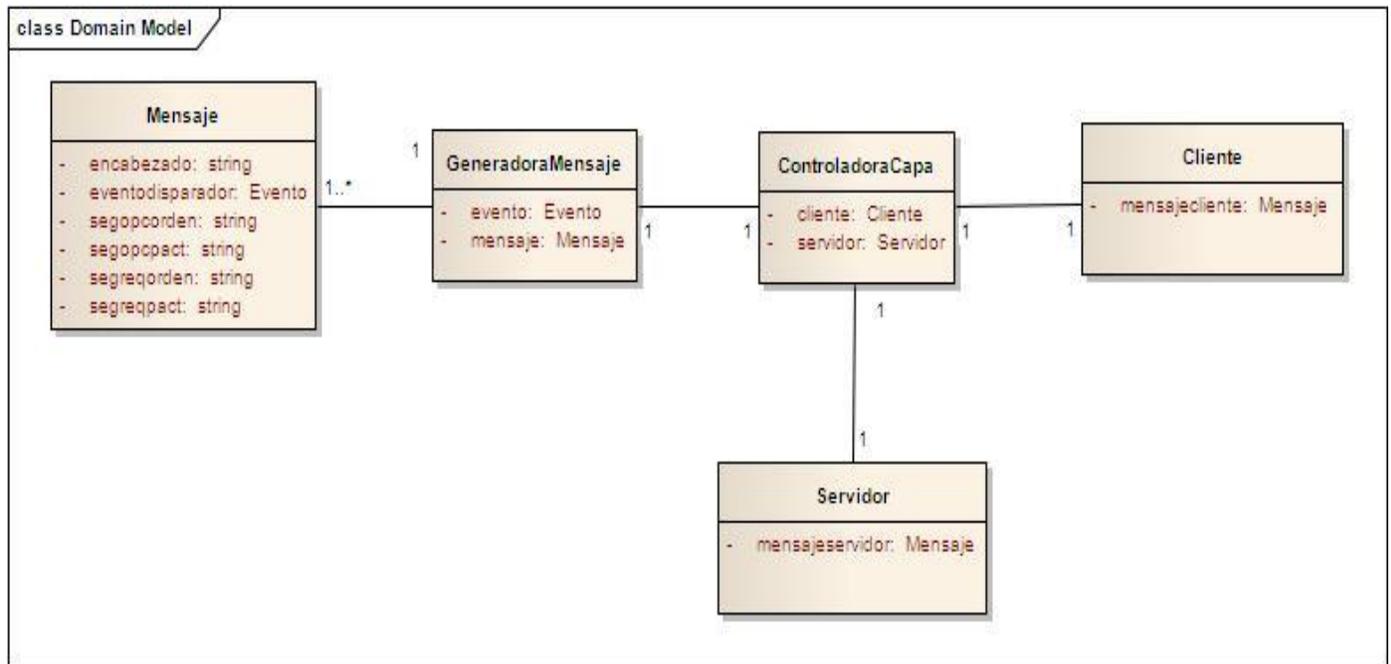


Fig. 5 Modelo de Dominio.

2.4 Especificación de requerimientos de software.

La ingeniería de requerimientos comprende las tareas relacionadas con la determinación de las necesidades o de las condiciones a satisfacer para un software nuevo o modificado, tomando en cuenta los diversos requerimientos de los inversores, que pueden entrar en conflicto entre ellos. Tiene como propósito definir y validar los requerimientos con los que debe cumplir el software; así como provee a los desarrolladores del sistema un mejor entendimiento de los requerimientos del sistema.

Una especificación de requerimientos del software es una descripción completa del comportamiento del sistema a desarrollar. También contiene requerimientos no funcionales (o suplementarios). Los requerimientos no funcionales son los requerimientos que imponen restricciones al diseño o funcionamiento del sistema (tal como requerimientos de funcionamiento, estándares de calidad, o requerimientos del diseño).

Los requerimientos se dividen en tres tipos:

- Funcionales: son los que el usuario necesita que efectúe el software.

- No funcionales: son los "recursos" para que trabaje el sistema de información (redes, tecnología).
- Empresariales u Organizacionales: son el marco contextual en el cual se implantará el sistema para conseguir un objetivo macro. (49)

2.4.1 Requerimientos funcionales.

Como se define en la ingeniería de requerimientos, los requerimientos funcionales establecen los comportamientos del sistema, son capacidades o condiciones que el mismo debe cumplir. A continuación se listan los requerimientos funcionales del sistema propuesto:

- RF_1 Notificar Mensaje "Admisión/Notificación de Visitas"
- RF_2 Notificar Mensaje de "Transferencia de un paciente"
- RF_3 Notificar Mensaje de "fin de visita o alta del paciente"
- RF_4 Notificar Mensaje de "registro del paciente"
- RF_5 Notificar Mensaje de " pre-admisión del paciente"
- RF_6 Notificar Mensaje de " cambio de estado de un paciente, externo-interno"
- RF_7 Notificar Mensaje de " cambio de estado de un paciente, interno-externo"
- RF_8 Notificar Mensaje de " actualización de información del paciente"
- RF_9 Notificar Mensaje de " partida y seguimiento de un paciente"
- RF_10 Notificar Mensaje de " arribo y seguimiento de un paciente"
- RF_11 Notificar Mensaje de "la cancelación de la admisión o visita de un paciente a la institución"
- RF_12 Notificar Mensaje de " la cancelación de la transferencia de un paciente"
- RF_13 Notificar Mensaje de " la cancelación del fin de visita o estadía del paciente en la institución."
- RF_14 Notificar Mensaje de " la planeación de admisión de un paciente."
- RF_15 Notificar Mensaje de " la planeación de transferencia de un paciente."
- RF_16 Notificar Mensaje de " la planeación del fin de visita de un paciente a la institución."
- RF_17 Notificar Mensaje de " intercambio de camas entre pacientes"
- RF_18 Notificar Mensaje de " la combinación de información del paciente (información actual-información previa)."
- RF_19 Notificar Mensaje de " la actualización del estado de la cama"
- RF_20 Notificar Mensaje de " la ausencia de un paciente."

Capítulo II: Características del Sistema, Análisis y Diseño de la Solución.

- RF_21 Notificar Mensaje de " el regreso de un paciente ausente."
- RF_22 Notificar Mensaje de "el borrado de la información de un paciente."
- RF_23 Notificar Mensaje de " la vinculación de información de pacientes."
- RF_24 Notificar Mensaje de " la cancelación de una planificación de fin de visita de una paciente."
- RF_25 Notificar Mensaje de " la cancelación de una planificación de transferencia de un paciente."
- RF_26 Notificar Mensaje de " la cancelación de la planificación de admisión de un paciente."
- RF_27 Notificar Mensaje de " información suplementaria de un paciente."
- RF_28 Notificar Mensaje de " la eliminación de información suplementaria de un paciente."
- RF_29 Notificar Mensaje de " la combinación o cambio de información suplementaria de un paciente."
- RF_30 Notificar Mensaje de " la actualización de la información de un paciente por recarga del paciente"
- RF_31 Notificar Mensaje de " la cancelación de la llegada de un paciente a una nueva localización dentro de la institución."
- RF_32 Notificar Mensaje de " la cancelación del movimiento de un paciente dentro de la institución."
- RF_33 Notificar Mensaje de " la combinación o cambio del identificador del paciente."
- RF_34 Notificar Mensaje de " la combinación o cambio del número de usuario del paciente."
- RF_35 Notificar Mensaje de " la combinación o cambio del identificador y el número de usuario de un paciente."
- RF_36 Notificar Mensaje de " la desvinculación de información de pacientes."
- RF_37 Notificar Mensaje de " la cancelación de la pre-admisión de un paciente."
- RF_38 Notificar Mensaje de " la combinación de los identificadores universales de persona y paciente."
- RF_39 Notificar Mensaje de " la combinación de los identificadores de lista de persona y paciente."
- RF_40 Notificar Mensaje de " la combinación del usuario y número de usuario del paciente."
- RF_41 Notificar Mensaje de " la combinación de visitas en un número de visita."
- RF_42 Notificar Mensaje de "el movimiento de información de un paciente a otro."

Capítulo II: Características del Sistema, Análisis y Diseño de la Solución.

- RF_43 Notificar Mensaje de "el movimiento de información de usuario de un paciente a otro."
- RF_44 Notificar Mensaje de "el movimiento de la información de una visita de un usuario a otro."
- RF_45 Notificar Mensaje de "el cambio de identificador universal de un paciente."
- RF_46 Notificar Mensaje de "el cambio de identificador de un paciente."
- RF_47 Notificar Mensaje de "el cambio de identificador universal alternativo de un paciente."
- RF_48 Notificar Mensaje de "el cambio de número de usuario de un paciente."
- RF_49 Notificar Mensaje de "el cambio de número de visita de un paciente."
- RF_50 Notificar Mensaje de "el cambio del identificador de visita alternativo de un paciente."
- RF_51 Notificar Mensaje de " respuesta en el modo original"
- RF_52 Notificar Mensaje de " respuesta en el modo mejorado"
- RF_53 Notificar los mensajes relacionados con los resultados de procedimientos
- RF_54 Notificar los mensajes para la " nueva reservación de cita/estudio"
- RF_55 Notificar los mensajes para la " reprogramación de cita/estudio "
- RF_56 Notificar los mensajes para la "modificación de la programación cita/estudio "
- RF_57 Notificar los mensajes para la "cancelación de cita/estudio "
- RF_58 Notificar los mensajes para la "descontinuación de la cita/estudio "
- RF_59 Notificar los mensajes para la "eliminación de una reservación para cita/estudio"
- RF_60 Notificar los mensajes para la "asignación de nuevos estudios a una misma cita "
- RF_61 Notificar los mensajes para la "modificación de estudios de una cita "
- RF_62 Notificar los mensajes para la "cancelación de estudios de una cita "
- RF_63 Notificar los mensajes para la "descontinuación de estudios en una cita"
- RF_64 Notificar los mensajes para la "eliminación de estudios en una cita "
- RF_65 Notificar los mensajes para el "bloqueo de espacios reservables para programación de citas o estudios "
- RF_66 Notificar los mensajes para el "desbloqueo a espacios reservables para programación de citas o estudios"
- RF_67 Notificar los mensajes para la "no presentación de un paciente a citas o estudios"
- RF_68 Notificar los mensajes relacionados con las solicitudes de programación y su respuesta
- RF_69 Notificar los mensajes para la "Petición de reservación de cita o estudio"
- RF_70 Notificar los mensajes para la "Petición de reprogramación de cita o estudio"

- RF_71 Notificar los mensajes para la “Petición de modificación de cita o estudio”
- RF_72 Notificar los mensajes para la “Petición de cancelación de cita o estudio”
- RF_73 Notificar los mensajes para la “Petición de discontinuación de cita o estudio”
- RF_74 Notificar los mensajes para la “Petición de eliminación de cita o estudio”
- RF_75 Notificar los mensajes para la “Petición de nueva asignación de estudios a una cita”
- RF_76 Notificar los mensajes para la “Petición de modificación a los estudios de una cita”
- RF_77 Notificar los mensajes para la “Petición de cancelación de estudios de una cita”
- RF_78 Notificar los mensajes para la “Petición de discontinuación de estudios en una cita”
- RF_79 Notificar los mensajes para la “Petición de eliminación de estudios en una cita”
- RF_80 Establecer la comunicación con otro sistema usando el protocolo MLLP de comunicación.
- RF_81 Conformar un mensaje para ser enviado usando el protocolo MLLP de comunicación.
- RF_82 Recibir mensajes enviados hacia el servicio usando el protocolo MLLP de comunicación.

2.5 Requerimientos no funcionales

Un requerimiento no funcional es, en la ingeniería de sistemas y la ingeniería de software, un requerimiento que especifica criterios que pueden usarse para juzgar la operación de un sistema en lugar de sus comportamientos específicos, ya que éstos corresponden a los requerimientos funcionales, (50) son propiedades o cualidades que el producto debe tener. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable. (51) Los que son clasificados de la siguiente forma:

2.5.1 Usabilidad

RNF1 El servicio web podrá ser utilizado por cualquier sistema.

2.5.2 Soporte

El soporte es un aspecto realmente amplio y abarca todas las acciones a tomar una vez que se ha terminado el desarrollo del software con motivos de asistir a los clientes de este, así como lograr su

Capítulo II: Características del Sistema, Análisis y Diseño de la Solución.

mejoramiento progresivo y evolución en el tiempo. Puede incluir: pruebas, adaptabilidad, mantenimiento, compatibilidad, configuración, servicios, instalación y portabilidad, por solo mencionar algunos. (52)

RNF2 El sistema contará antes de su puesta en marcha con un período de pruebas que permitirá identificar y corregir errores cometidos durante el desarrollo.

RNF3 Se generará la documentación apropiada para agilizar su mantenimiento y configuración.

2.5.3 Hardware

RNF4 Se escogieron estaciones de trabajo de 1GB de memoria RAM y 40 GB de disco duro.

2.5.4 Software

RNF5 Se requiere la presencia del servidor Web Apache.

RNF6 El sistema necesita contar con el framework .Net v3.5 para su funcionamiento.

RNF7 Deben contar con los navegadores Internet Explorer o Mozilla Firefox.

2.6 Especificación de Casos de Uso

Los casos de uso son utilizados para capturar el comportamiento deseado del sistema en desarrollo, sin tener que especificar cómo es implementado este comportamiento. Ofrecen un medio para que los desarrolladores, los usuarios finales del sistema y los expertos del dominio lleguen a una comprensión común del sistema. Un caso de uso describe un proceso de principio a fin, por tanto, una secuencia de eventos, las acciones y las transacciones que se requieren para realizarlo. (53)

2.6.1 Definición de los actores.

Actor	Descripción
Administrador	Rol responsable de configurar los puertos para hacer las conexiones necesarias para el envío de mensaje y recibo de mensajes.
Sistema1	Rol ficticio para representar los sistemas HIS-RIS como inicializadores de algún caso

	de uso.
Sistema2	Rol ficticio para representar los sistemas HIS-RIS-PACS como inicializadores de algún caso de uso.

2.6.2 Diagrama de Paquete: Administración de Pacientes (ADT).

(Remitirse a: <https://repositorio.cesim.prod.uci.cu/svn/tie/tie/IHE-Radiologia/>

1.Raiz/Expediente.Proyecto/1. ingenieria/1.2 arquitectura y diseño/Diagramas)

2.6.3 Diagrama de Paquete: Aceptación (ACK).

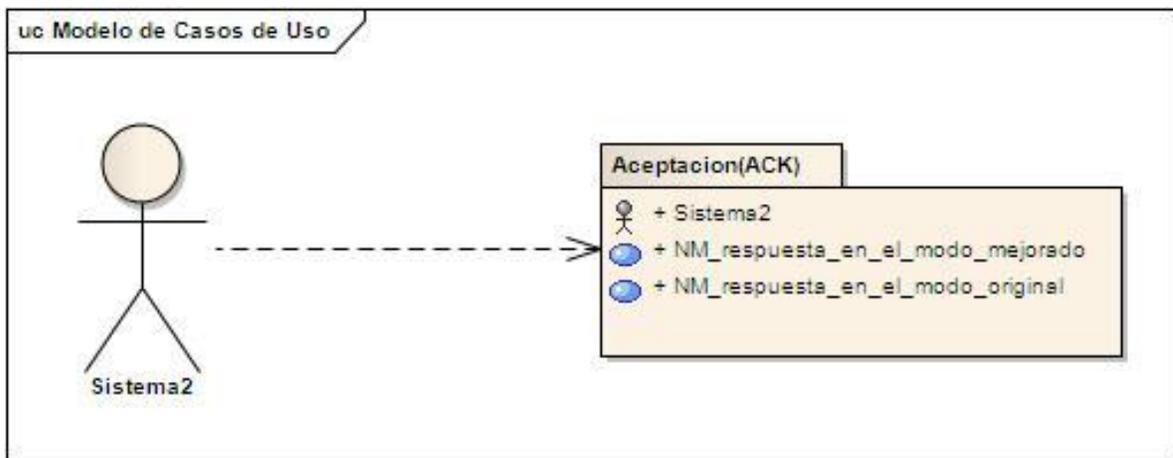


Fig. 6 Diagrama de paquete: Aceptación (ACK).

2.6.4 Diagrama de paquete: Programación de citas o estudios no solicitados. (SIU)

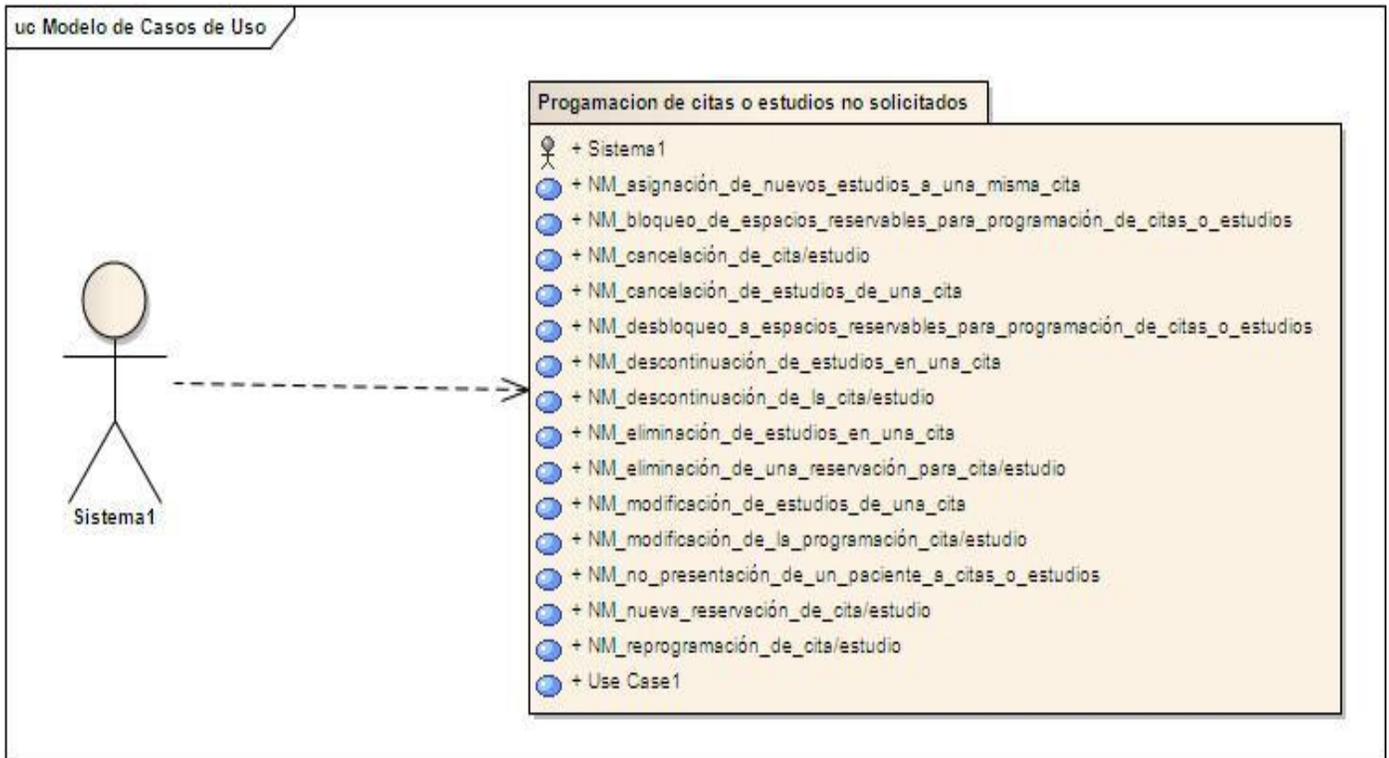


Fig. 7 Diagrama de paquete: Programación de citas o estudios no solicitados

2.6.5 Diagrama de Paquete: Programación de citas o estudios y sus respuestas. (SRM/SRR)

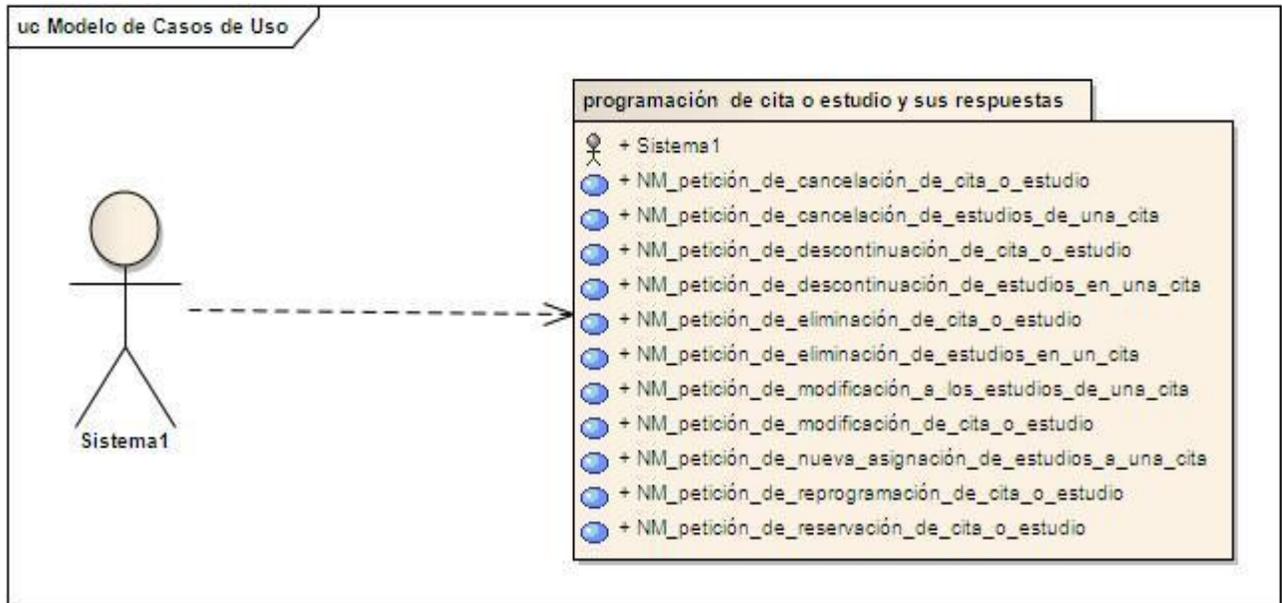


Fig. 8 Diagrama de paquete: Programación de citas o estudios y sus respuestas. (SRM/SRR)

2.6.6 Diagrama de Paquete: Solicitud de programación de citas o estudios y sus respuestas. (SQM/SQR)

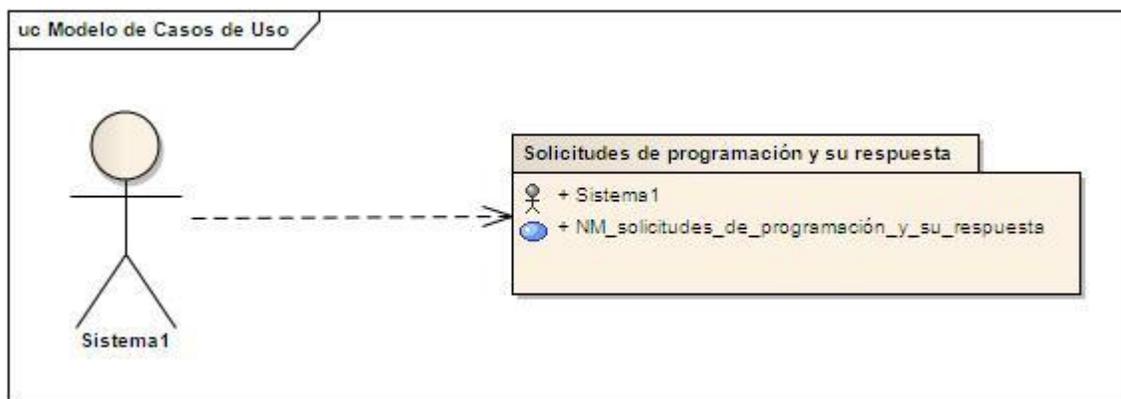


Fig. 9 Diagrama de paquete: Solicitud de programación de citas o estudios y sus respuestas. (SQM/SQR).

2.6.7 Diagrama de Paquete: Resultados. (ORU)

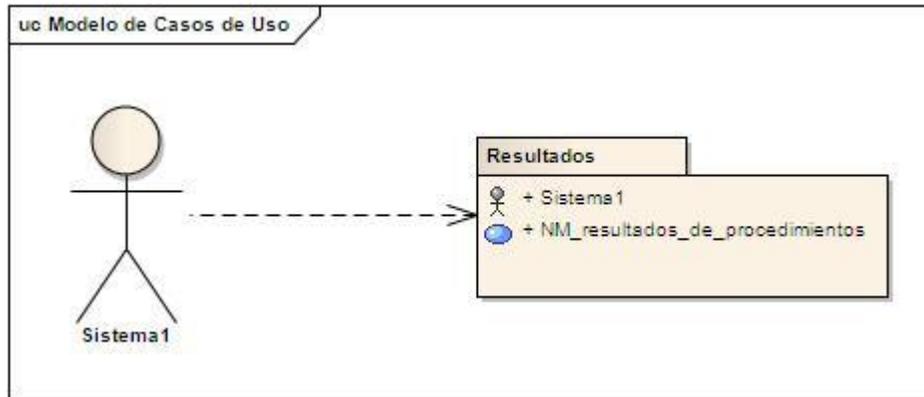


Fig. 10 Diagrama de paquete: Resultados. (ORU)

2.6.8 Diagrama de Paquete: Protocolo MLLP.

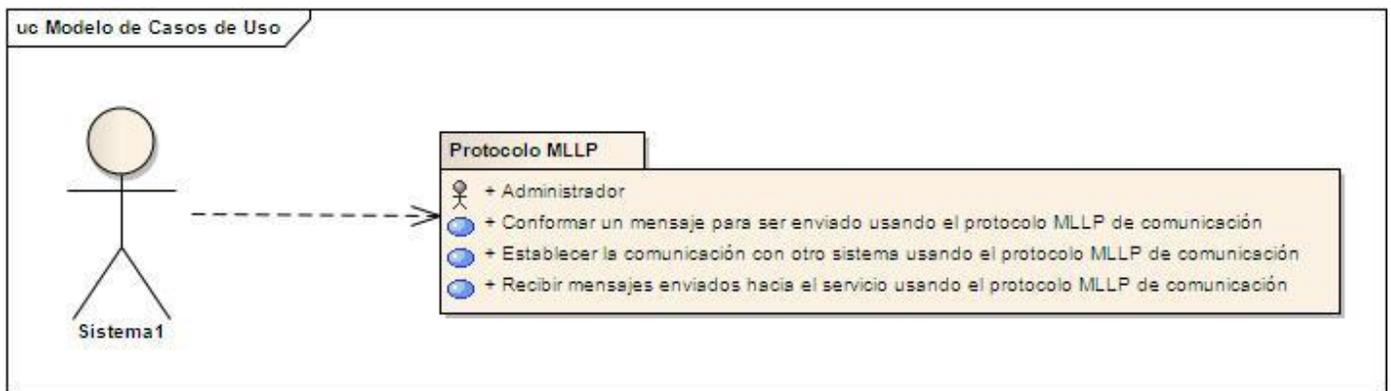


Fig. 11 Diagrama de paquete: Protocolo MLLP.

2.6.9 Descripción Textual de los Casos de Uso.

CU-1	Notificar Mensaje de "Admisión/Notificación de Visitas"
Actor	Sistema1
Descripción	El CU inicia cuando el actor admite el paciente y solicita al sistema la notificación de la admisión de un paciente.
Referencia	RF_1

Capítulo II: Características del Sistema, Análisis y Diseño de la Solución.

CU-2	Notificar Mensaje de "Transferencia de un Paciente"
Actor	Sistema1
Descripción	El CU inicia cuando el actor transfiere a un paciente y solicita al sistema la notificación de la transferencia de un paciente.
Referencia	RF_2

CU-21	Notificar Mensaje de "regreso de un paciente ausente."
Actor	Sistema1
Descripción	El CU inicia cuando el actor registra el regreso de un paciente ausente y solicita al sistema la notificación del regreso de un paciente ausente.
Referencia	RF_21

CU-30	Notificar Mensaje de "actualización de la información de un paciente por recarga del paciente"
Actor	Sistema1
Descripción	El CU inicia cuando el actor actualiza la información del paciente y solicita al sistema la notificación de la actualización de la información de un paciente por recarga del paciente.
Referencia	RF_30

CU-52	Notificar respuestas en el modo mejorado.
Actor	Sistema2
Descripción	El CU inicia cuando el sistema conforma un mensaje con la estructura requerida para su envío.
Referencia	RF_52

2.7 Arquitectura en Capas:

Los estilos arquitectónicos son los que definen la estructura de un componente de software y se componen de subsistemas con sus responsabilidades. Una vez concluida la ingeniería de requisitos, donde se definieron las características y restricciones del componente, se analizaron varios estilos arquitectónicos, con el fin de obtener la estructura que mejor se ajusta a los requisitos.

Luego de un estudio realizado sobre los estilos arquitectónicos existentes, se definió que el de arquitectura en capas, perteneciente a la familia de Llamada y Retorno, será el que se utilizará. El estilo de arquitectura en capas, es una organización jerárquica donde cada capa proporciona servicios a la capa inmediata superior y se sirve de las prestaciones que le brinda la inmediata inferior. Las capas o niveles pueden estar implementadas internamente por objetos o procedimientos. Además, permiten en caso de realizarse cambios, que sólo se afecte a la capa donde se encuentre la modificación.

2.8 Arquitectura Orientada a Servicios (SOA):

IBM: "...SOA es un modelo de componentes que interrelaciona las diferentes unidades funcionales de las aplicaciones, denominadas servicios, a través de interfaces y contratos bien definidos entre esos servicios. La interfaz se define de forma neutral, y debería ser independiente de la plataforma hardware, del sistema operativo y del lenguaje de programación utilizado. Esto permite a los servicios, construidos sobre sistemas heterogéneos, interactuar entre ellos de una manera uniforme y universal."

Las Arquitecturas Orientadas a Servicios, están motivadas por la creciente necesidad de los negocios de responder con rapidez a los cambios en el entorno comercial en que se desenvuelven. Esto los lleva a tener que cambiar sus sistemas tecnológicos con esa misma rapidez y para lograrlo es necesario que los componentes de esta infraestructura, sean tan reutilizables y poco interdependientes que permitan una rápida reestructuración de los mismos. (Véase figura 12)

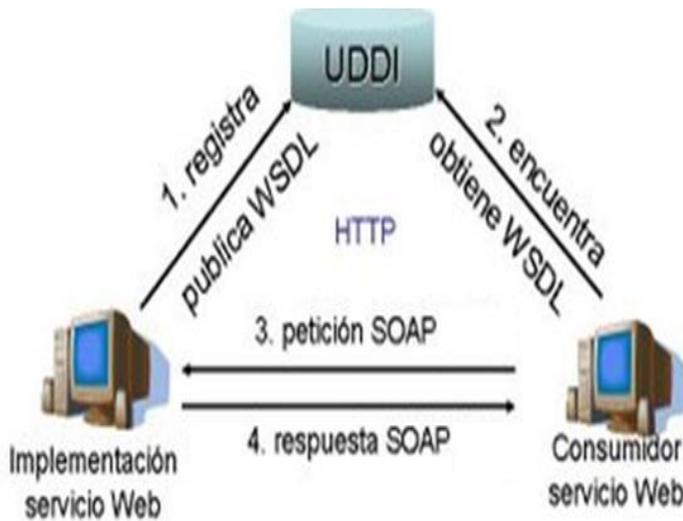


Figura 12: SOAP

2.9 Patrones GRASP

Los patrones GRASP (por sus siglas en inglés *General Responsibility Assignment Software Patterns*) son patrones generales de *software* para la asignación de responsabilidades. Se utilizan alta cohesión y controlador, para el desarrollo del Servicio Web debido a que permiten establecer algunos parámetros útiles para el diseño del producto.

- **Alta cohesión:** Indica que la información que almacena una clase debe ser coherente, de manera que todos sus métodos tengan un comportamiento bien definido. La cohesión es una medida de cuán relacionadas y enfocadas están las responsabilidades de una clase. Una alta cohesión caracteriza a las clases con responsabilidades estrechamente relacionadas que no realicen un trabajo enorme.
- **Controlador:** Se encarga de gestionar los eventos generados en capas anteriores y a partir de dichos eventos tomar las decisiones apropiadas, pudiendo invocar funcionalidades contenidas en capas más profundas como el acceso a datos. Su función es de mediador o intermediario, del controlador del negocio asociados. (54)

2.10 Clases del Diseño

2.10.1 Descripción de las clases del diseño

El Diagrama de Clases es el diagrama principal para el análisis y diseño. Un diagrama de clases presenta las clases del sistema con sus relaciones estructurales y de herencia. La definición de clase incluye definiciones para atributos y operaciones (55)

Capítulo II: Características del Sistema, Análisis y Diseño de la Solución.

Nombre: Mensajes	
Tipo de clase: Controladora	
Para cada responsabilidad:	
Nombre:	GenerarMensajeACK(): int
Descripción:	Se utiliza para generar mensajes ACK.
Nombre:	GenerarMensajeADR_A19(): int
Descripción:	Se utiliza para generar mensajes de respuestas a los ADT.
Nombre:	GenerarMensajeADT_A01(): int
Descripción:	Se utiliza para generar mensajes de Admisión/Notificación de Visitas.
Nombre:	GenerarMensajeADT_A02(): int
Descripción:	Se utiliza para generar mensajes de Transferencia de un paciente.
Nombre:	GenerarMensajeADT_A03 () : int
Descripción:	Se utiliza para generar mensajes de Fin de visita o alta del paciente.
Nombre:	GenerarMensajeADT_A05(): int
Descripción:	Se utiliza para generar mensajes de pre-admisión del paciente.
Nombre:	GenerarMensajeADT_A06(): int
Descripción:	Se utiliza para generar mensajes de cambio de estado de un paciente, externo-interno.
Nombre:	GenerarMensajeADT_A09(): int
Descripción:	Se utiliza para generar mensajes de partida y seguimiento de un paciente.
Nombre:	GenerarMensajeADT_A12(): int
Descripción:	Se utiliza para generar mensajes de cancelación de la transferencia de un paciente.
Nombre:	GenerarMensajeADT_A15(): int
Descripción:	Se utiliza para generar mensajes de planeación de transferencia de un paciente.
Nombre:	GenerarMensajeADT_A16(): int
Descripción:	Se utiliza para generar mensajes de planeación del fin de visita de un paciente a la institución.
Nombre:	GenerarMensajeADT_A17(): int
Descripción:	Se utiliza para generar mensajes de intercambio de camas entre pacientes.

Capítulo II: Características del Sistema, Análisis y Diseño de la Solución.

Nombre:	GenerarMensajeADT_A18(): int
Descripción:	Se utiliza para generar mensajes de combinación de información del paciente (información actual-información previa).
Nombre:	GenerarMensajeADR_A19(): int
Descripción:	Se utiliza para generar mensajes de la notificación de consultas de pacientes.
Nombre:	GenerarMensajeADT_A20(): int
Descripción:	Se utiliza para generar mensajes de ausencia de un paciente.
Nombre:	GenerarMensajeADT_A21(): int
Descripción:	Se utiliza para generar mensajes de regreso de un paciente ausente.
Nombre:	GenerarMensajeADT_A24(): int
Descripción:	Se utiliza para generar mensajes de cancelación de una planificación de fin de visita de una paciente.
Nombre:	GenerarMensajeADT_A30(): int
Descripción:	Se utiliza para generar mensajes de actualización de la información de un paciente por recarga del paciente.
Nombre:	GenerarMensajeADT_A37(): int
Descripción:	Se utiliza para generar mensajes de cancelación de la pre-admisión de un paciente.
Nombre:	GenerarMensajeADT_A38(): int
Descripción:	Se utiliza para generar mensajes de combinación de los identificadores universales de persona y paciente.
Nombre:	GenerarMensajeADT_A39(): int
Descripción:	Se utiliza para generar mensajes de combinación de los identificadores de lista de persona y paciente.
Nombre:	GenerarMensajeADT_A43(): int
Descripción:	Se utiliza para generar mensajes de movimiento de información de usuario de un paciente a otro.
Nombre:	GenerarMensajeADT_A45(): int
Descripción:	Se utiliza para generar mensajes de cambio de identificador universal de un paciente.

Capítulo II: Características del Sistema, Análisis y Diseño de la Solución.

Nombre:	GenerarMensajeADT_A50(): int
Descripción:	Se utiliza para generar mensajes de cambio del identificador de visita alternativo de un paciente.
Nombre:	GenerarMensajeORU_R01(): int
Descripción:	Se utiliza para generar mensajes relacionados con los resultados de procedimientos.
Nombre:	GenerarMensajeSIU_S12(): int
Descripción:	Se utiliza para generar mensajes de citas o estudios.
Nombre:	GenerarMensajeSQM_S25(): int
Descripción:	Se utiliza para generar mensajes de solicitudes de programación.
Nombre:	GenerarMensajeSQR_S25(): int
Descripción:	Se utiliza para generar mensajes de respuesta de las solicitudes de programación.
Nombre:	GenerarMensajeSRM_S01(): int
Descripción:	Se utiliza para generar mensajes de solicitud para la reservación de una cita o estudio.
Nombre:	GenerarMensajeSRR_S01(): int
Descripción:	Se utiliza para generar mensajes de respuestas de la solicitud para la reservación de una cita o estudio.

Nombre: Server

Tipo de clase: Controladora

Server	
<pre> - AETitle: string - id: string - listener: TopListener - mensajeFinal: string - port: int - thread_listener: Thread - timer: System.Timers.Timer </pre>	
<pre> - AcceptConnection() : void + Close() : void + ReceiveFile(object) : void + RecibirMensaje(int) : string + RecibirMensaje() : string + RetirarProtocolo(string) : string + Server(int, string, string) + Start() : void - timer_Elapsed(object, System.Timers.ElapsedEventArgs) : void + VerificarProtocolo(string) : bool </pre>	
<pre> «property» + Port() : int </pre>	

Para cada responsabilidad:

Nombre:	AcceptConnection(): void
Descripción:	Acepta la conexión que viene por socket de una pc.
Nombre:	Close(): void
Descripción:	Cierra el socket por el que está escuchando.
Nombre:	ReciveFile(object):void
Descripción:	Recibe los mensajes que llegan por la red.
Nombre:	RecibirMensaje(int): string
Descripción:	Recibe los mensajes de respuesta ACK.
Nombre:	RetirarProtocolo(string): string
Descripción:	Retira el protocolo MLLP del mensaje recibido.
Nombre:	Start(): void
Descripción:	Inicializar el servicio de recibo de mensajes.
Nombre:	VerificarProtocolo(stirng): bool
Descripción:	Verifica que el mensaje recibido cumpla con el protocolo MLLP.

2.11 Diagramas de clases del diseño

(Remitirse a: https://repositorio.cesim.prod.uci.cu/svn/tie/tie/IHE-Radiologia/1.Raiz/Expediente.Proyecto/1.ingenieria/1.2arquitectura_y_diseño/Diagramas)

2.12 Diagramas de Secuencia

Mediante los diagramas de Secuencia se muestra la secuencia de mensajes entre objetos, resaltando el orden temporal de los mensajes que se intercambian en un escenario.

(Remitirse a: https://repositorio.cesim.prod.uci.cu/svn/tie/tie/IHE-Radiologia/1.Raiz/Expediente.Proyecto/1.ingenieria/1.2arquitectura_y_diseño/Diagramas)

Se abordó en el presente capítulo las principales características del sistema, así como un análisis del modelo de dominio que contiene las principales clases por las que debe realizarse su desarrollo. Se definió como solución, desarrollar un servicio web que podrá ser utilizado por cualquier sistema, sin importar en que plataforma esté desarrollado. Se identificaron los requisitos que debe cumplir conformando las funcionalidades del sistema mediante la obtención de casos de usos. Fueron definidas las clases del diseño y la relación e interacción entre estas; detallándose así, cómo va a lograr el sistema su correcto funcionamiento.

Además, se realizó la descripción de la arquitectura del software a seguir, obteniendo como resultado una arquitectura basada en capas. Esta permitirá una mejor organización cuando sea necesaria la realización de cambios en el sistema. Así como los patrones de diseño aplicados, contribuyendo a obtener la calidad requerida y entendimiento, y de esta forma dar paso a la implementación.

CAPITULO 3: IMPLEMENTACIÓN Y PRUEBA.

El presente capítulo tiene como objetivo abordar el flujo de trabajo de implementación donde se describe cómo se implementan en términos de componentes los elementos del modelo de diseño. Se modelan los diagramas de componentes y de despliegue, dando una visión de cómo quedará desarrollada y distribuida la aplicación. Además, se especifica el tipo de prueba a utilizar para encontrar y documentar los defectos que puedan afectar la calidad del software y validar que trabaje como fue diseñado, cumpliendo con los requisitos planteados.

3.1 Modelo de implementación

El modelo de implementación especifica cómo los elementos del modelo de diseño, fundamentalmente las clases, se implementan en términos de componentes como ficheros de código fuente, ejecutables, librerías, entre otros. También describe cómo se organizan los componentes de acuerdo con los mecanismos de estructuración disponibles en el entorno de implementación y en el lenguaje o lenguajes de programación utilizados, y cómo dependen los componentes unos de otros, así como su organización de acuerdo a los nodos específicos en el modelo de despliegue. (56)

3.1.1 Diagrama de despliegue

Un diagrama de despliegue muestra las relaciones físicas entre los componentes hardware y software en el sistema final, es decir, la configuración de los elementos de procesamiento en tiempo de ejecución y los componentes de software (procesos y objetos que se ejecutan en ellos). Se representa mediante un grafo de nodos unidos por conexiones de comunicación. Un nodo es un objeto físico en tiempo de ejecución que representa un recurso computacional. (57) En la figura 13 se muestra el Diagrama de Despliegue del servicio web.

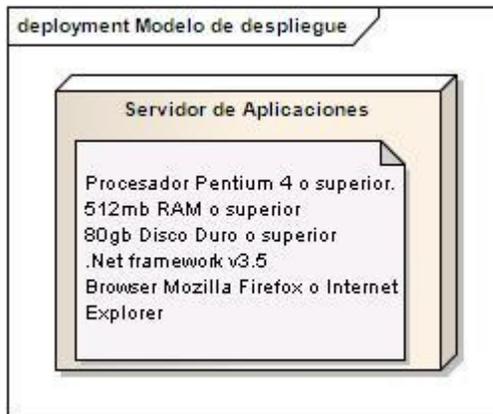


Fig. 13 Diagrama de Despliegue.

3.1.2 Diagrama de componentes

Dentro del flujo de implementación se crea como artefacto el diagrama de componentes que describe los elementos físicos del sistema y sus relaciones; así como muestra las opciones de realización incluyendo código fuente, binario y ejecutable. Los componentes representan todos los tipos de elementos de software que entran en la fabricación de aplicaciones informáticas. Estos pueden ser simples archivos, paquetes, bibliotecas cargadas dinámicamente, entre otros. (58) En la figura 14 se muestra el Diagrama de Componentes del servicio web.

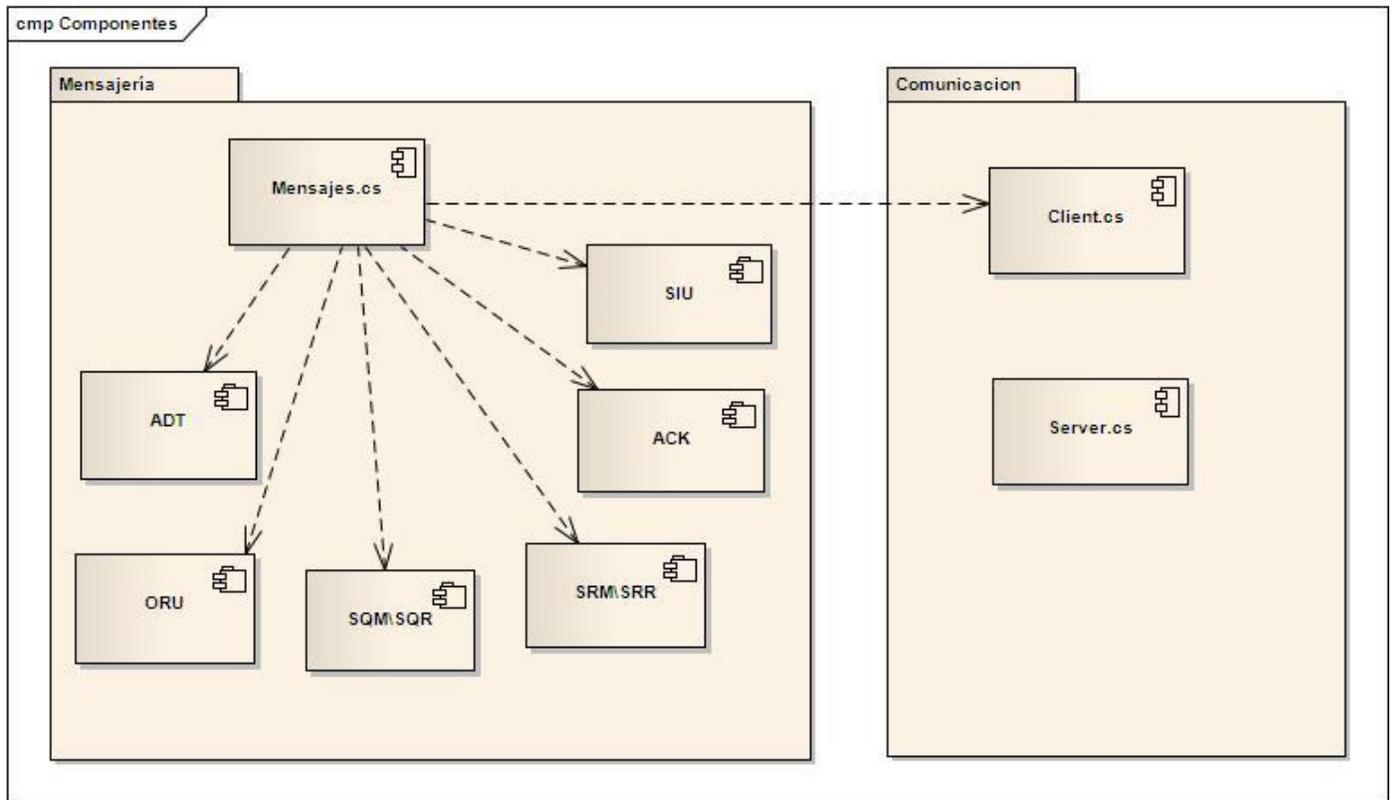


Fig. 14 Diagrama de Componentes.

3.2 Tratamiento de errores

El tratamiento de errores es un aspecto que se debe realizar sin importar el tipo de aplicación. Cuando no se tienen en cuenta los errores que puede dar el sistema y por consiguiente no se tratan los mismos es imposible comunicarle al usuario la causa y motivo del error.

En el sistema se identifican cada uno de los puntos en los que puede darse situaciones excepcionales que puedan dar al traste con errores de ejecución de las soluciones. Se definen cada uno de los posibles errores y como deben ser mostrados los mismos de manera entendible para el usuario, además este manejo de errores se realiza en dependencia del tipo de error lanzado. Un buen control de excepciones garantiza una mayor calidad de las funcionalidades brindadas; sobre este aspecto el lenguaje de programación empleado posibilita un manejo acorde de estos elementos, tratando los casos no básicos por separado, lo cual permite especificar qué acciones a realizar en dependencia del tipo de excepción. Ver figura 15.

EJEMPLO DE TRATAMIENTO DE EXCEPCIONES

```
7471 static public int GenerarMensajeACK(string IP, int puerto, string p_msh3,
7472                                     string p_msh4, string p_msh5, string p_msh6,
7473                                     string p_msh7, string p_msh10, string p_msh11,
7474                                     string p_msa1, string p_msa2, string p_msa3, string p_ed11, string p_ed13)
7475 {
7476     try
7477     {
7478         ACKCONTENT ack = new ACKCONTENT(p_msh3, p_msh4, p_msh5, p_msh6, p_msh7,
7479                                         p_msh10, p_msh11, p_msa1, p_msa2, p_msa3, p_ed11, p_ed13);
7480         XmlSerializer serializer = new XmlSerializer(typeof(ACKCONTENT));
7481         MemoryStream stream = new MemoryStream();
7482         serializer.Serialize(stream, ack);
7483         return Client.SendFile(IP, puerto, stream.GetBuffer());
7484     }
7485     catch (Exception E)
7486     {
7487
7488         throw new Exception("Error generando el mensaje ACK predeterminado :" + E.Message);
7489     }
7490 }
7491
```

Fig. 15 Tratamiento de excepciones.

3.3 Estrategias de codificación. Estándares y estilos a utilizar.

Un estándar de codificación completo comprende todos los aspectos de la generación de código. Si bien los programadores deben implementar un estándar de forma prudente, éste debe tender siempre a lo práctico. Un código fuente completo debe reflejar un estilo armonioso, como si un único programador hubiera escrito todo el código de una sola vez.

La legibilidad del código fuente repercute directamente en lo bien que un programador comprende un sistema de software. La mantenibilidad del código es la facilidad con que el sistema de software puede modificarse para añadirle nuevas características, modificar las ya existentes, depurar errores, o mejorar el

rendimiento. Aunque la legibilidad y la mantenibilidad son el resultado de muchos factores, una faceta del desarrollo de software en la que todos los programadores influyen especialmente es en la técnica de codificación. El mejor método para asegurarse de que un equipo de programadores mantenga un código de calidad es establecer un estándar de codificación sobre el que se efectuarán luego revisiones del código de rutinas.

Si se aplica de forma continuada un estándar de codificación bien definido, se utilizan técnicas de programación apropiadas, y, posteriormente, se efectúan revisiones del código de rutinas, caben muchas posibilidades de que un proyecto de software se convierta en un sistema de software fácil de comprender y de mantener.

A continuación se muestran los elementos de los estándares de codificación utilizados para el desarrollo del sistema en cuestión.

Notación Camello

Se emplea para denotar variables y parámetros. Especifica que la palabra de inicio del identificador comienza con minúscula. Si el identificador está compuesto por más de una palabra entonces éstas deben comenzar con mayúsculas. (59)

Notación Pascal

Se emplea para denotar los métodos (funciones) y las clases. Especifica que la palabra de inicio del identificador comienza con mayúscula. Si el identificador está compuesto por más de una palabra entonces éstas deben comenzar con mayúsculas. (60)

A continuación se especifican algunas restricciones para la nomenclatura y estándares de codificación.

Indentación	
Objetivo: Lograr una estructura uniforme para los bloques de código así como para los diferentes niveles de anidamiento.	
Inicio y fin de bloque	Se recomienda dejar dos espacios en blanco desde la instrucción anterior para el inicio y fin de bloque {}. Lo mismo sucede para el caso de las instrucciones if, else, for, while, do while, switch, foreach.
Aspectos Generales	El indentado debe ser de dos espacios por bloque de código. No se debe usar el tabulador; ya que este puede variar según la PC o la configuración de dicha tecla.

	<p>Los inicios ({) y cierre (}) de ámbito deber estar alineados debajo de la declaración a la que pertenecen y deben evitarse si hay sólo una instrucción.</p> <p>Nunca colocar { en la línea de un código cualquiera, esto requiere una línea propia.</p>	
Comentarios, separadores, líneas, espacios en blanco y márgenes.		
Objetivo: Establecer un modo común para comentar el código de forma tal que sea comprensible con sólo leerlo una vez.		
Ubicación de comentarios	Al inicio de cada clase o función y al final de cada bloque de código.	Se recomienda comentar al inicio de la clase o función especificando el objetivo de la misma así como los parámetros que usa (especificar tipos de dato, y objetivo del parámetro) entre otras cosas.
Líneas en blanco	Se emplean antes y después de métodos, clases y estructuras.	Se recomienda dejar una línea en blanco antes y después de la declaración de una clase o de una estructura y de la implementación de una función
Espacios en blanco	Entre operadores lógicos y aritméticos.	Se recomienda usar espacios en blanco entre estos operadores para lograr una mayor legibilidad en el código. Ejemplo: producto = nomproducto
Aspectos generales	Sobre el comentario	Se debe evitar comentar cada línea de código. Cuando el comentario se aplica a un grupo de instrucciones debe estar seguido de una línea en blanco. En caso de que se necesite comentar una sola instrucción se suprime la línea en blanco o se escribe a continuación de la instrucción
	Sobre los espacios en blanco	No se debe usar espacio en blanco: Después del corchete abierto y antes del cerrado de un arr Después del paréntesis abierto y antes del cerrado. Antes de un punto y coma.
Variables y constantes		
Apariencia de variables	Las variables tendrán un prefijo para el tipo de datos en minúscula.	El nombre que se le da a las variables debe comenzar con la primera letra en minúscula, la cual identificara el tipo de datos al que se refiere (ver tabla 1.1), en caso de que sea un nombre compuesto se empleará notación Camell. Ejemplo: sNombrePaciente
Apariencia de constantes	Todas sus letras en mayúscula	Se deben declarar las constantes con todas sus letras en mayúscula.
Aspectos generales	Nombres de las variables y constantes	El nombre empleado, debe permitir que con sólo leerlo se conozca el propósito de la misma.
Clases y Objeto		

Objetivo: Nombrar las clases e instancias de forma estándar para todas las aplicaciones.		
Apariencia de clases y objetos	Primera letra en mayúscula	Los nombres de las clases deben comenzar con la primera letra en mayúscula y el resto en minúscula, en caso de que sea un nombre compuesto se empleará notación Pascal. Ejemplo: MiClase(). Para el caso de las instancias se comenzara con un prefijo que identificara el tipo de dato, este se escribirá en minúscula.
Apariencia de atributos	Primera letra en minúscula	El nombre que se le da a los atributos de las clases debe comenzar con la primera letra en minúscula, la cual estará en correspondencia al tipo de dato al que se refiere, en caso de que sea un nombre compuesto se empleará notación Camell.
Apariencia de las funciones	Primera letra en mayúscula	Para nombrar las funciones se debe tratar de utilizar verbos que denoten la acción que hace la función. Se empleará notación Pascal. Ejemplo: funcion BuscarUnidad(). Si son funciones que obtienen un dato se emplea el prefijo get y si fijan algún valor se emplea el prefijo set
Declaración de parámetro en funciones	Agrupados por tipos Poner los string 1 numéricos 2, además, agrupar según valores por defecto.	Los parámetros que se le pasan a las funciones se recomienda sean declarados de forma tal que estén agrupados por el tipo de dato que contienen, especificando el tipo de datos (tabla 1.1).
Aspectos generales	Sobre las clases, los objetos, los atributos y las funciones.	El nombre empleado para las clases, objetos, atributos y funciones debe permitir que con sólo leerlo se conozca el propósito de los mismos.

Como resultado de este capítulo se logró una mejor comprensión del sistema, haciéndolo más sencillo y rápido de implementar. Se desarrollaron los modelos de componente y despliegue a fin de conseguir una implementación lo más cercana posible al diseño propuesto en el Capítulo 2. Se definieron políticas de implementación como el tratamiento adecuado de situaciones excepcionales con el objetivo de identificar los posibles errores de ejecución de la solución. Así como las restricciones del código, que especifican como deben trabajar cada uno de los desarrolladores del sistema, posibilitando una mayor eficiencia de la solución.

CONCLUSIONES

Al concluir este trabajo, se logró el desarrollo del servicio web para el envío de mensajes HL7 v2.3.1 dando cumplimiento al objetivo general de la investigación presentada. La puesta en marcha de este servicio permitirá ofrecer mejoras en cuanto a la comunicación entre sistemas, logrando así una mejor fluidez de la información entre los mismos.

Se realizó un estudio que condujo a la conclusión de que las soluciones existentes a nivel internacional poseen costos muy elevados para su obtención, por lo que dificultaría al país su utilización. Además, la solución que existe a nivel nacional no satisface las necesidades para una integración con otros productos desarrollados por otras empresas, ya que no implementa estándares tales como HL7 v2.3.1.

Todo el proceso de desarrollo del software ha estado sustentado en la utilización de metodologías, tecnologías y herramientas, tales como: RUP, para guiar el trabajo del equipo de desarrollo a través de un ciclo que permita de manera organizada traducir los requisitos de software a un sistema que garantice la solución a la problemática existente. Entre las herramientas escogidas figuran para el entorno de desarrollo .Net Framework v3.5, como IDE Visual Studio Team System 2008, para el modelado UML, Enterprise Architect v7.5 y como lenguaje de programación C# v3.0.

Durante la etapa de inicio del proyecto, se realizó un modelo de dominio para ofrecer una visión clara de los principales conceptos empleados asociados a las entidades implicadas. Además se describieron de forma detallada todos los casos de uso del sistema, que sirvieron de base para los posteriores flujos de Análisis y Diseño. Con la culminación de la etapa de elaboración del software, quedó definida una arquitectura basada en capas, dando paso a la implementación del servicio web. Esto es expresado mediante diagramas de componentes que reflejan la relación entre clases y paquetes.

Se realizó la implementación de un servicio web, que permite mejorar las comunicaciones entre los sistemas PACS, HIS y RIS. Lo que permitirá lograr una mejor integración entre sistemas de este tipo ya sean desarrollados por el Centro o por terceros proveedores y perfeccionar el trabajo en cualquier institución hospitalaria.

RECOMENDACIONES

Como resultado del presente trabajo se obtuvo un servicio web que aunque solo comprende los mensajes relacionados con el área de radiología, puede ser refinado con el objetivo de lograr un incremento en su calidad y para su utilización en otras áreas dentro del sector hospitalario, a continuación se exponen algunas recomendaciones a tener en cuenta para futuras versiones del producto:

- Implementar los mensajes HL7 v2.3.1 referente a las áreas de Cardiología, Laboratorio y Farmacia relacionados con el área de Radiología.
- Implementar la encriptación de los mensajes HL7 que viajan por la red para lograr la seguridad de los mismos.
- Instalar el sistema en posteriores versiones en plataformas libres.

REFERENCIAS BIBLIOGRÁFICAS

- (1) **Rodríguez López, M. y Rodríguez. García, R** (2010). "Propuesta de aplicación de los perfiles de integración de IHE entre los sistemas alas PACS–alas RIS–alas HIS.". La Habana, Universidad de las Ciencias Informáticas.
- (2) **Facmed**. from <http://www.facmed.unam.mx/emc/computo/ssa/HIS/hisindex.htm>
- (3) **HL7**. from <http://www.hl7.org/>.
- (4) **Alegre, L.** (2005). *RevistaeSalud*. Número 5. Vol. 2.
- (5) **Martínez Martínez, A.** from www.itescam.edu.mx/principal/sylabus/fpdb/recursos/r43242.PDF.
- (6) **García González, D. y Socias Pardo, J.** (2010). "Arquitectura para el sistema de almacenamiento de informes de estudios imagenológicos". La Habana, Universidad de las Ciencias Informáticas.
- (7) Ídem a la referencia 1.
- (8) **Rivero Castro, A. y Hernández Noguera, A.** (2010). " Aplicación para el registro de estudios imagenológicos. Solución para equipos de Oftalmología ". La Habana, Universidad de las Ciencias Informáticas.
- (9) **Martínez Carrillo, F.** from www.unipamplona.edu.co/.../portallG/.../diseno_desarrollo_pacs.pdf.
- (10) **González Abreu, L. E. y Fonseca Guzman, M** (2010). "Módulo de gestión para los reportes estadísticos de alas RIS". La Habana, Universidad de las Ciencias Informáticas
- (11) **Sánchez Rodríguez, M.** (2010). " Infraestructura de software para el almacenamiento y consulta de la Historia Clínica Electrónica del sistema alas HIS". La Habana, Universidad de las Ciencias Informáticas
- (12)Ídem a la referencia 6.
- (13)Ídem a la referencia 1.

- (14) **Alejo González, J. P.** (2005). "IHE, INTEGRANDO LA EMPRESA DE SALUD". Número 5 - I Trimestre Vol. 2.
- (15) **SEFM**, S. E. d. F. M. from www.sefm-ihe.info.
- (16) **Borges Cabrera, Y. y González Díaz, R** (2010). "Arquitectura de un Sistema para la Transmisión de Información Radiológica Inter-Hospitalaria". La Habana, Universidad de las Ciencias Informáticas.
- (17) Ídem a la referencia 16.
- (18) Ídem a la referencia 16.
- (19) **Salud, N. d. I.** from www.worldlingo.com/ma/enwiki/es/Health_Level_7
- (20) **Junta de Castilla y León.** (2010). "Elementos comunes de mensajería".
- (21) **Alma.** "Alma3D." from <http://www.alma3d.com/es/productos/integracion-hisrispacs-3d>.
- (22) Ídem a la referencia 21.
- (23) Ídem a la referencia 21.
- (24) **González López, D. y Fernández Orozco, A** (2010). "Arquitectura de un sistema para la edición de informes de estudios imagenológicos". La Habana, Universidad de las Ciencias Informáticas.
- (25) Ídem a la referencia 24.
- (26) Ídem a la referencia 24.
- (27) **Noemalife.** from http://www.dominion.es/Portals/0/images/new/Galileo_Brochure_02_09_ESP.pdf.
- (28) Ídem a la referencia 27.
- (29) Ídem a la referencia 27.
- (30) **Serrano, C. M.** "Sistemas de información radiológica." From http://www.geyseco.es/sociedades/segeca/descargas/TH237_CMartinez.pdf.

(31) Ídem a la referencia 27.

(32) **Corporation, I.** from www.intersystems.es/media/media_manager/pdf/2867.pdf.

(33) Ídem a la referencia 32.

(34) Ídem a la referencia 32.

(35) **Carestream, K.** from http://www.imadine.com.mx/pdf/CS_RISbro_6790-7135_AUG03_hi-res_es-es.pdf.

(36) Ídem a la referencia 35.

(37) Ídem a la referencia 35.

(38) **Font Hernández, A. y Gómez Suárez, Y** (2008). "Alas PACS Client. Sistema para la gestión de estudios imagenológicos". La Habana, Universidad de las Ciencias Informáticas

(39) **Gamboa Yero, A. J. y Guerra Pérez, A** (2009). "AlasPACSWorklist Servidor de listas de trabajo". . La Habana, Universidad de las Ciencias Informáticas.

(40) Ídem a la referencia 39.

(41) Ídem a la referencia 39.

(42) **Microsoft Corporation., N. F. C. O.** from <http://msdn.microsoft.com/en-us/library/zw4w595w.aspx>.

(43) **Carrau, I. J.** "Pruebas de conectividad y soluciones de integración para sistemas de salud". from <http://www.hl7elc.org/ihic2010/24.pdf>.

(44) Ídem a la referencia 43.

(45) **Microsoft.** "Acerca de Visual Studio." from <http://msdn.microsoft.com/es-es/vstudio/products/bb931214.aspx>.

(46) Ídem a la referencia 45.

- (47) **Averoff Cabrera, D. y Enriquez Pérez, M. L.** (2010). "Diseño de una herramienta para la gestión del otorgamiento de licencias para el sistema alas PACSViewer". La Habana, Universidad de las Ciencias Informáticas.
- (48) **Garbey Bermudes, Y. M. y Torres Rodríguez, F.** (2009). Módulo Laboratorio del Sistema de Información Hospitalaria alas HIS. La Habana, Universidad de las Ciencias Informáticas.
- (49) Ídem a la referencia 48.
- (50) Ídem a la referencia 48.
- (51) **Reina Arzuaga. y. Gómez Sosa, M.** (2009). Plataforma de Gestión de Contenidos para Dispositivos Móviles. Módulo Entrega de Contenido. La Habana, Universidad de las Ciencias Informáticas.
- (52) Ídem a la referencia 51.
- (53) **Pedroso Batista,G y Páez Padrón, M.** (2010). "Análisis y diseño de un sistema de gestión de libros de texto impresos para el área de Abastecimiento Técnico Material (ATM)". La Habana, Universidad de las Ciencias Informáticas.
- (54) **Larman, C.**(2002) "UML y patrones".
- (55) **UML.** from http://www.neuronsrl.com.ar/training/uml/uml_clases.html.
- (56) Ídem a la referencia 51.
- (57) **Fernández Vila, A.** [En línea] 20 de marzo de 2001. [Citado el: 29 de abril de 2009.] <http://tvdi.det.uvigo.es/~avilas/UML/node50.html>.
- (58) **UML.** From <http://www.dsi.uclm.es/asignaturas/42530/pdf/M2tema12.pdf>
- (59) **Almaguer, I. N. M.** (2011). Documento de Arquitectura de Software.
- (60) Ídem a la referencia 59.

BIBLIOGRAFÍA

Alegre, L. (2005). *RevistaeSalud*. Número 5. Vol. 2.

Alejo González, J. P. (2005). "IHE, INTEGRANDO LA EMPRESA DE SALUD". Número 5 - I Trimestre Vol. 2.

Alma. "Alma3D." from <http://www.alma3d.com/es/productos/integracion-hisrispacs-3d>.

Almaguer, I. N. M. (2011). Documento de Arquitectura de Software.

Averoff Cabrera, D. y Enriquez Pérez, M. L. (2010). "Diseño de una herramienta para la gestión del otorgamiento de licencias para el sistema alas PACSViewer". La Habana, Universidad de las Ciencias Informáticas.

Borges Cabrera, Y. y González Díaz, R (2010). "Arquitectura de un Sistema para la Transmisión de Información Radiológica Inter-Hospitalaria". La Habana, Universidad de las Ciencias Informáticas.

Carestream, K. from http://www.imadine.com.mx/pdf/CS_RISbro_6790-7135_AUG03_hi-res_es-es.pdf.

Carrau, I. J. "Pruebas de conectividad y soluciones de integración para sistemas de salud". from <http://www.hl7elc.org/ihic2010/24.pdf>.

Corporation, I. from www.intersystems.es/media/media_manager/pdf/2867.pdf.

Facmed. from <http://www.facmed.unam.mx/emc/computo/ssa/HIS/hisindex.htm>.

Fernández Vila, A. [En línea] 20 de marzo de 2001. [Citado el: 29 de abril de 2009.] <http://tvdi.det.uvigo.es/~avilas/UML/node50.html>.

Font Hernández, A. y Gómez Suárez, Y (2008). "Alas PACS Client. Sistema para la gestión de estudios imagenológicos". La Habana, Universidad de las Ciencias Informáticas

Gamboa Yero, A. J. y Guerra Pérez, A (2009). "AlasPACSWorklist Servidor de listas de trabajo". La Habana, Universidad de las Ciencias Informáticas.

Garbey Bermudes, Y. M. y Torres Rodríguez, F. (2009). Módulo Laboratorio del Sistema de Información Hospitalaria alas HIS. La Habana, Universidad de las Ciencias Informáticas.

García González, D. y Socias Pardo, J. (2010). "Arquitectura para el sistema de almacenamiento de informes de estudios imagenológicos". La Habana, Universidad de las Ciencias Informáticas.

González Abreu, L. E. y Fonseca Guzman, M (2010). "Módulo de gestión para los reportes estadísticos de alas RIS". La Habana, Universidad de las Ciencias Informáticas.

González López, D. y Fernández Orozco, A (2010). "Arquitectura de un sistema para la edición de informes de estudios imagenológicos". La Habana, Universidad de las Ciencias Informáticas.**HL7.** from <http://www.hl7.org/>.

HL7. from <http://www.hl7.org/>.

Junta de Castilla y León. (2010). "Elementos comunes de mensajería".

Larman, C.(2002) "UML y patrones".

Martínez Carrillo, F. from www.unipamplona.edu.co/.../portallG/.../diseno_desarrollo_pacs.pdf.

Martínez Martínez, A. from www.itescam.edu.mx/principal/sylabus/fpdb/recursos/r43242.PDF.

Microsoft. "Acerca de Visual Studio. ." from <http://msdn.microsoft.com/es-es/vstudio/products/bb931214.aspx>.

Microsoft Corporation., N. F. C. O. from <http://msdn.microsoft.com/en-us/library/zw4w595w.aspx>.

Noemalife. from http://www.dominion.es/Portals/0/images/new/Galileo_Brochure_02_09_ESP.pdf.

Pedroso Batista,G y Páez Padrón, M. (2010). "Análisis y diseño de un sistema de gestión de libros de texto impresos para el área de Abastecimiento Técnico Material (ATM)". La Habana, Universidad de las Ciencias Informáticas.

Reina Arzuaga. y. Gómez Sosa, M. (2009). Plataforma de Gestión de Contenidos para Dispositivos Móviles. Módulo Entrega de Contenido. La Habana, Universidad de las Ciencias Informáticas.

Rivero Castro, A. y Hernández Noguera, A. (2010). " Aplicación para el registro de estudios imagenológicos. Solución para equipos de Oftalmología " La Habana, Universidad de las Ciencias Informáticas.

Rodríguez López, M. y Rodríguez. García, R (2010). "Propuesta de aplicación de los perfiles de integración de IHE entre los sistemas alas PACS–alas RIS–alas HIS.". La Habana, Universidad de las Ciencias Informáticas.

Sánchez Rodríguez, M. (2010). " Infraestructura de software para el almacenamiento y consulta de la Historia Clínica Electrónica del sistema alas HIS". La Habana, Universidad de las Ciencias Informáticas.

Salud, N. d. I. from www.worldlingo.com/ma/enwiki/es/Health_Level_7.

SEFM., S. E. d. F. M. from www.sefm-ihe.info.

Serrano, C. M. "Sistemas de información radiológica." from http://www.geyseco.es/sociedades/segeca/descargas/TH237_CMartinez.pdf.

UML. from http://www.neuronsrl.com.ar/training/uml/uml_clases.html.

UML. From <http://www.dsi.uclm.es/asignaturas/42530/pdf/M2tema12.pdf>.

GLOSARIO DE TÉRMINOS

A

APIs: Es un conjunto de convenciones internacionales que definen cómo debe invocarse una determinada función de un programa desde una aplicación.

A.C.C: American Collage of Cardiology.

ATNA: Audit Trail and Node Authentication.

D

Diagramas: Facilitan el entendimiento de grandes cantidades de datos y la relación entre diferentes partes de los datos.

H

Hardware: Incluye todas las partes físicas del computador, es decir, aquellos dispositivos que se conectan entre sí para formar una sola unidad de trabajo.

HDF: Entorno de desarrollo de aplicaciones en salud.

H.I.M.S.S: Healthcare Information and Managment Systems Society.

I

IDE: Entorno de Desarrollo Integrado.

Interoperabilidad: Es la capacidad de compartir información, datos y servicios entre los distintos Sistemas Informáticos.

Integración: Es una estructura compuesta de ordenadores o Sistemas Informáticos de distintos tipos y procedencias que se relacionan entre sí de manera transparente.

M

MINSAP: Ministerio de Salud Pública.

O

OEM: Original Equipment Manufacturer.

OSI: Interconexión de Sistemas Abiertos.

P

Perfil de Integración IHE: Describen las necesidades clínicas de integración y la solución para llevarlas a cabo.

R

RIM: Reference Information Model.

R.S.N.A: Radiological Society of North America.

S

Software: Se refiere al equipamiento lógico o soporte lógico de un computador digital.

T

TIC: Tecnologías de la información y las Comunicaciones.

W

Web Services: Es un sistema de software diseñado para apoyar interoperable máquina a máquina la interacción a través de una red.

X

XML: Extensible Markup Language (lenguaje extensible de etiquetas) Es un meta-lenguaje que permite definir lenguajes de marcado adecuado a usos determinados. Su función principal es describir datos y no mostrarlos.