



UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS

FACULTAD 7

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

**Propuesta de infraestructura de software para la gestión
de la Historia Clínica Electrónica centralizada**

Autores: Yisel Taimi Ladrón de Guevara Chala
Rubén Miranda Silva

Tutores: Ing. Kenia Fernández Parra
Ing. Gerardo Morgade Donato

Ciudad de La Habana, Julio de 2010

“Año 52 de la Revolución”

Datos de Contacto

Ing. Kenia Fernández Parra

Graduada en el año 2007 de Ingeniero en Ciencias Informáticas en la Universidad de las Ciencias Informáticas. Posee categoría docente de Instructor. Ha impartido las asignaturas Matemática Discreta, Gestión de Software, Sistemas de Bases de Datos, Ética Informática y Metodología de la Investigación Científica. Se desempeñó por un periodo de 6 meses como líder del área temática productiva Gestión Hospitalaria (GEHOS). Actualmente se desempeña como analista en el proyecto Sistema de Información Hospitalaria perteneciente al Departamento Gestión Hospitalaria del Centro de Informática Médica (CESIM).

Correo electrónico: kfernandezp@uci.cu

Ing. Gerardo Morgade Donato

Graduado en el año 2009 de Ingeniero en Ciencias Informáticas en la Universidad de las Ciencias Informáticas. Ha impartido las asignaturas Matemática III y Desarrollo de aplicaciones empresariales con Java y PostgreSQL. Actualmente se desempeña como Jefe de Módulo en el proyecto Sistema de Información Hospitalaria perteneciente al Departamento Gestión Hospitalaria del Centro de Informática Médica (CESIM).

Correo electrónico: gmorgade@uci.cu

Resumen

Actualmente la información clínica se encuentra distribuida en varias localizaciones. El personal sanitario que atiende al paciente no dispone de los datos completos de la Historia Clínica. En el presente trabajo se realiza una propuesta de infraestructura de software para la gestión de la Historia Clínica Electrónica centralizada.

Los Sistemas de Información que manipulan la Historia Clínica Electrónica como el HIS, el RIS y el PACS se caracterizan por trabajar de manera independiente, no son interoperables ni se encuentran integrados. Además no existe un Centro de Datos que se mantenga actualizado y almacene y comparta la información del paciente. El objetivo del presente trabajo es desarrollar los procesos de comunicación entre los Sistemas de Información y el Centro de Datos, que garantice el acceso a la información asociada y la unicidad de los datos clínicos del paciente.

El desarrollo de la solución propuesta está guiado por el Proceso Unificado de Desarrollo, apoyándose en la notación BPMN. Se basa en tecnologías libres, multiplataforma y con una arquitectura orientada a servicios, empleándose la tecnología *Message Oriented Middleware*. Se utiliza Java como lenguaje de programación, PostgreSQL como Sistema de Gestión de Base de Datos, como servidor de aplicaciones JBoss AS y Apache Solr para la gestión de índices.

Entre los beneficios que aporta la solución se encuentran que el personal médico podrá contar con la información completa del paciente, aumentando la calidad de la atención brindada y la toma de mejores decisiones a nivel médico y administrativo.

Palabras Claves:

Centro de Datos, Historia Clínica, Historia Clínica Electrónica, interoperabilidad, procesos de comunicación, Sistemas de Información.

Tabla de Contenidos

Introducción.....	1
Capítulo 1. Fundamentación Teórica.....	5
1.1 Conceptos fundamentales relacionados con el dominio del problema	5
1.1.1 Historia clínica.....	5
1.1.2 Historia clínica electrónica	5
1.1.3 Centro de Datos	5
1.1.4 Paciente.....	5
1.1.5 Médico.....	6
1.1.6 Institución hospitalaria.....	6
1.1.7 Proceso de atención al paciente.....	6
1.2 Situación problemática.....	6
1.3 Sistemas automatizados existentes.....	7
1.3.1 HP-HCIS.2 (Health Care Information System).....	7
1.3.2 Microsoft HealthVault.....	8
1.3.3 Google Health	9
1.4 Arquitectura de Software.....	9
1.4.1 Arquitectura Orientada a Servicios.....	10
1.4.2 Message Oriented Middleware	11
1.5 Metodología de desarrollo	13
1.5.1 RUP	13
1.6 Tecnologías de desarrollo.....	14
1.6.1 Java	14
1.6.2 Capa de servicio.....	15
1.6.3 Capa de modelo	16
1.7 Servidor de aplicaciones	17
1.7.1 JBoss AS.....	17
1.8 Tecnologías de intercambio.	18
1.8.1 Servicios web.....	18
1.9 Gestor de base de datos.....	19
1.9.1 PostgreSQL.....	19
1.10 Estándares médicos	20
1.10.1 HL7	21
1.10.2 HL7- CDA.....	21
1.10.3 HL7- IHE.....	22
1.11 Herramientas.....	22
1.11.1 Visual Paradigm para UML.....	22
1.11.2 Eclipse.....	23
1.11.3 PgAdmin	23

1.11.4 Jboss Tools	24
1.11.5 Apache Solr.....	24
Capítulo 2. Características de la solución	26
2.1 Sistemas que van a interoperar	26
2.1.1 APS.....	26
2.1.2 HIS.....	27
2.1.3 RIS.....	27
2.1.4 PACS.....	27
2.2 Perfiles de Integración	29
2.2.1 Patient Identifier Cross-referencing (PIX).....	30
2.2.2 Cross Enterprise Document Sharing (XDS).....	30
2.2.3 Query for Existing Data (QED).....	31
2.3 Common Terminology Services (CTS)	32
2.4 Modelo de Dominio	33
2.5 Flujo de procesos	37
2.5.1 Flujo de procesos del PIX	39
2.5.2 Flujo de procesos del XDS	40
2.5.3 Flujo de procesos del QED.....	41
2.6 Requisitos del sistema.....	45
2.6.1 Requisitos funcionales	45
2.6.2 Requisitos no funcionales.....	46
2.6.3 Modelo de casos de uso del sistema.....	46
Capítulo 3. Análisis y diseño.....	49
3.1 Diagrama de Clases del Diseño.....	49
3.1.1 Descripción de las clases del diseño	49
3.2 Diagrama de Interacción	51
Capítulo 4. Implementación.....	54
4.1 Modelo de Despliegue	54
4.2 Tratamiento de errores	57
4.3 Seguridad.....	57
4.4 Estrategias de codificación. Estándares y estilos a utilizar	58
Conclusiones.....	60
Recomendaciones.....	61
Referencias Bibliográficas	62
Bibliografía.....	64
Glosario de Términos	68
Anexos	70

Índice de Figuras

Figura 2.1 Integración HIS-RIS-PACS.....	28
Figura 2.2 Modelo de dominio de Centro de Datos.....	36
Figura 2.3 Diagrama general de procesos del Centro de Datos.....	38
Figura 2.4 Diagrama de proceso Indexar documentos.....	42
Figura 2.5 Diagrama de proceso Consultar documentos.....	43
Figura 2.6 Diagrama de proceso Continuar consulta.....	44
Figura 2.7 Diagrama de casos de uso del sistema. Buscar documento.....	47
Figura 3.1 Diagrama de Clases del Diseño_Indexar documento.....	50
Figura 3.2 Diagrama de Clases del Diseño_Buscar documento.....	51
Figura 3.3 Diagrama de Secuencia_Consultar Documento por Enfermedad.....	52
Figura 3.4 Diagrama de Secuencia_Consultar Documento por Medicamento.....	52
Figura 4.1 Modelo de Despliegue.....	55

Introducción

La Historia Clínica (HC) del paciente es un registro que se desarrolla con el tiempo y documenta la relación médico-paciente. Tiene como objetivo fundamental recoger datos del estado de salud del mismo con el fin de facilitar la asistencia médica. Es el instrumento básico para el buen ejercicio sanitario porque sin ella es casi imposible que el médico pueda hacer un seguimiento correcto de la evolución de sus pacientes.

La HC convencional o en papel presenta diversas dificultades que se han hecho evidentes en la práctica diaria. Como son la presencia de una escritura pobre, ilegible e incompleta que dificulta la interpretación del mensaje que se quiere enviar. Además problemas de almacenamiento, gestión y etiquetado de los datos clínicos. Así como deterioro o pérdida y al encontrarse en formato duro, sólo puede existir en un lugar y un momento dado, provocando que la información no pueda ser localizada cuando se necesita.

Debido a los inconvenientes de almacenamiento, para facilitar su conservación varias instituciones de nuestro país han optado por la HC ambulatoria. En este caso el paciente es quien se encarga de guardar dicha historia. Lamentablemente así este documento pierde sus características docentes y legales y mantiene además todas las dificultades de la HC convencional.

Con la llegada de las computadoras comenzaron a desarrollarse pequeños sistemas informáticos que contribuían al análisis y procesamiento de datos e información. Se informatizaron procesos manuales con el objetivo de facilitar el trabajo de las personas involucradas en estos y distribuir la información de la forma más adecuada posible. La medicina no ha quedado exenta del creciente desarrollo científico y tecnológico, y en la década del '70 surgen los primeros sistemas informáticos enfocados en esta área. Estos sistemas sirven de apoyo a la asistencia de pacientes, determinación de diagnóstico y tratamientos, investigaciones científicas, procesamiento de estadísticas y apoyo a la docencia entre otros.

La creciente demanda de información adecuadamente estructurada, en combinación con la evolución de las ciencias informáticas, han permitido el surgimiento y desarrollo de la Historia Clínica Electrónica (HCE). Esta presenta indudables ventajas frente a la HC convencional. Soluciona problemas como la legibilidad, accesibilidad y estructura de la información. Además permanece inalterable en el tiempo para que su información pueda ser consultada cuando se necesite, requiere poco espacio de almacenamiento,

evita la duplicidad de pruebas, disminuye los tiempos de espera y posibilita el acceso simultáneo de varios profesionales, permitiendo llevar un mejor seguimiento del proceso asistencial.

Además de la informatización de la Historia Clínica del paciente, la inclusión de la informática en el área de la salud, propició el surgimiento de sistemas de información. Hoy en día estos cuentan con grandes tecnologías y software, que solucionan muchos problemas, desde los más sencillos hasta los más complejos. Apoyan el trabajo de los profesionales médicos y permiten mejorar los servicios que se le brindan al paciente.

Estos sistemas sobre los que se apoya el desarrollo de la asistencia sanitaria, como el HIS, el RIS y el PACS, en caso de estar presentes en una entidad, al carecer de interoperabilidad, no permiten el intercambio de información entre ellos, ocasionando la existencia de HCE duplicadas, lo cual dificulta la vista única del paciente. La ausencia de comunicación provoca que no se pueda acceder a los datos asociados al paciente en cada consulta efectuada en diferentes instituciones. Los mismos se encuentran fragmentados y desactualizados, por lo que no se cuenta con la información completa del mismo.

Para poder garantizar una asistencia eficiente y de calidad es necesario que dichos sistemas sean interoperables. De manera que se pueda compartir información y que esta sea legible y tratable por los diferentes sistemas con los que operan los profesionales de la salud.

En la actualidad, la situación habitual es que la información clínica esté distribuida en varias localizaciones, debido a que cada centro sanitario ya sea un hospital, un centro de atención primaria o una clínica mantiene la suya propia. Sin embargo, el personal sanitario que atiende a un paciente necesita acceder a información como diagnósticos y tratamientos previos, aunque estas residan en bases de datos ubicadas en diferentes localizaciones físicas e incluso en diferentes organizaciones sanitarias. Actualmente, la carencia de un marco de interoperabilidad fiable se considera como el mayor obstáculo para el intercambio de registros sanitarios electrónicos.

Aparejado a esto no existe un repositorio central de datos clínicos donde se almacene y se comparta la información relacionada al paciente, que se mantenga permanentemente actualizado y disponible con la seguridad requerida para el personal médico autorizado, que permita consultar la información correcta en el lugar adecuado y en el momento oportuno.

Otro punto a tener en cuenta es la complejidad de los términos médicos. En ocasiones estos son difíciles de recordar. Actualmente se han desarrollado diferentes terminologías para el sector de la salud. Sin embargo, la diversificación de estas dificulta la comunicación entre el personal médico.

Producto a esto surge la necesidad de crear un Centro de Datos para la HCE centralizada que garantice la unicidad de los datos clínicos del paciente.

Teniendo en cuenta lo antes analizado se plantea como **problema a resolver**: ¿Cómo garantizar un registro único de los datos clínicos del paciente?

El **objeto de estudio** lo constituyen los procesos relacionados a la atención médica del paciente, cuyo **campo de acción** se enmarca en los registros de información clínica generados como resultado de los procesos de atención médica del paciente.

Basado en esa idea se define como **objetivo general** del presente trabajo: Desarrollar los procesos de comunicación entre sistemas médicos y el centro de datos, así como el acceso a la información asociada, que garantice el registro único de los datos clínicos del paciente.

Para darle cumplimiento al objetivo propuesto, así como controlar y evaluar el proceso investigativo, se proponen como **tareas de la investigación**:

1. Evaluar las tendencias actuales en el mundo de los centros de datos relacionados a la atención médica.
2. Analizar los procesos de comunicación entre los sistemas de información médica y un centro de datos.
3. Analizar la arquitectura definida por el Departamento de Sistemas de Gestión Hospitalaria para el desarrollo de sus aplicaciones.
4. Obtener mediante el Proceso Unificado de Desarrollo, los artefactos generados en los flujos de trabajo de “Modelo de Negocio”, “Gestión de Requerimientos”, “Análisis y Diseño” e “Implementación”.
5. Implementar el perfil de integración *Query for Existing Data* (QED).
6. Diseñar la infraestructura tecnológica del Centro de Datos para la Historia Clínica Electrónica única del paciente.

El presente trabajo se ha estructurado en cuatro capítulos:

Capítulo 1. Fundamentación teórica: En él se analizan algunos sistemas informatizados similares al que se desea desarrollar con el propósito de lograr un mejor entendimiento del problema a resolver, además se muestran las tecnologías y software que serán utilizados.

Capítulo 2. Características de la solución: Se confecciona el modelo de dominio, se describen los procesos del sistema que serán implementados, se definen los requisitos tanto funcionales como no funcionales y se realiza una propuesta de la solución.

Capítulo 3. Análisis y diseño: Se elaboran los artefactos correspondientes a al Flujo de trabajo Análisis y Diseño, perteneciente a la metodología de desarrollo RUP y se describen las clases definidas del diseño.

Capítulo 4. Implementación: Se describe el Modelo de Despliegue, que constituye la distribución física de los componentes del sistema, así como los nodos que lo componen. Además se exponen los estándares y estilo de codificación que serán utilizados.

Capítulo 1. Fundamentación Teórica

Introducción

El presente capítulo tiene como objetivos fundamentales profundizar en la situación problemática y explicar conceptos básicos relacionados con el problema a resolver. Además se analizan los sistemas informatizados existentes relacionados con la problemática actual, que permitan realizar un mejor diseño de la solución propuesta. También se realiza un estudio de las tendencias y tecnologías actuales, metodologías y herramientas que serán utilizadas y la justificación de cada una de ellas.

1.1 Conceptos fundamentales relacionados con el dominio del problema

1.1.1 Historia clínica

Conjunto de documentos surgidos de la relación entre el médico y el paciente. Se considera como el único documento válido desde los puntos de vista clínico y legal. Es el eslabón principal en el sistema de información hospitalario, imprescindible en sus vertientes asistencial y administrativa.

1.1.2 Historia clínica electrónica

Incorpora las Tecnologías de la Información y las Comunicaciones (TIC) en el núcleo de la actividad sanitaria. Esto trae como consecuencias que la historia clínica deje de ser un registro generado entre un paciente y un profesional médico, para formar parte de un sistema integrado de información clínica.

1.1.3 Centro de Datos

Conjunto de servidores, oficina o departamento que se encarga del procesamiento de datos e información de forma sistematizada. Son grandes sistemas de información que contienen computadoras interconectadas en red y realizan 4 actividades fundamentales: la entrada de datos, su almacenamiento, procesamiento y salida.

1.1.4 Paciente

Es el sujeto que recibe los servicios de un médico u otro profesional de la salud, sometiéndose a un examen, un tratamiento o a una intervención. Persona que va a ser reconocida médicamente, que sufre dolor o malestar. Es el agente pasivo en el proceso sanitario.

1.1.5 Médico

Persona que se encuentra legalmente autorizada para ejercer la medicina. Recoge información sobre el estado del paciente mediante las distintas herramientas, realiza un análisis de la misma, obtiene un diagnóstico y desarrolla un programa de tratamiento y formación del paciente respecto a su propia salud. Es el agente activo en el proceso sanitario.

1.1.6 Institución hospitalaria

Establecimiento destinado a la prevención, diagnóstico y tratamiento de pacientes, donde se practican también la investigación y la enseñanza.

1.1.7 Proceso de atención al paciente

Servicios prestados por miembros de las profesiones sanitarias en beneficio del paciente.

1.2 Situación problemática

A la implantación de la HCE se le atribuye la innegable capacidad de mejorar la atención clínica del paciente. La creación de esta se efectúa en cada institución, por el personal médico que allí labora, no existiendo comunicación entre los distintos sistemas de información pertenecientes a las diferentes instituciones hospitalarias.

El paciente puede ser atendido en varias instituciones y los documentos generados como resultado de la atención radican en su mayoría, en los departamentos donde fue atendido previamente el paciente. Esto dificulta el proceso de consulta de la información por parte del médico, el cual no podrá disponer de la información completa de los datos clínicos del paciente.

A pesar de las ventajas que traería la implantación de la HCE y la existencia del Centro de Datos para la comunicación de las diferentes instituciones de salud existen dificultades como por ejemplo:

- La ausencia de un modelo igual para todos los sistemas, que estandarice la estructura de la información de la HCE, para que todo el equipo médico que interactúe con ella la perciba igual cada vez y el proceso de consulta de la información clínica en realidad apoye y facilite el trabajo del médico.

- Ausencia de estándares de comunicación entre los distintos sistemas de salud que existen, que les faciliten a los profesionales compartir información en distintos lugares y al mismo tiempo, sin tener en cuenta el tipo de sistema con que están interactuando y las peculiaridades de su instalación.
- La HCE puede ser creada, consultada y modificada por trabajadores médicos en diferentes situaciones (diagnóstico, prescripción farmacéutica, indicación de exámenes complementarios) y no existen reglas o políticas de acceso con las limitaciones apropiadas en cada caso, que regulen quién y bajo cuáles circunstancias pueden introducir cambios en los registros médicos y velen por aspectos tan importantes como la confidencialidad, privacidad y seguridad de los datos clínicos.
- La terminología médica cambia constantemente y al no existir un sistema que las estandarice se dificulta la comunicación entre los profesionales médicos.

1.3 Sistemas automatizados existentes

En la actualidad, en el mundo no existen muchos sistemas que cuenten con un centro de datos para la creación, consulta, modificación y almacenamiento de la HCE. Algunos ejemplos son: HP-HCIS.2 (Health Care Information System), Microsoft HealthVault y Google Health.

1.3.1 HP-HCIS.2 (Health Care Information System)

La empresa Hewlett-Packard, también conocida como HP, es proveedora de equipos de alta tecnología para negocios y para la vida. Las soluciones de la compañía estadounidense abarcan la infraestructura de tecnologías de la información, dispositivos de acceso, de informática personal y servicios globales destinados a consumidores, grandes compañías y a la pequeña y mediana empresa.

La HP, a través de su división de sanidad, ha implementado un sistema de informatización del historial médico del paciente, mediante la implantación de la HCE. HP-HCIS.2 (Health Care Information System) es un sistema web y de licencia propietaria que permite que la historia del paciente esté adecuadamente archivada, categorizada y accesible de forma instantánea para todo el personal autorizado. Al mismo se puede acceder desde cualquier departamento y computadora personal (PC) en cualquier hospital, lo cual traerá consigo una mayor rapidez de transmisión de la información y grandes beneficios para el paciente.

La implantación de la HCE por parte de la HP ha reportado una serie de ventajas:

- Garantiza la automatización del proceso de atención al paciente.
- Evita la pérdida de información y la duplicidad de la misma.
- Disminuye el margen de errores.
- Facilita el acceso a los datos clínicos.
- Garantiza la confidencialidad y una notable reducción de la presencia de papeles.

De manera general facilita el trabajo del personal médico y aumenta la calidad del servicio brindado al paciente.

1.3.2 Microsoft HealthVault

Microsoft Corporation es una empresa multinacional estadounidense dedicada al sector de la informática. Desarrolla, fabrica, licencia y produce software y equipos electrónicos. También ha incursionado en el área de la salud, sacando al mercado la plataforma Microsoft HealthVault.

Microsoft HealthVault es un sistema propietario que propone a sus clientes el control total de su información de salud, que pueda ser fácilmente compartida y actualizada. La misma se encuentra almacenada en un lugar central y con una seguridad mejorada y flexible.

Esta plataforma les permite a los pacientes almacenar copias de sus registros de salud en cuentas de fácil acceso. Además posibilita a sus usuarios cargar su información de forma física en diferentes dispositivos y proporcionar sus datos a especialistas médicos y entrenadores, logrando así que la información del cliente esté permanentemente actualizada y centralizada. Microsoft HealthVault es un proveedor de soluciones web para hospitales, asociaciones de salud, laboratorios y centros de imágenes, fabricantes de dispositivos, empresarios, farmacias, planes de salud, médicos y proveedores de aplicaciones.

La puesta en práctica de Microsoft HealthVault en los hospitales permite agilizar el proceso de admisión de pacientes y descarga de datos de los mismos. Los laboratorios pueden personalizar los servicios mediante la entrega de resultados al registro de un paciente HealthVault. En las farmacias la plataforma le permite al paciente gestionar su receta y la información referente a los medicamentos, y así proporcionar

esta información a su personal médico. Los planes de salud de HealthVault le ayudan al cliente a manejar situaciones crónicas, a prevenir enfermedades así como hacer elecciones de vida más sana.

1.3.3 Google Health

Google Inc. es la empresa propietaria de la marca Google, cuyo principal producto es el motor de búsqueda del mismo nombre. Fue fundada el 4 de septiembre de 1998 por Larry Page y Sergey Brin. En el año 2008 la compañía se adentra en el complicado y lucrativo mundo de la información de salud, lanzando su producto Google Health. Esta es una aplicación web completamente gratis a la cual se accede mediante el nombre de usuario y contraseña de la cuenta de correo electrónico de *Gmail*, desde cualquier lugar y en cualquier momento.

Google Health permite gestionar toda la información de salud y almacenarla en un lugar central. Además brinda la posibilidad de construir perfiles de salud, tanto propios como de familiares. Permite importar registros médicos de hospitales y farmacias, explorar servicios de salud en línea y buscar médicos por nombre y localidad.

También permite compartir el perfil de forma segura y confiable con otras personas que sólo podrán verlo, pero no tendrán permiso para cambiar nada. Posibilita que el usuario decida cuánto desea compartir, con quién y en qué momento. Con Google Health se puede mantener el día la información de salud, evitando la pérdida de la misma y el llenado de documentos cada vez que se visite un nuevo médico.

Estos sistemas pertenecen a compañías estadounidenses multimillonarias de licencia propietaria, en su mayoría costosos. Además no brindan información sobre sus principales funcionalidades, lo que impide hacer una correcta valoración de su factibilidad. Es por esto que se propone la creación de un Centro de Datos para Historia Clínica Electrónica centralizada que de solución a la problemática actual.

1.4 Arquitectura de Software

Hoy en día en todo proceso de desarrollo de software es de vital importancia que exista una estrecha relación entre las tecnologías y herramientas que serán utilizadas en la creación del mismo. Dicha relación permite lograr una mayor portabilidad, escalabilidad, seguridad y capacidad para asimilar cambios sin

necesidad de realizar grandes transformaciones. La Arquitectura de Software es de gran ayuda en este sentido.

En la actualidad existen muchas definiciones de Arquitectura de Software, según la IEEE-Std-1471-200 se define como: (1)

“...la organización fundamental de un sistema encarnada en sus componentes, las relaciones entre ellos y el ambiente y los principios que orientan su diseño y evolución.”

La Arquitectura de Software es un conjunto de decisiones significativas sobre la organización de un sistema de software; la selección de elementos estructurales y sus interfaces a través de los cuales se constituye el sistema; su comportamiento, según resulta de las colaboraciones entre esos elementos; la composición de esos elementos estructurales y de comportamiento en subsistemas progresivamente mayores; el estilo arquitectónico que guía esta organización: los elementos estáticos y dinámicos y sus interfaces, sus colaboraciones y su composición. (2)

1.4.1 Arquitectura Orientada a Servicios

Es importancia que las entidades de salud puedan interconectar información, personas y procesos, tanto con la propia entidad como con otras externas. La falta de integración entre los sistemas, aplicaciones y datos, hacen difícil obtener una respuesta rápida y efectiva ante las necesidades de los usuarios.

La Arquitectura SOA (por sus siglas en inglés) establece un marco de diseño para la integración de aplicaciones independientes de manera que desde la red pueda accederse a sus funcionalidades, las cuales se ofrecen como servicios. La forma más habitual de implementarla es mediante Servicios Web, una tecnología basada en estándares e independiente de la plataforma. Un servicio es una funcionalidad concreta que puede ser descubierta en la red y que describe tanto lo que puede hacer como el modo de interactuar con ella.

SOA supone una estrategia general de organización de los sistemas y aplicaciones, de forma que un conjunto de sistemas distribuidos y aplicaciones complejas se pueda transformar en una red de recursos integrados, simplificada y flexible. En las Arquitecturas Orientadas a Servicios el elemento básico es el servicio. Pero con este único elemento no se podría diseñar una arquitectura SOA.

Para realizar un desarrollo basado en esta arquitectura son necesarios cuatro elementos:

- ✓ **Operación:** Es la unidad de trabajo o procesamiento en una arquitectura SOA.
- ✓ **Servicio:** Es un contenedor de lógica. Estará compuesto por un conjunto de operaciones, las cuales las ofrecerá a sus usuarios.
- ✓ **Mensaje:** Para poder ejecutar una determinada operación, es necesario un conjunto de datos de entrada. Una vez ejecutada la operación, esta devolverá un resultado. Los mensajes son los encargados de encapsular esos datos de entrada y de salida.
- ✓ **Proceso de negocio:** Son un conjunto de operaciones ejecutadas en una determinada secuencia (intercambiando mensajes entre ellas) con el objetivo de realizar una determinada tarea.

Una aplicación SOA estará formada por un conjunto de procesos de negocio. Dichos procesos estarán compuestos por aquellos servicios que proporcionan las operaciones que se necesitan ejecutar para que el proceso de negocio se realice correctamente. Para ejecutar esas operaciones se envían los datos necesarios mediante los correspondientes mensajes.

1.4.2 Message Oriented Middleware

Una de las tecnologías base para la interconexión son los sistemas MOM o *Message Oriented Middleware*. Esta es una infraestructura cliente/servidor que aumenta la interoperabilidad, la portabilidad, y la flexibilidad de una aplicación. Permite el intercambio de datos mediante mensajes y que la misma sea distribuida sobre plataformas múltiples heterogéneas.

La tecnología MOM cuenta con una capa de software intermedia que implementa una interfaz que se encarga de la comunicación y transmisión de mensajes entre las diferentes aplicaciones. Con esto se puede poner en práctica una SOA (Arquitectura Orientada a Servicios). Cada servicio es expuesto mediante una cola de mensajes de tipo FIFO (*First In First Out*), donde el primero que entra es el primero que sale. En la actualidad existen algunas tendencias de estandarización, donde las interfaces APIs son implementadas con JMS mientras que los protocolos de comunicación utilizan servicios web.

Entre las ventajas que aporta el uso de esta tecnología se encuentran:

- Apoya balanceos de carga y prioridad permitiendo la recuperación de mensajes de la cola en cualquier orden.
- Los clientes son libres de realizar otras operaciones mientras esperan una respuesta del servidor.
- MOM permite muchas respuestas a una petición.
- Los productos son diseñados para utilizar sistemas de herencias y sistemas extensamente dispersos.
- Es conveniente para aplicaciones con transacciones con largas vidas, tales como *workflows*.
- Los productos con MOM soportan una tolerancia de incidentes: las colas persistentes permiten que los mensajes sean recuperados cuando el sistema deja de funcionar.

MOM consta de 2 posibilidades arquitectónicas:

✓ **Arquitectura centralizada:**

Donde todo pasa por un servidor de mensajes (también conocido como *router* o *broker* de mensajes), que permite la comunicación de aplicaciones con bajo acoplamiento, haciéndolas independientes entre sí.

✓ **Arquitectura descentralizada:**

Implementa la transmisión *multicast/broadcast* lo cual aumenta la complejidad del cliente, ya que recibe información local periódica (como suscripciones y autorizaciones) acerca de otros clientes.

La tecnología MOM puede ser de 2 modelos:

✓ **Mensajería punto a punto (*Point to Point*):**

Es una mensajería de Petición/Respuesta, donde el Emisor realiza una petición, la misma se inserta en la Cola de Peticiones, el Receptor la saca de esta cola y genera una respuesta que es enviada a la Cola de Respuestas, de donde el Emisor deberá sacarla.

✓ **Mensajería por difusión o broadcasting de mensajes (*Publish/Subscribe*):**

Es una mensajería mediante publicación/suscripción donde intervienen Productores y Consumidores. Los Consumidores se suscriben a un topic o tema, los Productores publican mensajes en un tema determinado y los mensajes sobre un tema se envían a todos los consumidores suscritos.

Después de un estudio se ha definido utilizar MOM con una arquitectura centralizada empleando ActiveMQ como servidor de mensajes, que brinde servicios como el ruteo, transformación y persistencia de mensajes. Implementándose los modelos *Point to Point* y *Publish/Subscribe*, en dependencia de la funcionalidad requerida.

1.5 Metodología de desarrollo

1.5.1 RUP

RUP (*Rational Unified Process*) es un proceso de desarrollo de software que captura las mejores prácticas del conocimiento de líderes en ingeniería de software. Proporciona a los equipos de desarrollo guías, estándares y recomendaciones para la construcción de software de alta calidad. (3)

RUP tiene como características fundamentales que está dirigido por casos de uso, es iterativo e incremental y centrado en la arquitectura. Además es un proceso integrador y su desarrollo está basado en componentes. Utiliza UML (*Unified Modeling Language*) como lenguaje de modelado.

Está dividido en 4 fases: Inicio, Elaboración, Construcción y Transición y 9 flujos de trabajo, de los cuales 6 son de ingeniería y 3 son de apoyo.

RUP tiene entre otras ventajas: (4)

- Mitigación temprana de posibles riesgos altos.
- Progreso visible en las primeras etapas.
- Temprana retroalimentación que se ajuste a las necesidades reales.
- Gestión de la complejidad.
- EL conocimiento adquirido en una iteración puede aplicarse de iteración a iteración.

1.5.1.1 BPMN

Business Process Modeling Notation (BPMN) es una notación gráfica que describe la lógica de los pasos de un proceso de negocio. Esta notación ha sido especialmente diseñada para coordinar la secuencia de los procesos y los mensajes que fluyen entre los participantes de las diferentes actividades. (5)

Un proceso de negocio es una colección de actividades que tiene una o varias entradas y produce una salida o resultado de valor para el cliente. Es la forma en que se organiza y coordina el trabajo para producir un servicio valioso.

BPMN es un estándar internacional de modelado de procesos que se inserta en la fase de Modelo de Negocio, propuesto por RUP. Crea un puente para disminuir la brecha entre los procesos de negocio y su implementación, con el objetivo de lograr una mejor administración de los mismos. Esta notación permite modelar procesos de forma unificada y estandarizada y tiene como objetivo fundamental que las partes involucradas puedan comunicar los procesos de manera clara, completa y eficiente. Actualmente está en el mercado su versión 2.0 Beta 1.

Contiene distintos elementos que se combinan para formar Diagramas de Procesos de Negocio (*Business Process Diagram*, BPD) y se clasifican en 4 categorías: objetos de flujo, objetos de conexión, canales y artefactos. Los objetos de flujo son los principales elementos que definen el comportamiento de los procesos, dentro de estos están los eventos, actividades y compuertas. Los objetos de conexión son utilizados para conectar dos objetos del flujo dentro de un proceso. Los canales se usan para organizar las actividades del flujo y los artefactos proveen información adicional sobre el proceso.

1.6 Tecnologías de desarrollo

1.6.1 Java

En el mundo la tecnología Java se usa en los principales sectores de la industria y está presente en un gran número de dispositivos, equipos y redes.

Java es toda una tecnología orientada al desarrollo de software con el cual se puede realizar cualquier tipo de programa. Hoy en día, la tecnología Java ha cobrado mucha importancia en el ámbito de Internet gracias a su plataforma J2EE (*Java 2 Enterprise Edition*). Pero Java no se queda ahí, ya que en la industria para dispositivos móviles también hay una gran acogida para este lenguaje. (6)

La tecnología Java fue desarrollada por *Sun Microsystems* para aplicaciones de software independiente de la plataforma de software o hardware y está compuesta básicamente por 2 partes: el lenguaje de programación y la plataforma. La plataforma Java consta de 2 elementos: la máquina virtual de Java (JVM

por sus siglas en inglés) y el Java API (*Application Programming Interface*). Java 2EE incluye varias especificaciones de API, tales como JMS y Servicios Web.

La tecnología Java despliega una multitud de posibilidades para los usuarios, pues permite que prácticamente cualquier aplicación (ya sean juegos, herramientas o programas de información y servicios) se ejecute en casi cualquier equipo o dispositivo.

Java ha sido probado, mejorado y ampliado por una comunidad especializada de más de 6,5 millones de desarrolladores, la mayor y más activa del mundo. Gracias a su versatilidad, eficiencia y portabilidad, Java se ha convertido en un recurso inestimable ya que permite a los desarrolladores: (7)

- Desarrollar software en una plataforma y ejecutarlo en prácticamente cualquier otra plataforma.
- Crear programas para que funcionen en un navegador web y en servicios web.
- Desarrollar aplicaciones para servidores como foros en línea, tiendas, encuestas y procesamiento de formularios HTML.
- Combinar aplicaciones o servicios que usan el lenguaje Java para crear servicios o aplicaciones totalmente personalizados.
- Desarrollar potentes y eficientes aplicaciones para teléfonos móviles, procesadores remotos, productos de consumo de bajo coste y prácticamente cualquier tipo de dispositivo digital.

1.6.2 Capa de servicio

1.6.2.1 Seam

JBoss Seam es una nueva y potente infraestructura para desarrollar aplicaciones Web 2.0 de próxima generación, al unificar e integrar tecnologías como AJAX, *JavaServer Faces* (JSF), *Enterprise Java Beans* (EJB3), *Java Portlets*, *Business Process Management* (BPM), *Drools*, *Hibernate* y *JPA* en una única solución. Seam ha sido diseñado desde cero para eliminar la complejidad, en la arquitectura y los niveles de la API, permitiendo a los desarrolladores complejas aplicaciones web haciendo uso simple de anotaciones POJOs sin importar el tipo de componente.

Las aplicaciones Seam son conceptualmente simples y requieren significativamente menos código (en Java y en XML) para obtener las mismas funcionalidades. Añade herramientas muy útiles para el desarrollo de aplicaciones Web y está basado en estándares ampliamente utilizados y probados (escalables, portables y reusables). Soporta los contextos de página, aplicación, sesión, evento y conversación que le facilitan al desarrollador el control del ciclo de vida de la información implicada en la realización de una tarea. El más importante quizás, es el contexto de conversación, pues permite extender el ciclo de vida y el manejo de la información en un proceso dado en dependencia de la necesidad del programador.

1.6.2.2 Apache Mod_jk

Se trata de un *plugin* Tomcat-Apache precisamente para permitir la comunicación entre los servidores de aplicación Tomcat y Apache. Su utilización constituye una solución escalable y robusta para aplicaciones web. Permite el balanceo de carga de las aplicaciones distribuyendo las peticiones HTTP mediante la clusterización. Además ofrece la replicación de sesiones con el fin de permitir tolerancia ante las caídas de estas por ocurrir fallos en el servidor y así evitar la pérdida de las mismas.

1.6.3 Capa de modelo

1.6.3.1 Framework Hibernate

Hibernate es una herramienta para la plataforma Java que facilita el mapeo de atributos entre una base de datos relacional y el modelo de objetos de una aplicación, mediante archivos declarativos (XML) que permiten establecer estas relaciones. Hibernate es una herramienta ORM (*Object-Relational Mapping*) completa que mantiene una excelente reputación en la comunidad de desarrollo, destaca entre sus ventajas una buena documentación y estabilidad.

1.6.3.2 JPA (Java Persistence API)

Java Persistence API, más conocida por su sigla JPA, es la API de persistencia desarrollada para la plataforma Java EE, incluida en el estándar EJB3. Esta API busca unificar la manera en que funcionan las utilidades que proveen un mapeo objeto-relacional.

El mapeo objeto-relacional, es decir, la relación entre entidades Java y tablas de la base de datos, se realiza mediante anotaciones en las propias clases de entidad, por lo que no se requieren ficheros descriptores XML. También pueden definirse transacciones como anotaciones JPA.

1.7 Servidor de aplicaciones

Un servidor de aplicaciones es un dispositivo de software que proporciona servicios de aplicación a las computadoras clientes. Generalmente gestiona la mayor parte de las funciones de lógica del negocio y de acceso a datos de la aplicación. Uno de sus principales beneficios es la centralización y disminución de la complejidad en el desarrollo de aplicaciones.

1.7.1 JBoss AS

JBoss Application Server es el servidor de aplicaciones de código abierto más ampliamente desarrollado del mercado. Por ser una plataforma certificada J2EE, soporta todas las funcionalidades de J2EE 1.4, incluyendo servicios adicionales como *clustering*, *caching* y persistencia. JBoss es ideal para aplicaciones Java y aplicaciones basadas en la web. También soporta *Enterprise Java Beans* (EJB) 3.0, lo que hace que el desarrollo de las aplicaciones del empresario sea mucho más simple. (8)

JBoss es el servicio encargado de ejecutar aplicaciones Java en el servidor y devolver los datos correspondientes a las peticiones realizadas por el usuario. Es un producto de software libre y una solución de bajo coste frente a otras aplicaciones corporativas que realizan el mismo papel. (9)

JBoss permite crear, implementar, integrar, organizar y presentar aplicaciones y servicios web en una arquitectura orientada a servicios. Entre sus principales ventajas están:

- Disminuir la complejidad de desarrollo.
- Mejorar la experiencia del usuario final.
- Proporcionar una flexibilidad que no se puede comparar con la de sus competidores.
- Disminuir notablemente el coste total de su propiedad.

1.8 Tecnologías de intercambio.

1.8.1 Servicios web

Existen múltiples definiciones sobre lo que son los Servicios Web o *Web Service*, lo que muestra su complejidad a la hora de dar una adecuada definición que englobe todo lo que son e implican. Una posible sería hablar de ellos como un conjunto de aplicaciones o de tecnologías con capacidad para inter-operar en la Web. Estas aplicaciones o tecnologías intercambian datos entre sí con el objetivo de ofrecer unos servicios. Estos servicios proporcionan mecanismos de comunicación estándares entre diferentes aplicaciones, que interactúan entre sí para presentar información dinámica al usuario. (10)

Sus principales ventajas son:

- Proporciona interoperabilidad entre aplicaciones de software que se ejecutan en plataformas distintas.
- Permiten la lógica de negocio de muchos sistemas diferentes para ser expuestos en la web. Esto da a sus aplicaciones la libertad de elegir los servicios web que ellos necesitan, desarrollarlos nuevamente para cada cliente.
- Permiten fácilmente combinar software y servicios de diferentes empresas y lugares para conformar un nuevo y mejorado servicio integral.
- Permiten la reutilización de servicios y componentes dentro de una infraestructura.

1.8.1.1 Axis 2.0

Apache Extensible Interaction System 2 (Axis2 por sus siglas en inglés) es una aplicación basada en Java, tanto del lado del cliente como del lado del servidor, para el desarrollo de servicios web. Proporciona un modelo de objetos completo y una arquitectura modular que facilita la tarea de añadir funcionalidad y dar soporte para nuevos servicios web.

Permite de modo sencillo realizar tareas como enviar, recibir y procesar mensajes SOAP (*Simple Object Access Protocol*), crear clases de implementación del lado del cliente y del lado del servidor empleando WSDL (*Web Service Description Language*). Es muy útil para la búsqueda de información.

1.8.1.2 JMS

Java Message Service (JMS) o servicio de mensajería de Java es un estándar de mensajería que permite a los componentes de aplicaciones basadas en Java 2 Enterprise Edition crear, enviar, recibir y leer mensajes. También hace posible la comunicación confiable de manera síncrona y asíncrona.

JMS mejora la productividad del programador mediante la definición de un conjunto de conceptos de mensajería y estrategias de programación que será apoyado por toda la tecnología de JMS compatible con los sistemas de mensajería. Además proporciona un servicio flexible y fiable para el intercambio de datos y eventos en todo el sistema.

1.9 Gestor de base de datos

Un sistema gestor de base de datos se define como el conjunto de programas que administran y gestionan la información contenida en una base de datos. Ayuda a realizar las siguientes acciones: definición de los datos, mantenimiento de la integridad de los datos dentro de la base de datos, control de la seguridad y privacidad y manipulación de los datos. (11)

El gestor de base de datos proporciona varios programas no visibles al usuario que se encargan de la privacidad, integridad y seguridad de los datos, así como la interacción con el sistema operativo. Brinda una interfaz entre los datos y los usuarios finales. En la actualidad en el mundo existen varios sistemas para la gestión de base de datos. Estos presentan características y funcionalidades diferentes, por lo que se realiza un estudio y análisis de los más utilizados y difundidos en el desarrollo de aplicaciones.

1.9.1 PostgreSQL

PostgreSQL es un potente motor de bases de datos, que tiene prestaciones y funcionalidades equivalentes a muchos gestores de bases de datos comerciales. (12)

Es un sistema de gestión de bases de datos relacional orientada a objetos y libre, publicado bajo la licencia BSD (*Berkeley Software Distribution*). Tiene como característica fundamental la alta concurrencia, que permite que mientras un proceso escribe en una tabla, otros accedan a la misma tabla sin necesidad

de bloqueos, lo cual será de gran utilidad en el Centro de Datos, ya que la misma HCE podrá ser consultada y modificada por varios profesionales al mismo tiempo.

1.10 Estándares médicos

En los últimos años ha crecido considerablemente la gestión de la información en las instituciones hospitalarias, centrada en los procesos de atención al paciente. La correcta gestión de información requiere de la interacción de múltiples sistemas de información cuándo sea necesario.

En la actualidad las instituciones de la salud cuentan con unidades que trabajan y se organizan de forma diferente, la mayoría cuenta con sistemas de información independientes. Los pacientes suelen atravesar los diferentes niveles de atención primario, secundario y terciario, pasando por uno o más médicos y especialistas, sin la posibilidad de poder comunicar, coordinar y compartir la información que se genera en los distintos puntos de atención. Como resultado se crean verdaderas islas de información entre las distintas instituciones, servicios y profesionales.

La comunicación de información es un componente fundamental de cualquier sistema. En el área de la salud se transfiere información entre los profesionales de la salud, entre instituciones, entre los profesionales y sistemas informáticos de soporte de decisión.

Una comunicación efectiva requiere que el emisor y el receptor de información compartan un “marco de referencia” común que permita la interacción. Los estándares proveen ese marco común, promoviendo una uniformidad en la denominación de los componentes del sistema de salud, ya sean objetos, diagnósticos, personas e intervenciones. (13) Estos estándares son un conjunto de reglas y definiciones que especifican el modo en que la información del paciente es almacenada e intercambiada electrónicamente. (14)

Hoy día los sistemas de información aislados no son recomendables, se necesitan sistemas integrados, que permitan la comunicación entre diferentes instituciones de salud y consultar información actualizada de los pacientes. Para que diferentes sistemas puedan integrar la información del paciente es necesaria la transferencia de un sistema a otro. Esta transferencia generalmente se ejecuta a través de múltiples interfaces adaptadas y personalizadas.

1.10.1 HL7

Una solución al problema de múltiples interfaces es el estándar de interoperabilidad *Health Level Seven* (HL7). HL7 nació en el año 1987, es una de las organizaciones más importantes en estándares de mensajería en informática médica. Es una implementación de la capa 7 del modelo de redes OSI, (*Open System Interconnection*) destinado a utilizarse en el ámbito de la salud. (15)

Desde 1994 la organización está acreditada como SDO (*Standards Developing Organization*) por el ANSI (*American National Standards Institute*). (16)

Su misión es proveer un marco completo de estándares relacionados para el intercambio, la integración y la recuperación de información electrónica de salud que soporte la práctica y la gestión clínica. HL7 permite el intercambio y la integración de los datos que provienen del proceso de la atención médica, a través del desarrollo de guías, metodologías y servicios en general, esto lo logra debido a que:

- ✓ Permite el intercambio de información entre aplicaciones desarrolladas por diferentes proveedores de software.
- ✓ Reduce el trabajo en papel, mejorando el soporte a las decisiones y permitiendo la integración de la información de salud entre diferentes servicios.
- ✓ Permite la conectividad entre sistemas heterogéneos a costos competitivos.
- ✓ Ofrece flexibilidad, porque puede implementarse usando diversas tecnologías de software.

1.10.2 HL7- CDA

La arquitectura clínica de documentos (CDA siglas en inglés), es una iniciativa muy importante dentro de HL7, que define un modelo para el intercambio de documentos clínicos, como por ejemplo reportes de consultas, resúmenes de internación e informes de estudios. Las especificaciones del CDA son flexibles y permiten la utilización de plantillas de acuerdo a diferentes escenarios. (17)

Es un estándar basado en XML para el marcaje de documentos que especifica la estructura y semántica de documentos clínicos con el propósito de facilitar su intercambio en un entorno de interoperabilidad.

Entre sus principales características están: (18)

- ✓ **Persistencia:** Un documento clínico continúa existiendo sin alteraciones por un período de tiempo definido.

- ✓ **Responsabilidad:** Un documento clínico debe ser mantenido por una organización a la cual se asigna su cuidado.
- ✓ **Potencial para autenticación:** Un documento clínico es un paquete de información que tiene prevista su autenticación legal.
- ✓ **Contexto:** Un documento clínico establece un contexto para su contenido: paciente, prestador, participantes, etc.
- ✓ **Compleitud:** La autenticación o firma de un documento clínico se aplica a todo su contenido y no a porciones del contenido sin el contexto del documento.
- ✓ **Legibilidad:** El documento debe ser leído por los seres humanos sin inconvenientes.

1.10.3 HL7- IHE

IHE (*Integrating the Healthcare Enterprise*), es una iniciativa de profesionales de la salud y empresas cuyo objetivo es mejorar la comunicación entre los sistemas de información que se utilizan en la atención al paciente. Define unos 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. (19)

IHE es una recomendación de uso de estándares, es una organización cuya misión es el desarrollo y la promoción de su recomendación en particular y de los estándares médicos en general, es un conjunto de especificaciones que forman un marco técnico.

1.11 Herramientas

1.11.1 Visual Paradigm para UML

Visual Paradigm para UML es una herramienta UML profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. El software de modelado UML ayuda a una más rápida construcción de aplicaciones de calidad, mejores y a un menor coste. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación. (20)

Esta herramienta es muy útil porque permite colaboración en un equipo de trabajo. Sus miembros pueden ver y editar el mismo proyecto o incluso el mismo diagrama de forma simultánea, puede importar/exportar de Microsoft Excel y varios formatos de imagen. Además su distribución es gratuita.

1.11.2 Eclipse

Eclipse es un software creado originalmente por IBM (*International Business Machines*) y actualmente, la Fundación Eclipse, una entidad sin ánimo de lucro, fundada por IBM se encarga de llevar a cabo el desarrollo de este proyecto. La definición que le da la Fundación Eclipse acerca de su software es: "una especie de herramienta universal - un IDE abierto y extensible para todo y nada en particular".

Eclipse es un entorno integrado de desarrollo (IDE, *Integrated Development Environment*) multiplataforma englobado en el movimiento de software libre para crear aplicaciones clientes de cualquier tipo.

Este IDE cuenta con un editor de código, un compilador/intérprete y un depurador. Eclipse sirve como IDE Java y cuenta con numerosas herramientas de desarrollo de software. También da soporte a otros lenguajes de programación además de brindar la posibilidad de añadir a la plataforma base de Eclipse extensiones (*plugins*) para extender sus funcionalidades.

1.11.3 PgAdmin

PgAdmin es la aplicación gráfica más popular y robusta de licencia Open Source para la administración de bases de datos en PostgreSQL. Se puede usar en Linux, FreeBSD, Solaris, Mac OS X y Windows. Permite gestionar versiones a partir de PostgreSQL 7.3, así como versiones comerciales de PostgreSQL. Está diseñado para responder a las necesidades de los usuarios, desde escribir consultas SQL simples hasta desarrollar bases de datos complejas.

Su interfaz gráfica soporta todas las características de PostgreSQL y permite que la administración sea mucho más sencilla. La aplicación incluye también un editor SQL con resaltado de sintaxis, un editor de código en el servidor, un agente para lanzar script programados y mucho más. La conexión al servidor puede hacerse mediante conexión TCP/IP o *Unix Domain Sockets*, y puede encriptarse mediante SSL para mayor seguridad.

1.11.4 Jboss Tools

JBoss Tools es un conjunto de plugins de Eclipse que tiene como objetivo ayudar a los desarrolladores a crear aplicaciones webs de forma rápida y sencilla.

Los módulos de JBoss Tools son:

- ✓ **RichFaces VE:** El editor visual aportado por Exadel proporciona el apoyo para la edición visual de páginas HTML, JSF, JSP y Facelets. También incluye soporte visual para las librerías de componentes JSF incluyendo JBoss RichFaces.
- ✓ **Seam Tools:** Incluye soporte para seam-gen, RichFaces VE.
- ✓ **Hibernate Tools:** Soporta el mapeo de archivos, anotaciones y JPA con la ingeniería inversa, completamiento de código, asistentes de proyecto, refactorización, ejecución interactiva de HQL/JPA-QL/Criteria.
- ✓ **JBoss AS Tools:** Fácil de iniciar, detener y debuguear al estar integrado con Eclipse. También incluye funciones para el despliegue eficaz de cualquier tipo de proyecto en el IDE.
- ✓ **Drools IDE:** Editor de ficheros de reglas, debugueo e inspección de reglas.
- ✓ **jBPM Tools:** Edición del flujo de trabajo del jBPM, motor de procesos BPM.
- ✓ **JBossWS Tools:** Desarrollo, invocación, inspección y pruebas de servicios web sobre http con la adición y soporte de características JBossWS.

1.11.5 Apache Solr

El Servidor Apache HTTP es un servidor web de tecnología *Open Source*, sólido y para uso comercial, desarrollado por la *Apache Software Foundation*. Apache ha publicado la versión 1.4 de Solr, un motor de búsquedas web hecho 100% en java. Está basado en *Apache Lucene* y puede ejecutarse en un contenedor web como Tomcat. Entre las características de este servidor se destacan sus servicios de administración, caché, replicación, índices, capacidad para procesar documentos de texto como Word y PDF.

En esta nueva versión se ha incrementado el rendimiento del indexado y la búsqueda, se han simplificado las capacidades de replicación, se ha mejorado la integración con bases de datos, se ha añadido la

capacidad para manipular documentos de texto rico, y se proporcionan múltiples *plugins* nuevos para extender su funcionalidad por defecto. (21)

Apache Lucene es un motor de búsqueda de texto sobre cualquier tipo de contenido. Básicamente Lucene se divide en dos grandes partes, la indexación y la búsqueda. En la indexación Lucene crea un índice sobre el que luego realicen las búsquedas, y el cual construye mediante el análisis de la información que le llega. En el proceso de búsqueda Lucene usa el índice que ha construido anteriormente para devolver la información que coincida. (22)

La gran potencia y complejidad de Lucene reside en el hecho de que sea un framework, es decir, ofrecen las herramientas para construir un servicio de indexación y búsqueda utilizando el conjunto de clases que ofrece.

Conclusiones

Los sistemas similares automatizados existentes que han sido estudiados en la presente investigación cuentan con muy poca documentación y no brindan información sobre las funcionalidades disponibles, por lo que se dificulta decidir si estas soluciones se adecuan o no a las necesidades reales. A esto se le suma que estos sistemas pertenecen a compañías multimillonarias estadounidenses y en algunos casos su adquisición resulta muy costosa.

A partir del estudio de los sistemas analizados surge la necesidad de crear un Centro de Datos para la Historia Clínica Electrónica centralizada que se adapte a las necesidades actuales y sea capaz de comunicarse con sistemas de saludes nacionales e internacionales.

Para realizar el diseño del Centro de Datos se propone como metodología de desarrollo de software Rational Unified Process (RUP) con notación BPMN como estándar internacional para la fase de Modelo del Negocio, en aras de lograr una mejor administración de los procesos de negocio.

En el presente capítulo se realiza un estudio del conjunto de tecnologías y tendencias actuales que formarán parte de la propuesta de infraestructura tecnológica del Centro de Datos.

Capítulo 2. Características de la solución

Introducción

En el presente capítulo se describen cada uno de los perfiles de integración que formarán parte de la propuesta de solución así como los principales procesos encontrados durante el análisis del problema. Además se realiza una breve descripción de los conceptos fundamentales encontrados en el Modelo de Dominio propuesto que ayudarán a una mejor comprensión del mismo.

2.1 Sistemas que van a interoperar

El desarrollo vertiginoso de la rama de la informática ha traído numerosas ventajas para todos los sectores de la sociedad, pero sin dudas, el sector de la medicina, ha sido uno de los más beneficiados. 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. No basta con tener datos e información, hay que procesarla, analizarla, interpretarla y utilizarla.

La Informática Médica incluye soluciones para el sector de la salud en cinco líneas de desarrollo: Sistemas para la Atención Primaria de Salud, Sistemas de Gestión Hospitalaria, Sistemas Especializados de Salud, Sistemas de Apoyo a la Salud y Sistemas para la Gestión y Procesamiento de Imágenes Médicas.

2.1.1 APS

Uno de los sistemas que se encuentra en desarrollo actualmente es el software para la Atención Primaria de Salud (APS). En Cuba, por ejemplo, la atención primaria da solución aproximadamente al 80% de los problemas de salud de la población. Es el principal punto de contacto entre la población y los sistemas de servicio sociales y de salud.

Sus actividades están relacionadas fundamentalmente con los Centros de Salud Familiar o de la Comunidad, Unidades Móviles, Unidades Rurales, así como en Centros Especializados en Atención Primaria de Salud (CEAPS) que cumplen objetivos concretos en la sociedad. Se centran en actividades de promoción, investigación, prevención, prescripción y protección de la salud en la comunidad, así como acciones curativas básicas. Es una parte muy importante debido a la continuidad de la atención al paciente que ofrece y el flujo de información a lo largo de todo el sistema de atención de salud.

2.1.2 HIS

El Sistema de Información Hospitalaria (*Hospital Information System*, HIS) tiene como propósito fundamental la optimización de los recursos humanos y materiales para satisfacer las necesidades de las diferentes áreas de las organizaciones de salud. Está encaminado a incorporar las tecnologías informáticas al profesional de la salud para lograr que estas auxilien y optimicen sus funciones y por ende, brindar un servicio de óptima calidad al paciente.

El HIS está compuesto por varios subsistemas o módulos, entre los que se encuentran Admisión, Emergencia, Consulta externa, Laboratorio, Hospitalización, Farmacia, entre otros. Todos estos módulos integrados, permiten disponer de información exacta, obtener datos estadísticos más reales, hacer mejores planificaciones y evaluaciones económicas y administrativas.

2.1.3 RIS

El Sistema de Gestión de Información Radiológica (*Radiologic Information System*, RIS) informatiza la actividad radiológica de un paciente, desde la petición de una cita para la realización de un estudio hasta la elaboración del informe que se entrega luego de cada consulta.

Permite el registro de la información de especialistas, y los equipos médicos, así como el control de una historia clínica imagenológica, obtención de hojas de cargo, estadísticas médicas y listas de trabajo tanto para especialistas como equipos que produzcan imágenes en formato DICOM 3.0 (*Digital Imaging and Communications in Medicine*).

2.1.4 PACS

El Sistema para la Transmisión, Visualización y Almacenamiento de Imágenes Médicas (*Picture Archiving and Communication System*, PACS) tiene como objetivo gestionar, controlar y almacenar los estudios imagenológicos en las instituciones hospitalarias. Este sistema transmite digitalmente las imágenes electrónicas e informes médicos, lo que elimina los enormes archivos y la tediosa búsqueda de información. PACS ofrece un acceso oportuno y eficiente a las imágenes, su interpretación y presentación.

La interacción con el RIS es fundamental para el mejor aprovechamiento de las capacidades del PACS. El RIS proporcionará al PACS toda la información sobre las citas existentes. 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 éste, el RIS envía una copia al PACS y la notificación de que el informe ha sido realizado. (23)

Todos estos sistemas con que hoy cuenta la salud tienen como propósito fundamental facilitar y apoyar el trabajo de especialistas y brindarle al paciente un servicio de excelencia. Pero lamentablemente, todos ellos son enormes islas de información que no se comunican entre sí y por tanto no ofrecen una vista única del paciente. Es por esto que se hace necesaria la interoperabilidad entre ellos, que permita transferir y/o intercambiar información clínica.

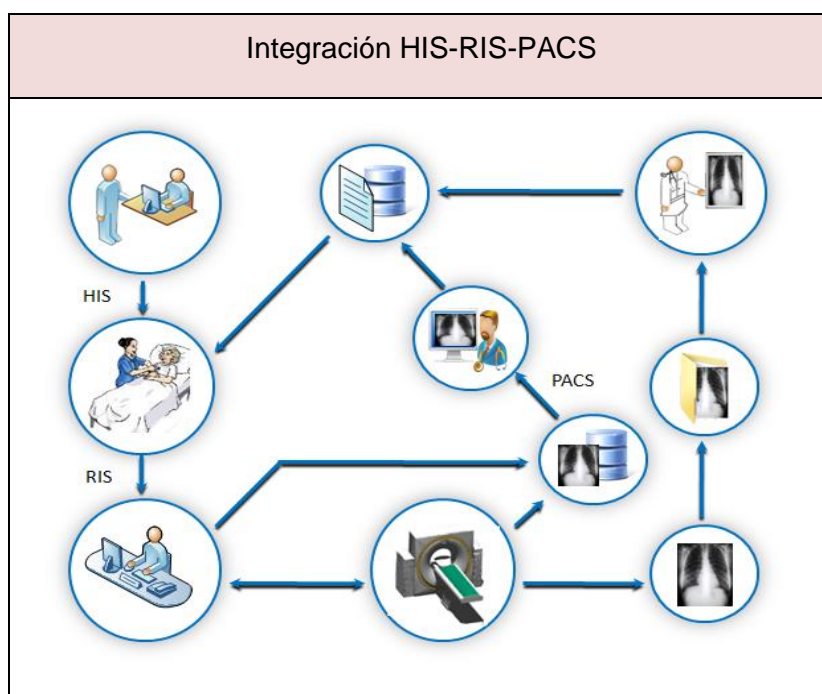


Figura 2.1 Integración HIS-RIS-PACS

Para lograr la interoperabilidad entre los sistemas se hace necesaria la utilización del estándar HL7. Éste es el puente de unión de esos sistemas. Provee un marco completo de estándares relacionados para el intercambio, la integración y la recuperación de información electrónica de salud que soporte la práctica y la gestión clínica.

Los sistemas de salud manejan terminología propia de los médicos que es difícil recordar en la mayoría de los casos. HL7 ofrece además un conjunto de Servicios de Terminología Común o *Common Terminology Services* (CTS) que dan solución a este problema.

Otro problema que surge con la integración de sistemas es la forma en que son mostrados los documentos clínicos. La arquitectura de documentos clínicos o CDA, es un estándar muy importante dentro de HL7. Este especifica la estructura y semántica de los documentos clínicos, definiendo un modelo para el intercambio de los mismos.

Para mejorar los procesos de comunicación entre los sistemas de información que se utilizan para llevar a cabo la atención del paciente, surge *Integrating the Healthcare Enterprise* (IHE).

2.2 Perfiles de Integración

IHE define unos 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. 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 también los componentes funcionales, a los que llamaremos 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). (24)

Un perfil de integración incluye un caso de uso y la serie completa de procedimientos compuestos por una serie de pasos individuales. Los perfiles de integración incluyen especificaciones técnicas detalladas sobre el uso e implementación de los estándares en los que se basa, asegurando de esta forma un flujo de información ininterrumpido entre diferentes aplicaciones para cumplir con el caso de uso específico.

Los perfiles de integración se agrupan en documentos o dominios, que los aglutinan en áreas clínicas. Estos dominios se conocen también como Marcos Técnicos (*Technical Framework*). Algunos de los definidos hasta el momento son: Radiología (RAD), Radioterapia (RO), Cardiología, Infraestructura de las Tecnologías de Información (ITI por sus siglas en inglés), Dispositivos de Monitorización y Coordinación del Cuidado de Paciente (PCC siglas en inglés).

El Marco Técnico ITI define como implementar las políticas principales relativas a seguridad e intercambio de datos en cualquier organización que maneje información clínica en formato electrónico, independientemente de cualquier especialidad médica. Dentro de este se encuentran los perfiles de integración *Patient Identifier Cross-referencing* (PIX) y *Cross Enterprise Document Sharing* (XDS).

El Marco Técnico PCC define las implementaciones de las normas establecidas para promover el adecuado intercambio de información clínica y coordina la óptima atención al paciente entre los proveedores de salud procedentes de centro de atención diferentes. Dentro de este dominio se enmarca el Perfil de Integración *Query for Existing Data* (QED).

2.2.1 Patient Identifier Cross-referencing (PIX)

Este perfil se enfoca en la referencia cruzada de identificadores de pacientes procedentes de distintos Dominios de Identificación de Pacientes. Esos identificadores cruzados pueden ser utilizados por otros sistemas “Consumidores de Identidades” para correlacionar información sobre un único paciente procedente de fuentes o sistemas que almacenan datos del mismo paciente, pero con identificadores diferentes.

La utilización de este perfil en los distintos sistemas que quieran comunicarse les permitirá a los profesionales médicos tener una vista mucho más completa de la información del paciente. Mediante este perfil los usuarios del sistema de salud con los permisos correspondientes podrán acceder a la información del paciente si tener en cuenta si este se ha atendido o no en la entidad hospitalaria. Acción que podrán realizar siempre que exista conexión con el Centro de Datos.

Este perfil se encarga de la búsqueda de pacientes tanto en la BD Local de la entidad como en el Centro de Datos, de registrar y actualizar la información de pacientes. Además permite realizar la mezcla de pacientes que por errores hayan sido creados en más de una ocasión y posean más de una HCE, permitiendo la vista única de los datos clínicos del paciente y una HCE centralizada.

2.2.2 Cross Enterprise Document Sharing (XDS)

En la actualidad se le considera un perfil revolucionario que está permitiendo la existencia de una Historia Clínica Electrónica basada en documentos distribuidos. Permite registrar y compartir dichos documentos clínicos entre diferentes empresas o sistemas de salud, que pueden ser desde pequeñas consultas hasta

centros de atención primaria y hospitales. Tiene como características fundamentales que es escalable, de fácil acceso e independiente de la estructura hospitalaria y del tipo de documento.

Para la utilización de este perfil es necesario aplicar un conjunto de políticas en las organizaciones pertenecientes a un mismo Dominio de Afinidad¹ que van a interoperar. Estas políticas permitirán el intercambio de información cuando el paciente requiera realizar un cambio en su situación o en su proveedor de salud.

IHE nos indica que estas políticas deben incluir al menos: (25)

- Definición de los identificadores comunes de pacientes.
- Definición de los mecanismos de obtención de consentimiento.
- Definición del control de acceso a datos.
- Definir el formato, contenido, estructura, organización y representación de la información clínica.

El perfil XDS permite intercambiar información clínica de diferentes tipos entre distintas organizaciones. Es fundamental para lograr la HCE única del paciente ya que posibilita que los cambios realizados en la HCE de un paciente almacenada en la BD Local de una entidad sean actualizados también en el Centro de Datos y viceversa.

2.2.3 Query for Existing Data (QED)

Una amplia variedad de sistemas a menudo necesitan del acceso a la información clínica dinámica almacenada y mantenida en sistemas que gestionen la Historia Clínica Electrónica. Este perfil permite que la información esté ampliamente disponible para otros sistemas dentro y entre las empresas, para apoyar la prestación de un mejor cuidado clínico. (26)

El perfil de integración (QED) apoya la dinámica de los servicios de consulta de los datos clínicos del paciente. La utilización de este perfil en los diferentes sistemas que quieran comunicarse permitirá la consulta de documentos clínicos de la HCE de pacientes que se encuentren tanto en la BD Local de una entidad como en el Centro de Datos. Los documentos serán registrados ordenadamente, empleando los índices. La indexación proporciona agilidad en las búsquedas, lo que se traduce en mayor rapidez a la

¹ Dominio de Afinidad: Es un conjunto de empresas de salud que han acordado trabajar juntos utilizando un conjunto común de políticas y compartir una estructura común.

hora de mostrar resultados.

2.3 Common Terminology Services (CTS)

Common Terminology Services (CTS) o los Servicios de Terminología Común de HL7 definen algunas interfaces de programación de aplicaciones (API por sus siglas en inglés) que pueden ser utilizadas por un software para acceder al contenido terminológico.

Un API es conjunto de estructuras de datos que deben integrarse dentro de la versión 3 del marco de mensajería de HL7 para permitir una gran variedad de contenido terminológico, sin la necesidad de una migración importante o reescribir. (27)

HL7 versión 3 consta de dos capas distintas entre las aplicaciones de procesamiento de mensajes y el vocabulario de destino: la API de Mensajes y la API de Vocabulario. La capa superior, la API de mensajes, se comunica con el software de mensajería, y lo hace en términos de dominios de vocabulario, contextos, conjuntos de valores, código de atributos y otros artefactos del modelo de mensaje de HL7. La capa inferior, la API de vocabulario, se comunica con el software de servicio de terminología, y lo hace en términos de sistemas de código, los códigos de concepto, denominaciones, relaciones y entidades de la terminología específica.

La API de mensajes es específica para HL7. Su propósito principal es permitir una amplia variedad de aplicaciones de procesamiento de mensajes para crear, validar y traducir derivados de tipos de datos de una manera coherente y reproducible.

La API de vocabulario está destinada a ser genérica. Permite a las aplicaciones consultar diferentes terminologías de forma bien definida. La API de mensajes utiliza la API de vocabulario.

Además de estas dos APIs, el servicio CTS utiliza los siguientes componentes para lograr su objetivo: *Mensaje Runtime* o Mensaje en tiempo de ejecución, *Vocabulary Runtime* o Vocabulario en tiempo de ejecución, *Mensaje Browsing* o Buscador de mensaje, *Vocabulary Browsing* o Buscador de vocabulario y *Code Mapping* o Asignar código.

- **Mensaje en tiempo de ejecución:** Describe los servicios utilizados por la creación del mensaje, el procesamiento y el software de traducción. Traduce el texto suministrado al código que utiliza el sistema

destino y devuelve entre otras cosas, una lista de los dominios de vocabulario que tienen concordancia con el texto introducido.

- **Vocabulario en tiempo de ejecución:** Es un conjunto de funciones que son utilizadas por el API de mensajes y la capa de servicios de navegador, así como el servicio Capa de Vocabulario navegador. Determina si el código introducido es o no válido. Retorna el nombre preferido o común del concepto introducido en el idioma suministrado.
- **Asignar código:** Devuelve el nombre que le fue asignado al servicio por el proveedor de servicios, así como su descripción, los autores, etc. Además una lista de asignaciones que son compatibles con el servicio.
- **Buscador de mensajes:** Retorna el nombre del servicio y su descripción. Devuelve, entre otras cosas, una lista de atributos cuyo nombre coincide con el texto suministrado en el explorador. Busca toda la información existente sobre el vocabulario suministrado y determina si el código del concepto introducido tiene un valor válido o no.
- **Buscador de vocabulario:** Devuelve el nombre del servicio y su descripción. Retorna, entre otras cosas, una lista de los códigos de concepto cuyo nombre coincida con el introducido en el lenguaje introducido.

Para obtener información de un término determinado, la aplicación o el sistema solicita al servicio de mensaje en tiempo de ejecución llenar los detalles de un atributo. El servicio, a su vez, realiza varias llamadas al servicio de la API de Vocabulario para obtener la información solicitada. Para mostrar una respuesta al usuario el API de Vocabulario se comunica con el API de Mensaje.

2.4 Modelo de Dominio

El Modelo de Dominio es una representación visual estática del entorno real de la solución que se está modelando. Proporciona una vista de las clases que intervienen en el negocio y sus relaciones.

Para comprender mejor a continuación se ofrece una explicación de los conceptos fundamentales identificados durante el análisis del problema.

- **HCE:** Contenedor de información sobre el paciente. Está compuesta por documentos clínicos.
- **Repositorio de documentos clínicos:** Contiene los documentos clínicos que forman parte de la HCE de un paciente.
- **Base de Datos:** Contiene todo tipo de información referente al paciente, las entidades hospitalarias, los sistemas de salud así como los usuarios que interactúan con ellos.
- **ActiveMQ Network of Brokers:** Se utiliza para el envío de documentos clínicos.
- **Axis Clúster:** Se utiliza para publicar y consumir servicios web que implementen funcionalidades que no requieran el envío de documentos clínicos.
- **BRIDGE:** Puente que comunica los diferentes RHIOS con la estructura completa del Centro de Datos.
- **BD Local:** Almacena los datos de los pacientes de una entidad.
- **Repositorio local de documentos:** Almacena los documentos clínicos de la HCE de los pacientes pertenecientes a una entidad determinada.
- **CTS WS:** Servicio web contenido en el BRIDGE que permite gestionar los términos médicos.
- **PIX WS:** Servicio web contenido en el BRIDGE que se encarga del registro y la identificación de pacientes.
- **XDS WS:** Servicio web contenido en el BRIDGE que permite compartir documentos clínicos entre sistemas de información y el Centro de Datos.
- **QED WS:** Servicio web contenido en el BRIDGE que permite consultar documentos clínicos guardados tanto en el repositorio de las entidades como en el Centro de Datos.
- **RHIO:** Es la asociación entre una entidad hospitalaria y los sistemas de salud que pudiese contener.
- **Sistemas de salud:** Sistema que incorpora las TIC al servicio de la salud, con el objetivo de facilitar el trabajo de los profesionales y mejorar la calidad de los servicios prestados al paciente.
- **APS:** Atención Primaria de Salud. Se centra en actividades de prevención y protección de la salud en la comunidad, así como actividades curativas básicas.
- **HIS:** Sistema de Información Hospitalaria. Gestiona los recursos humanos y materiales para satisfacer las necesidades de las diferentes áreas de las organizaciones de salud.
- **RIS:** Sistema de Gestión de Información Radiológica. Informatiza la actividad radiológica del paciente.

- **PACS:** Sistema para la Transmisión, Visualización y Almacenamiento de Imágenes Médicas. Gestiona, controla y almacena los estudios imagenológicos en las instituciones hospitalarias.
- **Entidad:** Establecimiento destinado al cuidado y prevención de la salud.
- **Policlínico:** Espacio territorial con límites demográficos definidos en el que reside una población que recibe atención integral de salud. Puede ser docente, asistencial y administrativo. Constituye un eslabón fundamental dentro de la Atención Primaria de Salud.
- **Hospital:** Es el establecimiento destinado a la prevención, diagnóstico y tratamiento de los individuos que tienen alguna enfermedad o padecimiento. Lugar donde se practican además la investigación y la enseñanza. Contiene Áreas, Departamentos y Servicios.
- **Clínica Imagenológica:** Clínicas especializadas en la realización de estudios imagenológicos.
- **Usuario:** Persona que interactúa con alguno de los Sistemas de salud.
- **Médico:** Recoge información sobre el estado del paciente mediante las distintas herramientas, realiza un análisis de la misma, obtiene un diagnóstico y desarrolla un programa de tratamiento y formación del paciente respecto a su propia salud. Es el agente activo en el proceso sanitario.
- **Enfermero:** Abarca los cuidados que se prestan a las personas de todas las edades, familias, grupos y comunidades, enfermos o sanos. Incluye la promoción de la salud, la prevención de la enfermedad, y los cuidados de los enfermos y discapacitados.
- **Secretario:** También se conoce como asistente o recepcionista. Es la responsable de la gestión de turnos y la entrega de resultados.
- **Especialista:** Es el que emite los informes diagnósticos. Pueden ser radiólogos, cardiólogos, especialistas en Medicina Nuclear, gastroenterólogos, entre otros especialistas en el área de imagenología.
- **Técnico de estadísticas:** Persona que utiliza un conjunto de recursos matemáticos con el fin de organizar y resumir una gran cantidad de información obtenida de la realidad y sacar conclusiones respecto a ella.
- **Técnico en radiología:** Son los que trabajan directamente con el equipo de adquisición de imágenes, que puede ser Tomografía Axial Computarizada (TAC), Resonancia Magnética (RM), Ultrasonido (US), entre otros.
- **Director de policlínico:** Es la persona que dirige y lleva el control administrativo del policlínico. Es el responsable de su buen funcionamiento.

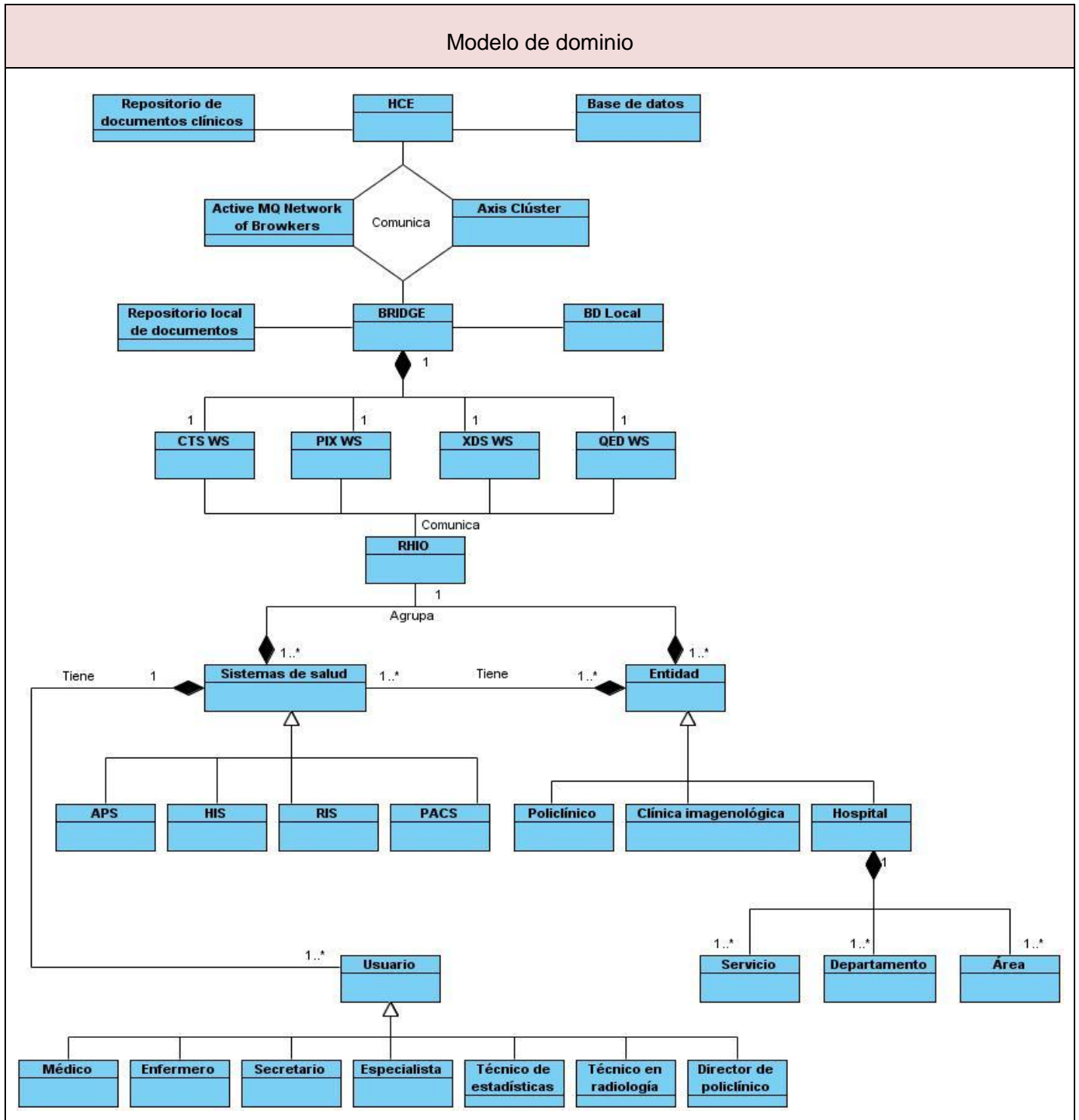


Figura 2.2 Modelo de dominio del Centro de Datos

2.5 Flujo de procesos

Para modelar los procesos que serán implementados se usan los Diagramas de Procesos de Negocio que propone BPMN. Los Diagramas de Procesos del Sistema son una representación gráfica y abstracta de los procesos de una organización. Muestran los actores involucrados, sus actividades, entradas y salidas de esas actividades, recursos involucrados y eventos que dirigen el proceso.

Diagrama de Proceso del Centro de Datos

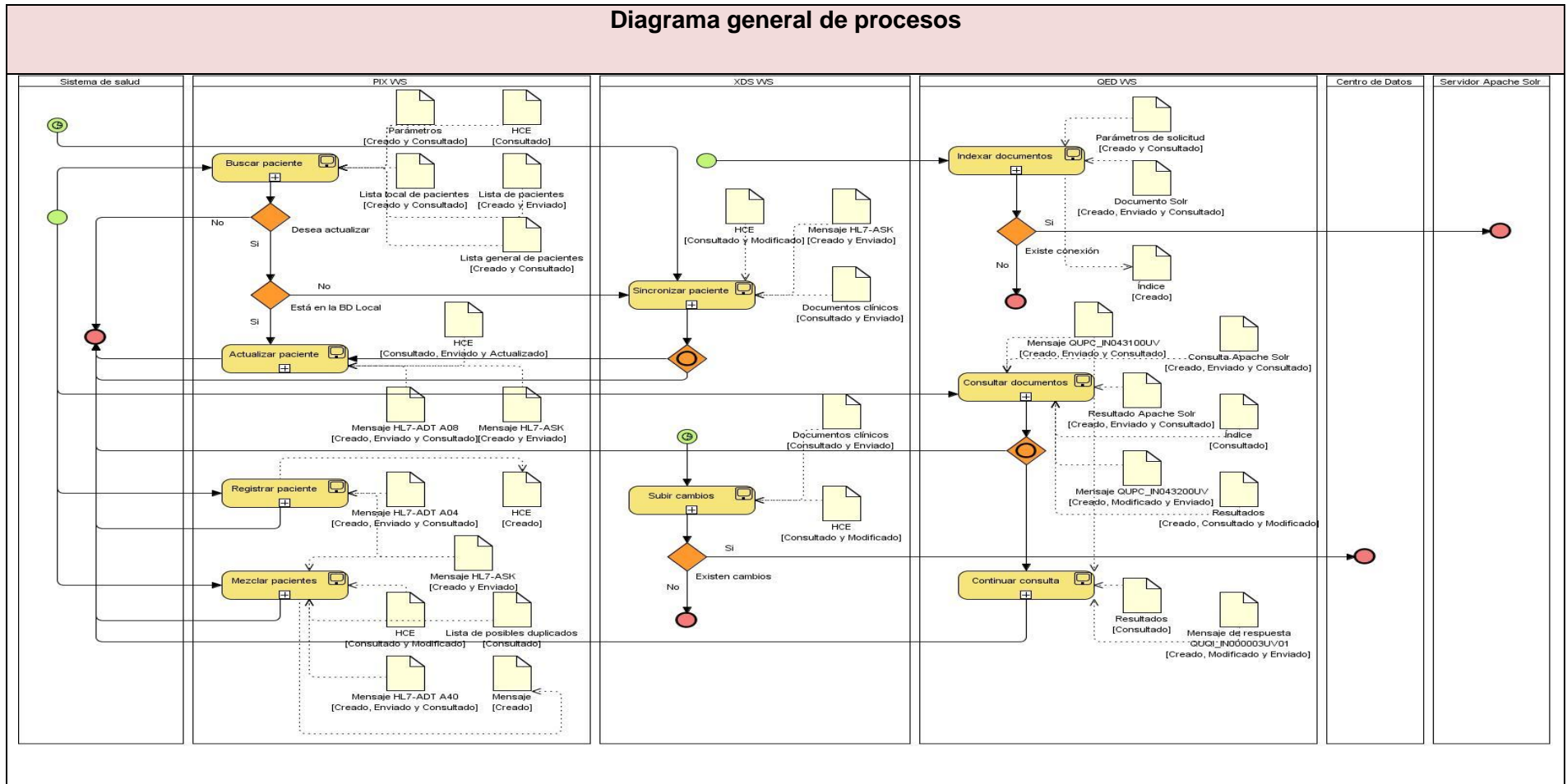


Figura 2.3 Diagrama general de procesos del Centro de Datos

2.5.1 Flujo de procesos del PIX

2.5.1.1 Buscar paciente:

Permite realizar la búsqueda de pacientes tanto en la BD Local como en el Centro de Datos. El usuario de la entidad que solicita la búsqueda envía a través del servicio web PIX WS el identificador del paciente y una variable booleana que indica el tipo de búsqueda que desea hacer. Si la variable tiene valor *true* indica que la búsqueda se realizará tanto en la BD Local como en el Centro de Datos, si es *false* solamente se busca en la BD Local. Devuelve una Lista de pacientes que es mostrada al usuario. (Ver Anexo 1)

2.5.1.2 Registrar paciente:

Permite registrar los datos de un paciente y almacenarlos en la BD Local de la entidad. El usuario de la entidad introduce los parámetros requeridos para registrar o crear un paciente. Dichos datos se codifican en la propia entidad y se envían en forma de un mensaje HL7 a través del PIX WS, este los decodifica y crea una nueva HCE del paciente, almacenando los datos recibidos en la BD Local. (Ver Anexo 2)

2.5.1.3 Actualizar paciente:

Permite acceder y actualizar los datos de un paciente, independientemente de si éste se atendió o no alguna vez en la entidad. Para esto necesita del proceso Buscar paciente descrito anteriormente, contenido en este perfil y del proceso Sincronizar paciente, del perfil XDS.

El usuario de la entidad realiza una búsqueda por identificador del paciente que se desea actualizar. Selecciona el paciente de la lista mostrada e introduce los parámetros que desea actualizar. Estos datos se codifican en la propia entidad y se envían en forma de un mensaje HL7 a través del PIX WS, este los decodifica y los procesa. Si el paciente se ha atendido con anterioridad en la entidad y sus datos se encuentran en la BD Local, actualiza la información del paciente. Si el paciente no es encontrado, se solicita la sincronización del paciente (proceso contenido en el perfil XDS) y posteriormente se le actualizan sus datos. (Ver Anexo 3)

2.5.1.4 Mezclar pacientes:

Permite mezclar 2 pacientes que tienen HCE con alto grado de similitud y se determina que son la misma persona.

El usuario consulta la Lista de posibles duplicados de la BD Local y selecciona los pacientes que desea fusionar. El servicio web PIX WS los mezcla, guarda los cambios en la BD Local y envía una notificación al Centro de Datos para que los pacientes sean mezclados también ahí y garantizar la vista única de los datos del paciente. (Ver Anexo 4)

2.5.2 Flujo de procesos del XDS

2.5.2.1 Sincronizar paciente:

Permite mantener actualizada la BD Local de una entidad, solicitándole al Centro de Datos los documentos clínicos de la HCE de un paciente determinado.

El usuario realiza una búsqueda de la HCE que desea sincronizar. Si es encontrada en la BD Local se busca la revisión actual de sus documentos y se envía la solicitud de sincronización. El Centro de Datos compara los documentos y envía los nuevos documentos al XDS, éste los persiste y guarda los cambios en la BD Local.

Si el paciente no se encuentra en la BD Local (porque no se haya atendido nunca en la entidad) el XDS solicita la HCE completa del paciente al Centro de Datos. Éste le envía todos los documentos clínicos y el XDS los guarda en la BD Local. Este proceso se ejecuta también automáticamente en el sistema, sin la intervención de ningún usuario. (Ver Anexo 5)

2.5.2.2 Subir cambios:

Permite mantener actualizado el Centro de Datos, enviando los cambios realizados en las HCE almacenadas en la BD Local.

Este proceso se ejecuta automática y periódicamente en el sistema. Busca los cambios que se han realizado en las HCE de la BD Local, carga los documentos nuevos encontrados y los envía hacia el Centro de Datos. Éste los recibe y los almacena. (Ver Anexo 6)

2.5.3 Flujo de procesos del QED

2.5.3.1 Indexar documentos:

Permite adicionar los documentos clínicos del paciente al índice de documentos. El XDS WS envía al QED WS la solicitud para indexar un documento. Si hay conexión con el servidor Apache Solr, se crea un documento de este tipo y se envía al servidor, éste lo indexa y lo almacena. Si no hay conexión se termina el proceso.

2.5.3.2 Consultar documentos:

Permite consultar documentos de HCE de pacientes almacenadas en el índice de documentos. El sistema envía los parámetros de una consulta y la cantidad de resultados que desea le sean mostrados. Si hay conexión con el servidor Apache Solr, se crea la consulta de este tipo que será enviada al propio servidor. Éste consulta los índices que tiene almacenados y crea un resultado que es enviado al QED.

El QED procesa los resultados y los adiciona a un mensaje de respuesta que es enviado al usuario en dependencia de la cantidad solicitada. El mensaje informa además si queda algún resultado que no ha sido mostrado al usuario y si ocurrieron o no errores durante el proceso. El usuario recibe la información y si quedan resultados decide si quiere continuar con la búsqueda.

El proceso finaliza en algunos de los siguientes casos: si ocurrieron errores en el mensaje que contenía la consulta realizada por el sistema, si no existe conexión con el servidor Apache Solr (encargado del índice de documentos) y al obtener resultados que serán mostrados al usuario.

2.5.3.3 Continuar consulta:

Permite obtener más resultados de una consulta de documentos realizada con anterioridad. El sistema envía un mensaje al QED, este lo analiza y determina si es para continuar o cancelar una consulta. Si decide cancelar, se envía al sistema un mensaje de respuesta y termina el proceso.

Si el mensaje recibido es de continuar una consulta, se adicionan los resultados al mensaje de respuesta en dependencia de lo solicitado por el usuario, se verifican si todos los resultados encontrados de la consulta realizada anteriormente fueron devueltos y se adiciona toda esa información al mensaje de respuesta, que es enviado al usuario.

Diagrama de proceso Indexar documentos.

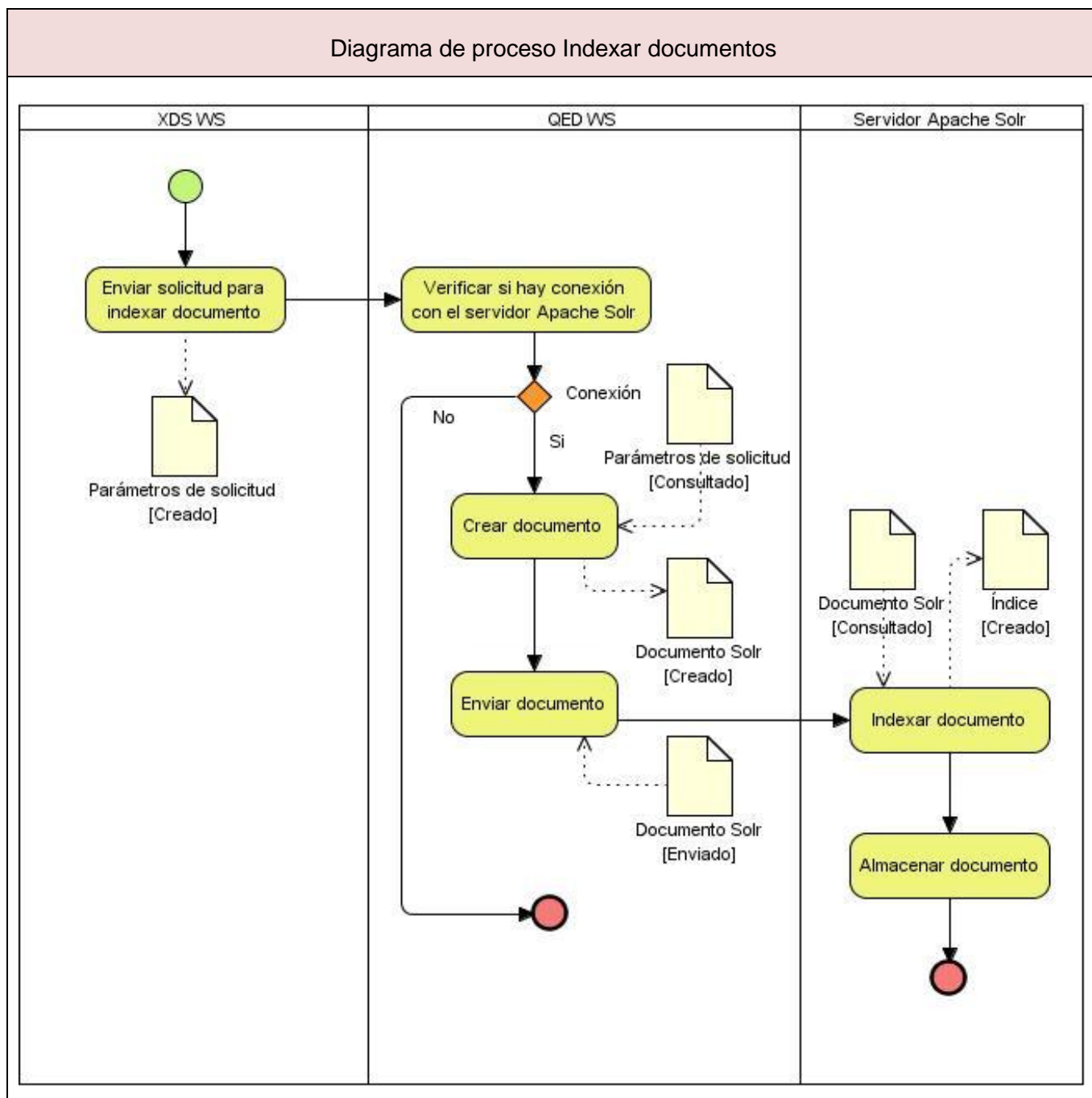


Figura 2.4 Diagrama de proceso Indexar documentos

Diagrama de proceso Consultar documentos.

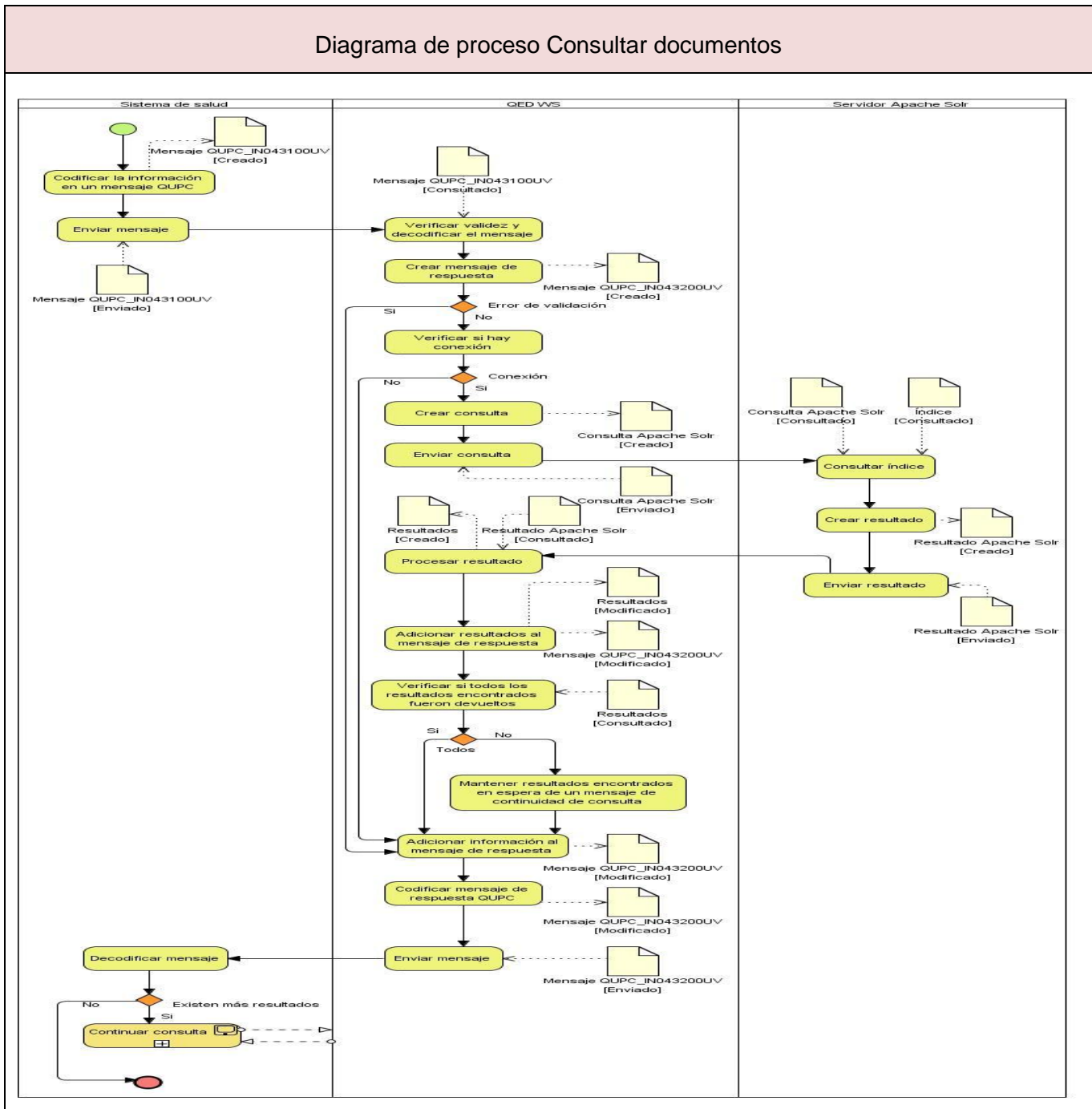


Figura 2.5 Diagrama de proceso Consultar documentos

Diagrama de proceso Continuar consulta.

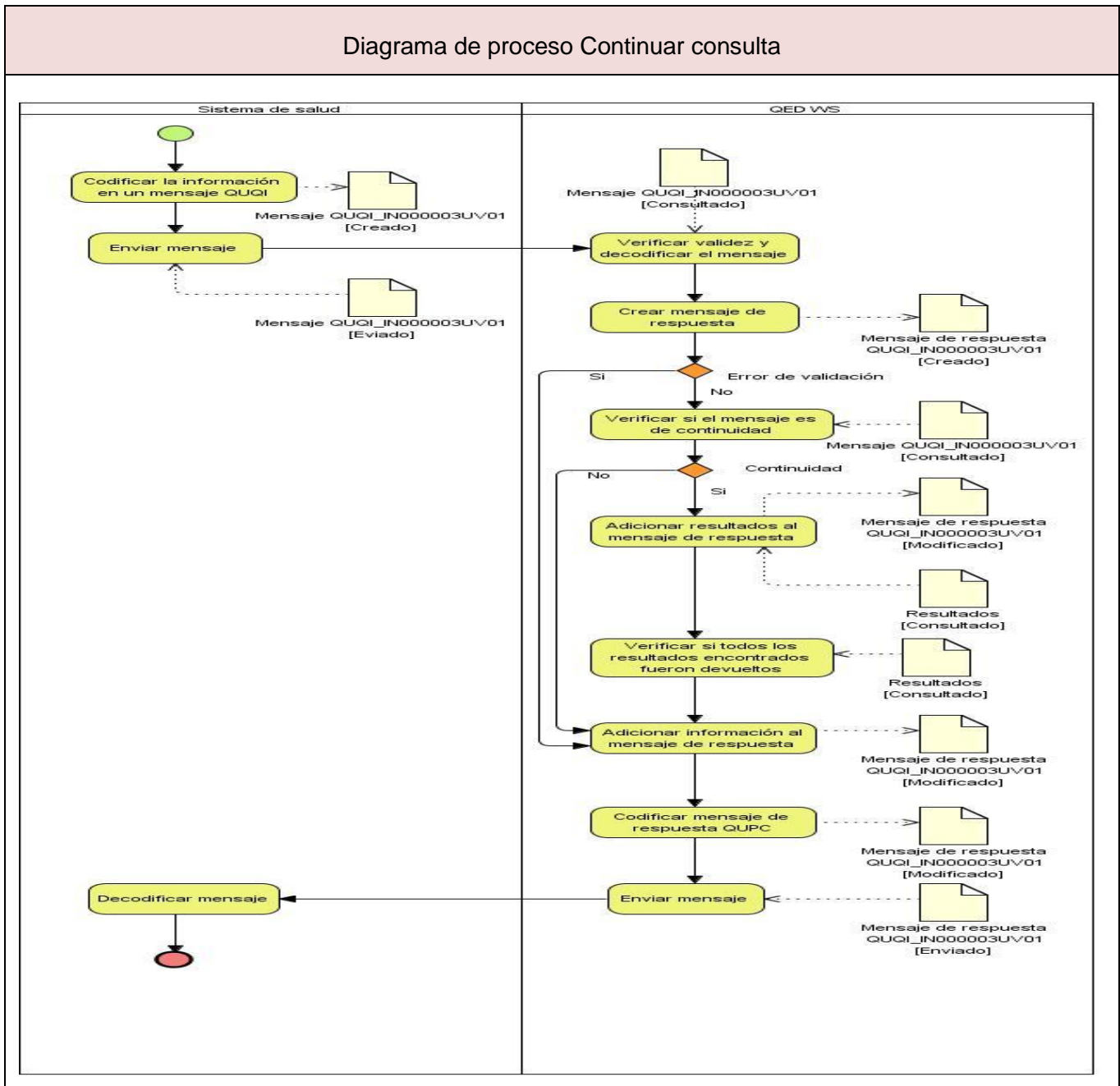


Figura 2.6 Diagrama de proceso Continuar consulta

2.6 Requisitos del sistema

La solución que se propone forma parte del Sistema de Información Hospitalaria alas HIS. La implementación del perfil *Query for Existing Data* (QED) permitirá, conjuntamente con los otros perfiles de integración antes descritos, la interoperabilidad entre los diferentes sistemas de salud que puedan estar en una institución hospitalaria.

2.6.1 Requisitos funcionales

Los requisitos funcionales surgen de la necesidad fundamental de la existencia del producto. Son las capacidades o condiciones que el sistema debería tener, las acciones que debería realizar para alcanzar el objetivo por el cual fue concebido.

A partir de los procesos del sistema estudiados se pueden identificar los siguientes requisitos funcionales.

El sistema debe permitir:

RF 1- Conectarse al servidor Apache Solr.

RF 2- Crear documento Apache Solr.

RF 3- Enviar documento al servidor Apache Solr.

RF 4- Indexar documento Apache Solr.

RF 5- Validar mensaje HL7.

RF 6- Decodificar mensaje HL7.

RF 7- Crear mensaje HL7.

RF 8- Crear consulta Apache Solr.

RF 9- Consultar índice de documentos.

RF 10- Enviar mensaje HL7.

RF 11- Continuar consulta de documentos.

RF 12- Cancelar consulta de documentos.

2.6.2 Requisitos no funcionales

Los requisitos o requerimientos no funcionales son propiedades o cualidades que el producto debe tener. En muchos casos este tipo de requerimientos son fundamentales para el éxito del sistema o la solución. Son las características que hacen el producto más atractivo, rápido y confiable.

2.6.2.1 Requerimientos de software:

- ✓ Para los servidores de aplicaciones:

Deben tener como Sistema Operativo Linux, el servidor de aplicaciones JBoss 4.2x Java Runtime Environment (JRE) 1.6, la Máquina virtual de Java (JVM) 1.6 y los servidores Active MQ y Apache Solr.

- ✓ Para los servidores de base de datos:

Debe tener el sistema operativo Linux y el gestor de base de datos PostgreSQL 8.3.

2.6.2.2 Requerimientos de hardware:

- ✓ Para los servidores de aplicaciones y de base de datos:

Debe tener un procesador Intel Dual-Core y contar con 4 GB de memoria RAM.

2.6.3 Modelo de casos de uso del sistema

Los requisitos o condiciones que la solución propuesta debe permitir han sido expresados como casos de uso. Estos constituyen fragmentos de funcionalidades que el sistema ofrece para aportar un resultado de valor para sus actores, que son los que interactúan con él e intercambian información, pero no son parte del sistema. Un diagrama de casos de uso del sistema se representa gráficamente como la relación entre actores y procesos (casos de uso del sistema).

El actor del sistema que se ha identificado es:

Actores del sistema	Funciones
Usuario	Es la persona autorizada que interactúa con el sistema y tiene acceso a la Historia Clínica de los pacientes. Puede ser un médico, enfermero, especialista, etc.

Tabla 1 Actor del sistema.

2.6.3.1 Diagrama de Casos de Uso del Sistema

Proceso Consultar documentos.

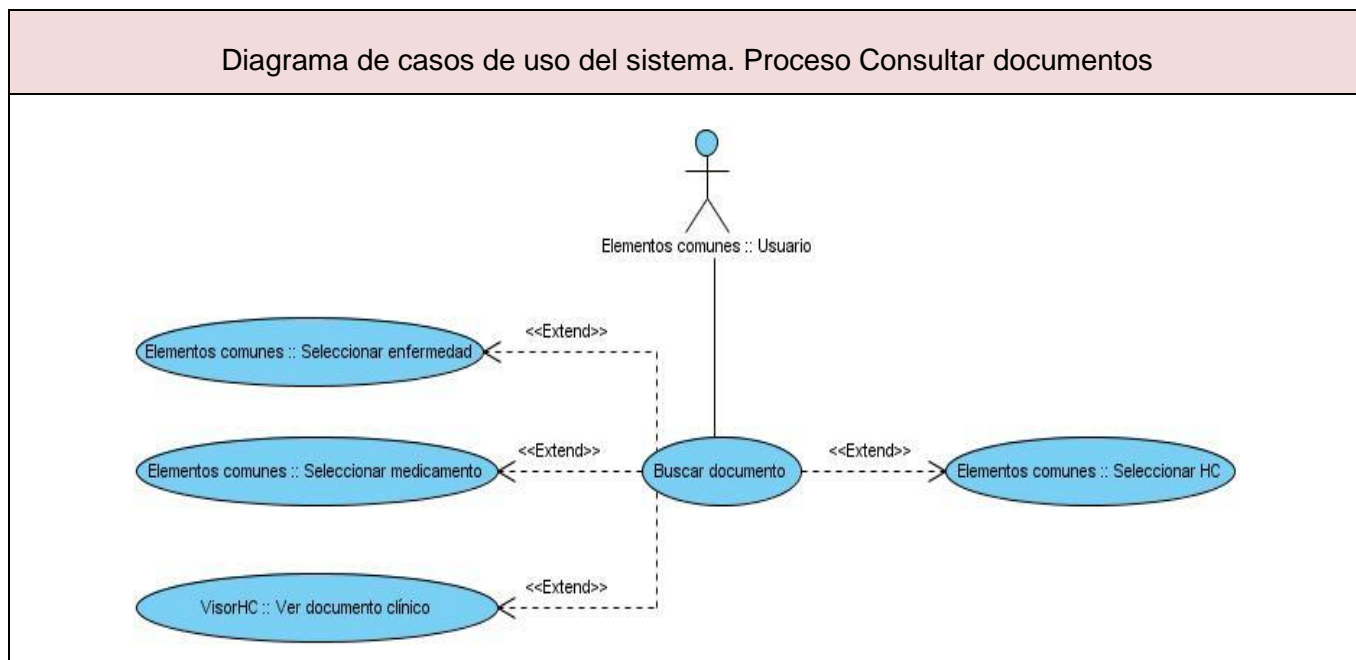


Figura 2.7 Diagrama de casos de uso del sistema. Buscar documento.

2.6.3.1.1 Descripción del caso de uso Buscar documento

El caso de uso Buscar documento inicia cuando el actor accede a la opción de Buscar documento. El sistema muestra una página para buscar el paciente del que se desean consultar los documentos si no se ha seleccionado ya el paciente (en caso de que el usuario acceda a la opción de Buscar documento a través de un módulo del sistema). Luego el usuario elige la opción de buscar por Enfermedad o por Medicamento, el sistema muestra los documentos encontrados y el usuario puede ver los documentos que desea, acción con la cual finaliza el Caso de Uso.

Conclusiones

En el presente capítulo se describieron cada uno de los perfiles de integración que forman parte de la propuesta de solución y permitirán la interoperabilidad entre distintos sistemas de salud como el APS, HIS, RIS y PACS, garantizando la unicidad de los datos clínicos del paciente mediante la HCE única.

Se identificaron los principales conceptos del problema relacionados en el Modelo de Dominio, los procesos a implementar, así como los requisitos funcionales del sistema. Todo esto ayudará a un mejor entendimiento de la solución que se desea construir, permitiendo comenzar el diseño de la solución propuesta.

Capítulo 3. Análisis y diseño

Introducción

Después de haber identificado los procesos que serán implementados, se crean las condiciones para comenzar el diseño del sistema. El flujo Análisis y Diseño que propone RUP, tiene mayor peso en la Fase de Elaboración del producto. Aquí se diseña la arquitectura, se identifican las clases del diseño y la interacción entre objetos. En este capítulo se confeccionará el Diagrama de Clases del Diseño, se describirán cada una de las clases que lo componen, sus métodos y atributos, y se confeccionará el diagrama de secuencia.

3.1 Diagrama de Clases del Diseño

Los Diagramas de Clases del Diseño son un tipo de Diagrama de Estructura y como tal, se enfoca en los elementos que deben existir en el sistema modelado. Estos diagramas son de tipo estático y describen la estructura del sistema, mostrando sus clases, atributos, operaciones y relaciones entre ellas.

Los atributos o características son valores que posee un objeto determinado, mientras que las operaciones o métodos son las actividades que pudiera realizar este objeto.

3.1.1 Descripción de las clases del diseño

Para una mejor comprensión del diseño propuesto se hace una breve descripción de las clases controladoras que intervienen.

- **Indexer:**
En esta clase controladora se realiza la conexión con el servidor Apache Solr y se crea el documento del mismo tipo (Apache Solr), el cual se envía al servidor para ser indexado.
- **MyNamespaceContext:**
Es una clase controladora que se utiliza para definir el Namespace de las expresiones XPath para la consulta al documento CDA.
- **SolrConnection:**
Clase controladora que se encarga de establecer la conexión con el servidor Apache Solr.

- **BusquedaControlador:**
Clase controladora que se encarga de la gestión de las páginas clientes y el procesamiento de los datos que proceden de ellas como los medicamentos y las enfermedades, utilizados para realizar la búsqueda de documentos.
- **Searcher:**
Clase controladora que se encarga de procesar el mensaje HL7 QUPC_IN043100UV para enviar la consulta Apache Solr al servidor.
- **Validattor:**
Clase controladora que se encarga de validar los mensajes HL7.
- **QUPC_IN043100UV:**
Clase controladora que se utiliza para almacenar toda la información contenida en el mensaje HL7 QUPC_IN043100UV.
- **ControlActProcess:**
Clase controladora que se utiliza para almacenar toda la información contenida en el mensaje HL7 QUPC_IN043100UV.

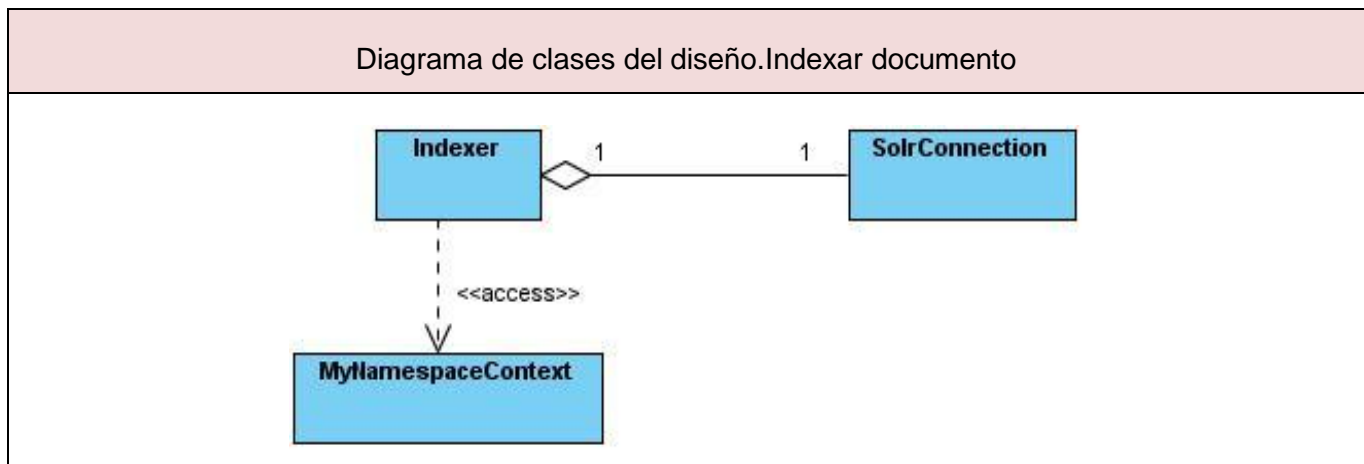


Figura 3.1 Diagrama de Clases del Diseño_ Indexar documento

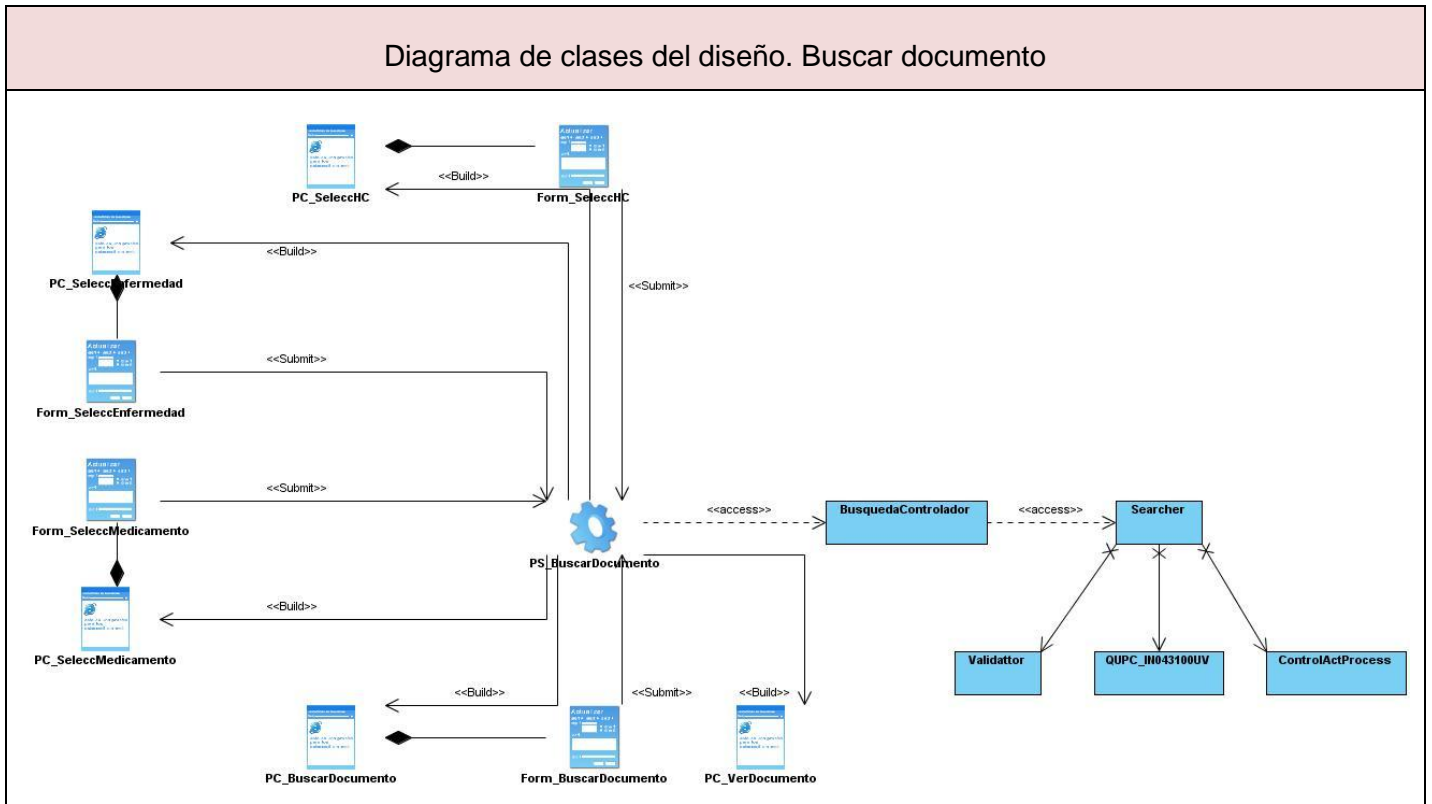


Figura 3.2 Diagrama de Clases del Diseño_Buscar documento

3.2 Diagrama de Interacción

Los Diagramas de Interacción son un tipo de Diagrama de Comportamiento y se enfocan en lo que debe suceder en el sistema modelado. Pueden ser de Secuencia o Colaboración. Los Diagramas de Secuencia muestran precisamente la secuencia detallada de la interacción entre los objetos del diseño a través del tiempo. Contienen detalles de la implementación del escenario que se modela, incluyendo los objetos y clases, así como los mensajes intercambiados entre los objetos.

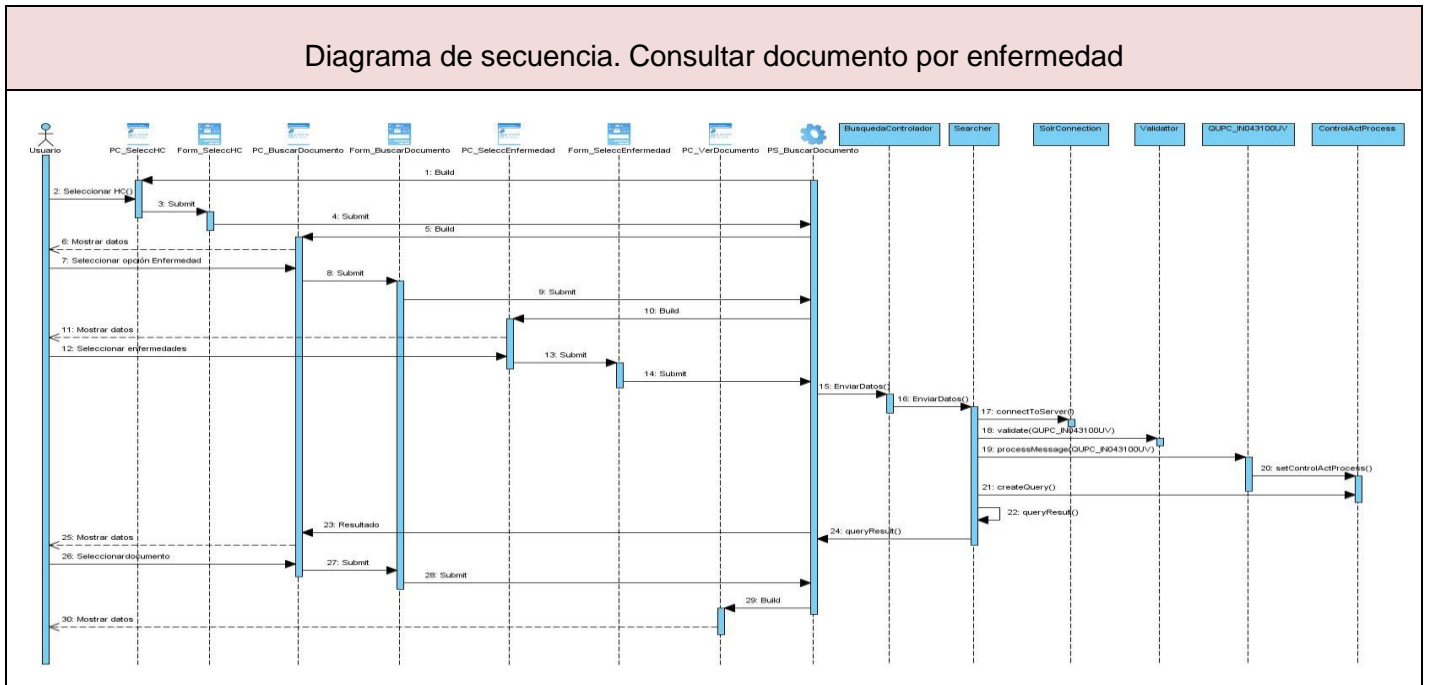


Figura 3.3 Diagrama de Secuencia_Consultar Documento por Enfermedad

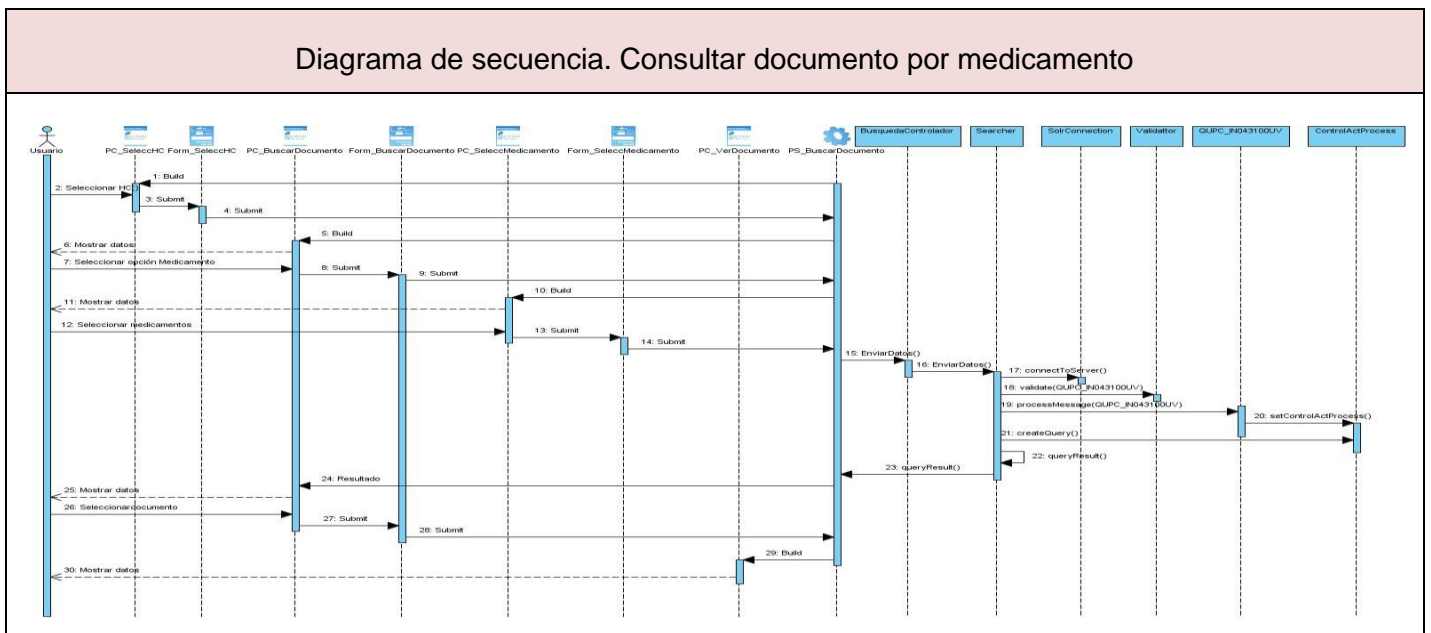


Figura 3.4 Diagrama de Secuencia_Consultar Documento por Medicamento

Conclusiones

En este capítulo se han identificado algunas de las clases del diseño necesarias para la implementación de los procesos del perfil de integración QED, se elaboraron los diagramas de secuencia correspondientes que le proporcionan al programador una idea más clara como fluye este proceso y se describieron las clases encontradas.

Capítulo 4. Implementación

Introducción

Después de realizar el análisis y el diseño de la solución propuesta, se tienen creadas todas las condiciones para comenzar la implementación de la misma. En este capítulo se elaborará el Modelo de Despliegue, el cual muestra la ubicación física de los componentes necesarios para que se puedan ejecutar los procesos comprendidos en cada uno de los perfiles de integración descritos en capítulos anteriores y se describen los nodos que lo componen. También se abordan temas como el tratamiento de errores, la seguridad de la solución implementada y los estándares y estilos de codificación utilizados.

4.1 Modelo de Despliegue

El Modelo de Despliegue es un tipo de Diagrama de Estructura que describe la distribución física del sistema. Se emplea para modelar el hardware utilizado en las implementaciones del sistema y las relaciones entre sus componentes.

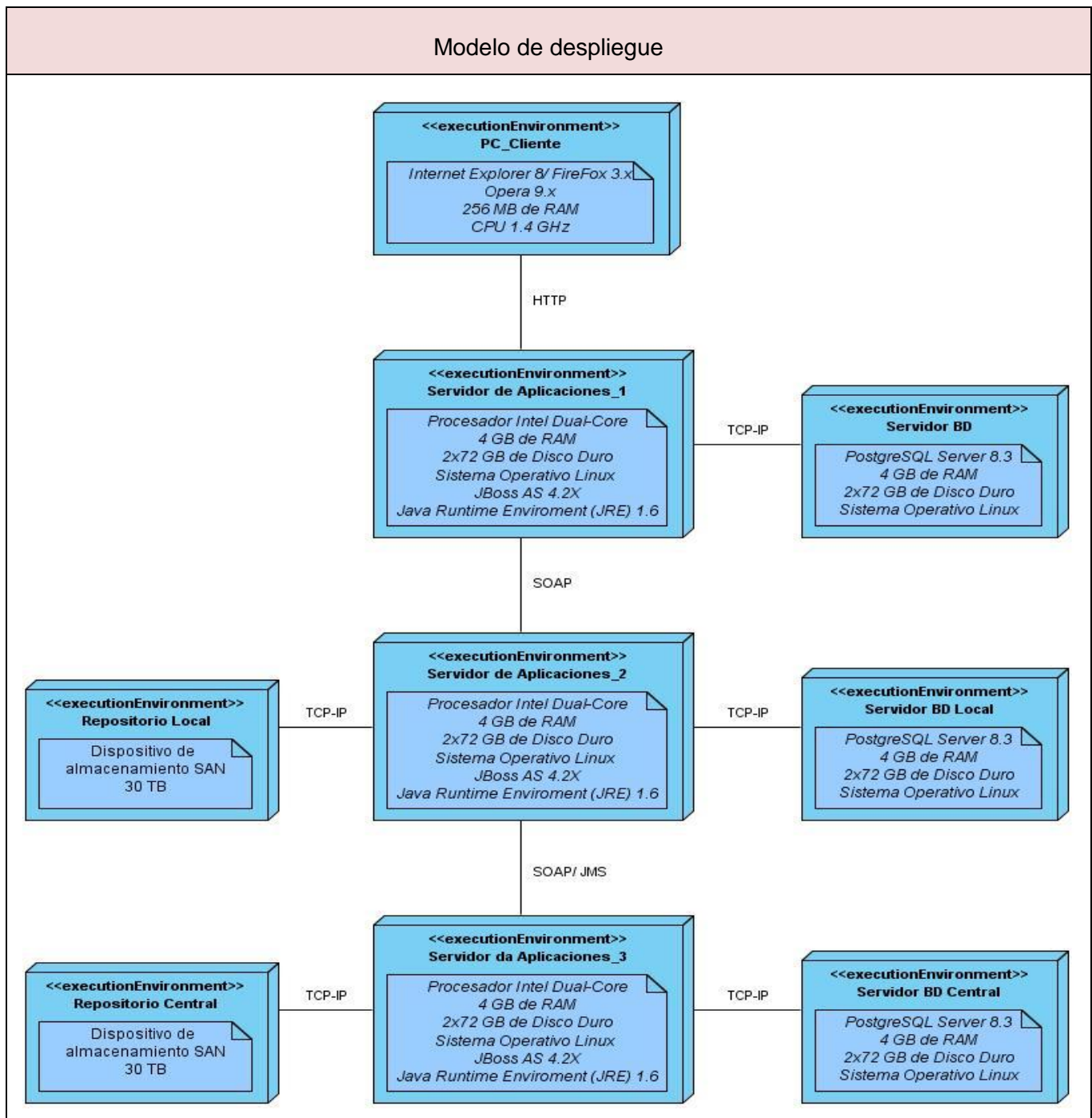


Figura 4.1 Modelo de Despliegue

A continuación se hace una breve descripción de los componentes que interactúan:

- ✓ PC_Cliente: Es la computadora donde el usuario hace uso de los Sistemas de Información. Para esto se conecta al sistema a través del navegador Internet Explorer, FireFox u Opera. Debe tener al menos 256 MB de memoria RAM y contar con CPU de velocidad 1.4 GHz.
- ✓ Servidor de Aplicaciones_1: Servidor a través del cual la PC_Cliente se conecta mediante el protocolo TCP-IP a la Base de Datos (BD) del sistema. Debe tener un procesador Intel Dual-Core, contar con 4 GB de memoria RAM y tener el sistema operativo Linux, el servidor de aplicaciones JBoss 4.2x y Java Runtime Environment (JRE) 1.6.
- ✓ Servidor BD: Servidor donde se almacenan los datos del Sistema de Información. Debe tener el gestor de base de datos PostgreSQL 8.3, 4GB de memoria RAM y el sistema operativo Linux.
- ✓ Servidor de Aplicaciones_2: Es el Bridge o Puente que contiene los procesos de los perfiles de integración PIX, XDS y QED y el servicio de HL7 CTS. Se conecta con el Servidor de Aplicaciones_1 a través del protocolo SOAP y con el Servidor de Aplicaciones _3 a través de SOAP o el servicio de mensajería de Java JMS. Debe tener un procesador Intel Dual-Core, contar con 4 GB de memoria RAM y tener el sistema operativo Linux, el servidor de aplicaciones JBoss 4.2x y Java Runtime Environment (JRE) 1.6.
- ✓ Repositorio Local: Es un dispositivo del almacenamiento SAN (*Storage Area Network*). Se conecta con el Bridge a través del protocolo SCSI, perteneciente a la familia TCP-IP. Debe poseer una capacidad de al menos 30 Terabyte.
- ✓ Servidor BD Local: Repositorio de documentos clínicos pertenecientes a una entidad determinada. Se conecta al Bridge a través del protocolo TCP-IP. Debe tener un procesador Intel Dual-Core y contar con 4 GB de memoria RAM.
- ✓ Servidor de Aplicaciones_3: Servidor del Centro de Datos, se conecta mediante el protocolo SOAP o JMS al Bridge y utilizando TCP-IP al Repositorio Central y la BD Central. Debe tener un procesador Intel Dual-Core, contar con 4 GB de memoria RAM y tener el sistema operativo Linux, el servidor de aplicaciones JBoss 4.2x y Java Runtime Environment (JRE) 1.6.

- ✓ Servidor BD Central: Servidor donde se almacena la información perteneciente al Centro de Datos. Se conecta al Servidor de Aplicaciones_3 a través del protocolo TCP-IP. Debe tener el gestor de base de datos PostgreSQL 8.3, 4GB de memoria RAM y el sistema operativo Linux.
- ✓ Repositorio Central: Es un dispositivo del almacenamiento SAN (*Storage Area Network*). Se conecta con el Bridge a través del protocolo SCSI, perteneciente a la familia TCP-IP. Debe poseer una capacidad de al menos 30 Terabyte.

4.2 Tratamiento de errores

Un error es un fallo que se incluye en un programa, no logrando el producto deseado y afectando al cliente. Las excepciones son el mecanismo recomendado para la propagación de errores que se produzcan durante la ejecución de las aplicaciones. Otro punto de vista es cuando se desarrolla un software, que se intenta proveer de cierta funcionalidad al usuario, si esa funcionalidad no se cumple, se puede decir que el software presenta errores. Estos son detectables por el usuario, lo que influye en la calidad externa del software.

El HIS propone el tratamiento de excepciones principalmente en las regiones críticas de código, que son los fragmentos donde se manipulan datos que son insertados o modificados en la base de datos y en las validaciones de los datos que son insertados por el usuario.

En la implementación del perfil de integración QED la validación del mensaje que inicia el proceso de búsqueda de documentos es uno de los primeros pasos, donde se valida la estructura y el contenido del mensaje y de ocurrir algún error se envía un mensaje y se le informa al usuario.

4.3 Seguridad

La seguridad es de gran importancia para cualquier Sistema de Información y más si se trata de la salud de las personas y se manipula la Historia Clínica de las mismas, la cual constituye un documento legal y por ende, toda medida de seguridad que se tome respecto a ella será poca.

4.4 Estrategias de codificación. Estándares y estilos a utilizar

Un estándar de codificación comprende todos los aspectos de la generación de código y deber ser práctico. Un código fuente en su totalidad debe ser de fácil entendimiento y reflejar un estilo armonioso, como si un único programador lo hubiese programado de una sola vez. Al comenzar un proyecto de software, es necesario establecer un estándar de codificación para asegurarse que todos los programadores del proyecto trabajen de forma coordinada. Este debería indicar como operar con la base del código existente en caso de realizar modificaciones y/o mantenimiento al sistema.

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. Usar técnicas de codificación sólidas y realizar buenas prácticas de programación con vistas a generar un código de alta calidad es de gran importancia para la calidad del software y para obtener un buen rendimiento.

Algunas de las especificaciones que se utilizan en el código son:

- Deben escribirse comentarios al inicio de cada clase y método, con el objetivo de brindar una breve descripción de los propósitos de cada funcionalidad.
- Los nombres de las clases deben ser lo más simple y sugerente posible, utilizando palabras completas y abreviaturas conocidas.
- Los métodos deberán ser verbos, en mayúsculas y minúsculas, con la letra inicial mayúscula y la primera letra de las palabras internas en mayúscula.
- Los nombres de las variables deben ser cortos y significativos, de manera que se entienda con facilidad su significado. Se deben evitar las variables de una sola letra, excepto para las temporales de corto uso.
- Todos los nombres de variables de instancia o de clase deben estar constituidos por palabras con la primera letra de la primera palabra en minúscula y la primera letra de las palabras internas en mayúscula.

Entre las restricciones que se utilizan en el diseño de interfaz se encuentran:

- En los casos en que se requiera seleccionar un rango de fechas este se pondrá mediante las etiquetas **Desde:** y **Hasta:** asociadas a los calendarios correspondientes. En ambos casos se utilizará para ello el *rich:calendar*.
- El *simpleTogglePanel* asociado a los criterios de búsqueda se titulará **Criterios de búsqueda**.
- Cuando en una misma página se tiene una selección múltiple esta se realizará mediante *checkBox*, manteniéndose en la lista de origen marcados aquellos elementos que fueron seleccionados y que se encuentran en la lista de elementos seleccionados.
- Los editores de texto se pondrán alineados a la izquierda. Los asociados con etiquetas se pondrán debajo de la etiqueta, alineados ambos a la izquierda. Tendrán una longitud de 150 píxel.
- La sección para mostrar los datos del paciente será nombrada **Datos generales del paciente**. La foto que aparecerá tendrá un tamaño de 74x74 píxel, en el caso de mostrarse los datos del paciente. Los datos a mostrar en la primera columna son en la primera fila: "**Nombre:**", en la segunda fila: "**Primer apellido:**", en la tercera fila "**Segundo apellido:**". Los datos de la segunda columna son en la primera fila "**Carné Identidad:**", en la segunda fila "**Fecha de nacimiento:**" y en la tercera fila "**Sexo:**" [Masculino/Femenino/Desconocido/No especificado].

Conclusiones

En el presente capítulo se ha definido el Modelo de Despliegue que constituye la distribución física de los componentes de la solución que se propone. Además se abordaron temas como el tratamiento de errores, la seguridad y los estándares de codificación que son utilizados.

Conclusiones

Con la realización del presente trabajo y el cumplimiento de las tareas propuestas se pueden plantear las siguientes conclusiones:

- Durante el estudio realizado se determinó que los sistemas analizados no poseen interoperabilidad y existe poca información sobre sus funcionalidades, lo que impide hacer una correcta valoración de su factibilidad, evidenciando la necesidad de un Centro de Datos para la gestión de la Historia Clínica Electrónica centralizada.
- El análisis de la comunicación entre los sistemas de información y el Centro de Datos permitió la definición de los procesos implementados reflejando la necesidad de utilizar una Arquitectura Orientada a Servicios, la cual mejora la interoperabilidad, portabilidad y flexibilidad de la solución propuesta.
- La fragmentación, desactualización y el difícil acceso a los documentos clínicos del paciente originaron la necesidad de la implementación del proceso Indexar documento y el perfil de integración QED que permiten adicionar nuevos documentos al índice, garantizando durante el proceso de consulta facilidad, rapidez y eficiencia en las mismas.
- La estructura de los componentes físicos para la puesta en práctica de la solución garantiza una mejor ejecución de cada uno de los procesos de comunicación pertenecientes a los perfiles de integración que fueron objeto de estudio.

De esta forma se cumplió con el objetivo del trabajo de diploma, por lo que se obtuvo una solución informática, que permite mejorar los procesos relacionados a la atención médica, garantizando la centralización y unicidad de los datos clínicos del paciente.

Recomendaciones

Con el objetivo de complementar y mejorar la solución propuesta y ofrecer un mejor servicio a usuarios y pacientes se hacen las siguientes recomendaciones:

- ✓ Incorporar a la funcionalidad Buscar documentos la Búsqueda libre, donde el usuario pueda hacer una búsqueda de palabras o frases que desea encontrar en un documento clínico.
- ✓ Incorporar a la funcionalidad Buscar documentos la implementación de la búsqueda de documentos teniendo en cuenta los resultados obtenidos en los exámenes de laboratorio clínico.
- ✓ Adicionar un servicio para la gestión de la terminología médica que apoye y facilite el trabajo del usuario.

Referencias Bibliográficas

1. **IEEE**. Recommended Practice for Architectural Description of Software-Intensive Systems. [En línea] 2000. <http://www.enterprise-architecture.info/Images/Documents/IEEE%201471-2000.pdf>.
2. **Reynoso, Carlos Billy**. Introducción a la Arquitectura de Software. [En línea] <http://www.bing.com/search?q=spanish%20msdn%20arquitectura%20roadmap%20arq%20intro%20mspx&FORM=BMMENO>.
3. Centro de Formación Avanzada de Tecnologías. [En línea] http://it-institute.org/index.php?option=com_content&task=view&id=5&Itemid=26. 16.
4. **Luis Carlos Díaz, Angela Carrillo y Deicy Alvarado**. *IS, RUP y UML en el Contexto de ADOO Análisis y Diseño OO*. 2008.
5. **Corporate Headquarters**. *BisAgi Process Modeler*.
6. Ciberaula. [En línea] http://java.ciberaula.com/articulo/que_es_java/. 1.
7. Conozca más sobre la tecnología Java. [En línea] <http://java.com/es/about/>.
8. **Rasmussen, Anders Ingeman**. Osalt. com Código abierto como alternativa. [En línea] 2009. <http://www.osalt.com/es/jboss>.
9. Master Magazine. [En línea] 2004. <http://www.mastermagazine.info/articulo/9565.php>. 2.
10. World Wide Web Consortium. [En línea] 2008. <http://www.w3c.es/Divulgacion/GuiasBreves/ServiciosWeb>. 3.
11. **Álvarez, Sara**. Desarrollo Web. [En línea] <http://www.desarrolloweb.com/articulos/sistemas-gestores-bases-datos.html>.
12. Ídem 11. [En línea]
13. **HL7**. Introducción al mundo de los estándares. [En línea] www.hl7.org.ar. 4.
14. Ídem 13. [En línea]
15. **HL7**. Introducción a HL7 v2. [En línea] www.hl7.org.ar. 6.
16. Ídem 15. [En línea]
17. Ídem 15. [En línea]
18. **HL7**. Clinical Document Architecture. [En línea] www.hl7.org.ar. 9.
19. IHE España. [En línea] 2009. www.ihe-e.org. 10.

20. Free Download Manager. [En línea]
http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_%28M%C3%8D%29_14720_p/. 15.
21. **Asociación JavaHispano**. JavaHispano. [En línea] 2010.
http://www.javahispano.org/contenidos/es/apache_solr_1_4_motor_de_busqueda_100_java/ .
22. **Borillo, Ricardo**. El rincón práctico de los estándares web. [En línea] <http://xml-utils.com/2006/10/21/indexacion-de-contenido-con-apache-lucene/> .
23. **Leodan Vega Izaguirre, Alejandro Planos González**. *Alas RIS Sistema de gestión de información radiológica*. 2008.
24. **IHE-Europe aisbl**. Changing the way healthcare connects. [En línea] Marzo de 2009. <http://www.ihe-e.org/docweb/faq2009.pdf>.
25. *Changing the way healthcare connects*. **Río, Javier Quiles del**. 2008.
26. **ACC, HIMSS and RSNA**. *Patient Care Coordination Technical Framework Volume I*. 2007.
27. **Mayo Clinic**. Mayo Clinic. [En línea] <http://informatics.mayo.edu/LexGrid/index.php?page=ctsspec> .

Bibliografía

1. **ACC, HIMSS and RSNA.** Patient Care Coordination Technical Framework Volume I. [En línea] 2007. <http://www.sefm-ihe.info/> .
2. Adictos al trabajo. [En línea] 2010. <http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=webservicesAxis>.
3. Calidad Proactiva. [En línea] <http://www.proactiva-calidad.com/java/rmi/introduccion.html>.
4. *Changing the way healthcare connects.* . **Río., Javier Quiles del.** 2008.
5. Cibernética. [En línea] http://www.cibernetia.com/manuales/servicios_web/4_wsdl.php .
6. Ciencias Computacionales. [En línea] <http://genesis.uag.mx/posgrado/revistaelect/compu/com002.htm>.
7. Connecting for Health. [En línea] 2010. <http://www.connectingforhealth.nhs.uk/systemsandservices/pacs/learn>.
8. **Corporate Headquarters.** BisAgi Process Modeler. [En línea] <http://manual-es.com/Business->.
9. *Cross Enterprise Document Sharing (XDS)* . **Saccavini, Claudio.**
10. Desarrollo Web. [En línea] <http://www.desarrolloweb.com> .
11. Dirección de Informática. [En línea] 2008. www.di.sld.cu .
12. **Donato., Ing. Gerardo Morgade.** *ClioBD. Sistema de control de versiones para bases de datos.* Junio de 2009.
13. *Expectación excesiva acerca de la pronta implantación de la historia clínica electrónica.* **Gérvas, Juan.** 2000, Editorial.
14. **Flores, Milagro Díaz.** Practicante de todo. [En línea] <http://www.usmp.edu.pe/publicaciones/boletin/fia/info49/articulos/RUP%20vs.%20XP.pdf>.
15. **González, Héctor Suárez.** Manual de Hibernate. [En línea] 21 de Marzo de 2003. <http://www.javahispano.org/contenidos/archivo/77/ManualHibernate.pdf>.
16. **Hewlett-Packard Development Company.** HP. [En línea] 2009. http://h41131.www4.hp.com/es/es/press/Sanitas_implanta_la_historia_clnica_electrnica_en_sus_hospitales_gracias_a_la_tecnologa_de_HP.html.
17. **Lanza, José Luis Alonso.** *La historia clínica electrónica: ideas, experiencias y reflexiones.*

18. **Mayo Foundation.** Mayo Clinic. [En línea] 2009.
<http://informatics.mayo.edu/LexGrid/index.php?page=ctsspec>.
19. *Metodologías De Desarrollo De Software* . **Sanchez, María A. Mendoza.** Junio 7 del 2004.
20. **Microsoft Corporation.** .Net Framework Developer Center. [En línea] 2010.
<http://social.msdn.microsoft.com/Forums/en-US/asmxandxml/thread/435f43a9-ee17-4700-8c9d-d9c3ba57b5ef>.
21. **Microsoft Corporation.** La Arquitectura Orientada a Servicios (SOA) de Microsoft aplicada al mundo real. [En línea] Diciembre de 2006.
<https://www.microsoft.com/latam/servidores/applications.mspix>.
22. **Miller., Ing. Vladimir Guerra.** *Estándar de formato para la implementación de la Historia Clínica Electrónica en Cuba.* . 2008.
23. **Open sistemas.** Open sistemas. [En línea] 2008.
http://www.opensistemas.com/sectores/sanidad_y_farmacia_copy_1/servidor_de_aplicaciones/.
24. pgAdmin PostgreSQL Tools. [En línea] <http://www.pgadmin.org/>.
25. **Programación en castellano.** Java en Castellano. [En línea] 1999.
<http://www.programacion.com/java/tutorial/hibernate/>.
26. **Rasmussen, Anders Ingeman.** Osalt.com. [En línea] 2009. <http://www.osalt.com/es/jboss>.
27. **Red Hat.** Jboss Community. [En línea] <http://www.jboss.com/>.
28. **Red Hat.** Seam Framework. [En línea] <http://seamframework.org/> .
29. **Romero., Lic. Maykell Sánchez.** *Infraestructura de software para el almacenamiento y consulta de la Historia Clínica Electrónica del sistema alas HIS.* Enero 2010.
30. Sparx Systems. [En línea] 2009 . <http://www.sparxsystems.com.ar/> .
31. Tecnologías de la Sociadead de la Información. [En línea] <http://tundidor.com/blog/?tag=axis2> .
32. **The Apache Software Foundation.** The Apache Software Foundation . [En línea]
<http://tomcat.apache.org/>.
33. The Apache Software Fundation. [En línea] 2009.
http://ws.apache.org/axis2/1_5_1/userguide.html#whatis.
34. The Developer Portal for SOA. [En línea] 2010. <http://wso2.org/library/1719c>.
35. **The Eclipse Foundation.** Eclipse. [En línea] 2010. <http://www.eclipse.org/>.
36. Visual Paradigm. [En línea] <http://www.visual-paradigm.com/product/vpuml/editions/enterprise.jsp>.
37. W3C. [En línea] 2001. http://www.w3.org/TR/wSDL#_service.

38. **IEEE**. Recommended Practice for Architectural Description of Software-Intensive Systems. [En línea] 2000. <http://www.enterprise-architecture.info/Images/Documents/IEEE%201471-2000.pdf>.
39. **Reynoso, Carlos Billy**. Introducción a la Arquitectura de Software. [En línea] <http://www.bing.com/search?q=spanish%20msdn%20arquitectura%20roadmap%20arq%20intro%20m spx&FORM=BMMENO>.
40. Centro de Formación Avanzada de Tecnologías. [En línea] http://it-institute.org/index.php?option=com_content&task=view&id=5&Itemid=26.
41. **Luis Carlos Díaz, Angela Carrillo y Deicy Alvarado**. *IS, RUP y UML en el Contexto de ADOO Análisis y Diseño OO*. 2008.
42. **Corporate Headquarters**. *BisAgi Process Modeler* .
43. Ciberaula. [En línea] http://java.ciberaula.com/articulo/que_es_java/.
44. Conozca más sobre la tecnología Java. [En línea] <http://java.com/es/about/>.
45. **Rasmussen, Anders Ingeman**. Osalt. com Código abierto como alternativa. [En línea] 2009. <http://www.osalt.com/es/jboss>.
46. Master Magazine. [En línea] 2004. <http://www.mastermagazine.info/articulo/9565.php>.
47. World Wide Web Consortium. [En línea] 2008. <http://www.w3c.es/Divulgacion/GuiasBreves/ServiciosWeb>.
48. **Álvarez, Sara**. Desarrollo Web. [En línea] <http://www.desarrolloweb.com/articulos/sistemas-gestores-bases-datos.html>.
49. **HL7**. Introducción al mundo de los estándares. [En línea] www.hl7.org.ar.
50. **HL7**. Introducción a HL7 v2. [En línea] www.hl7.org.ar.
51. **HL7**. Clinical Document Architecture. [En línea] www.hl7.org.ar.
52. IHE España. [En línea] 2009. www.ihe-e.org.
53. Free Download Manager. [En línea] http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_%28M%C3%8D%29_14720_p/.
54. **Asociación JavaHispano**. JavaHispano. [En línea] 2010. http://www.javahispano.org/contenidos/es/apache_solr_1_4_motor_de_búsqueda_100_java/ .

55. **Borillo, Ricardo**. El rincón práctico de los estándares web. [En línea] <http://xml-utils.com/2006/10/21/indexacion-de-contenido-con-apache-lucene/> .
56. **Leodan Vega Izaguirre, Alejandro Planos González**. *Alas RIS Sistema de gestión de información radiológica*. 2008.
57. **IHE-Europe aisbl**. Changing the way healthcare connects. [En línea] Marzo de 2009. <http://www.ihe-e.org/docweb/faq2009.pdf>.
58. *Changing the way healthcare connects*. **Río, Javier Quiles del**. 2008.
59. **ACC, HIMSS and RSNA**. *Patient Care Coordination Technical Framework Volume I*. 2007.
60. **Mayo Clinic**. Mayo Clinic. [En línea] <http://informatics.mayo.edu/LexGrid/index.php?page=ctsspec> .

Glosario de Términos

Berkeley Software Distribution (BSD): Licencia de software libre que permite el uso de código fuente en software no libre, muy cercana al dominio público.

Compilador Jasper: Compilador que utiliza el servidor de aplicaciones Tomcat para convertir las páginas JSPs en servlests.

Diagrama de Procesos de Negocio (Business Process Diagram, BPD): Es un Diagrama diseñado para representar gráficamente la secuencia de todas las actividades que ocurren durante un proceso, incluye además toda la información que se considera necesaria para el análisis.

Formato de texto enriquecido o Rich Text Format (RTF): Es una forma particular para dar formato a un texto. El RTF es un formato de texto compatible, en el sentido que puede ser migrado desde y hacia cualquier versión de Word, e incluso muchos otros procesadores de textos y de aplicaciones programadas.

Interfaz de Programación de Aplicaciones o Application Programming Interface (API): Conjunto de funciones o procedimientos que ofrece cierta biblioteca para ser usado por otro software.

Iteración: Es un ciclo de desarrollo completo dando como resultado una entrega de producto ejecutable (interna o externa).

Lenguaje de Marcas de Hipertexto o HyperText Markup Language (HTML): Es el lenguaje de marcado predominante para la construcción de páginas web. Es usado para describir la estructura y el contenido en forma de texto, así como para complementar el texto con objetos tales como imágenes.

Lenguaje de Marcas Extensible o Extensible Markup Language (XML): Es un formato que permite la lectura de datos a través de diferentes aplicaciones.

Lenguaje Unificado de Modelado o Unified Modeling Language (UML): Está compuesto por diferentes elementos gráficos que se combinan para conformar diagramas.

Máquina Virtual de Java o Java Virtual Machine (JVM): Una de las partes fundamentales de la plataforma Java, interpreta y ejecuta instrucciones expresadas en código binario, que es generado por el compilador del lenguaje Java.

Servlets: Programa escrito en Java que se ejecuta en el marco de un servicio de red, (un servidor HTTP, por ejemplo), y que recibe y responde a peticiones de uno o más clientes.

Simple Object Access Protocol (SOAP): Es un protocolo estándar que permite la comunicación y la interoperabilidad entre diversas aplicaciones web desarrolladas bajo tecnologías diferentes.

Small Computers System Interface (SCSI): Sistema de Interfaz para Pequeñas Computadoras, es una interfaz estándar para la transferencia de datos entre distintos dispositivos del bus de la computadora.

Storage Area Network (SAN): Una SAN es una red de almacenamiento dedicada que proporciona acceso de nivel de bloque a LUNs. Un LUN, o número de unidad lógica, es un disco virtual proporcionado por la SAN. El administrador del sistema tiene acceso al disco virtual como si fuera un disco directamente conectado a la misma. El administrador puede particionar y formatear el disco en cualquier medio que él elija.

Tecnologías de la Información y las Comunicaciones (TIC): Son aquellas herramientas computacionales e informáticas que procesan, almacenan, sintetizan, recuperan y presentan información representada de la más variada forma. Son de carácter innovador y creativo, pues dan acceso a nuevas formas de comunicación. Se relacionan con mayor frecuencia con el uso de la Internet y la informática.

Web Services Description Language (WSDL): Está basado en un formato XML (Extensible Markup Language), permite describir la interfaz pública de los servicios web, la forma de comunicación, así como sus características, localización y aquellos parámetros y métodos que soporta.

Anexos

Anexo 1. Diagrama de proceso Buscar paciente.

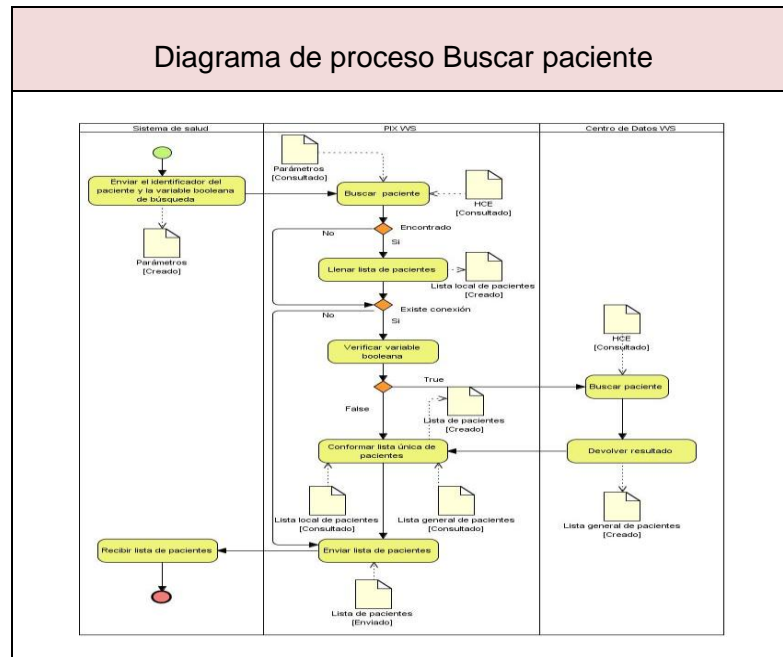


Figura 1 Diagrama de proceso Buscar paciente

Anexo 2. Diagrama de proceso Registrar paciente.

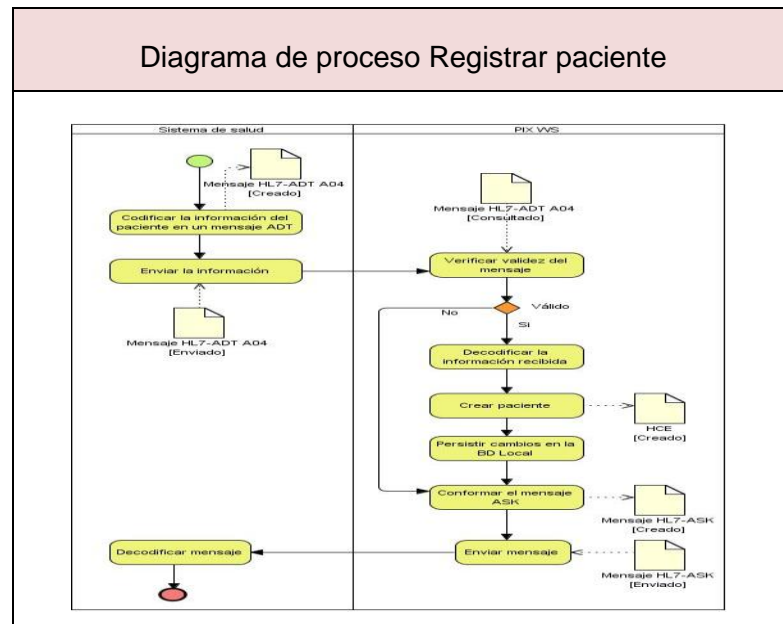


Figura 8 Diagrama de proceso Registrar paciente

Anexo 3. Diagrama de proceso Actualizar paciente.

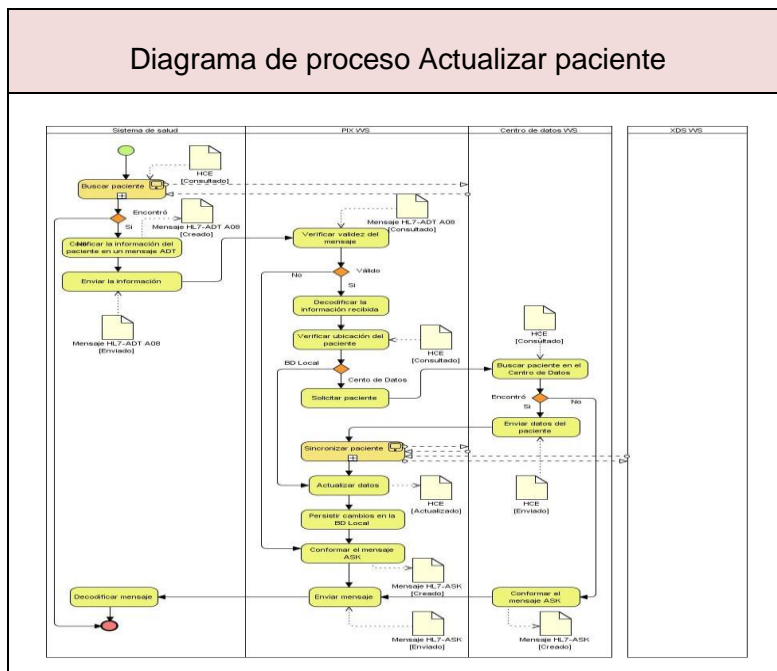


Figura 3 Diagrama de proceso Actualizar paciente

Anexo 4. Diagrama de proceso Mezclar pacientes.

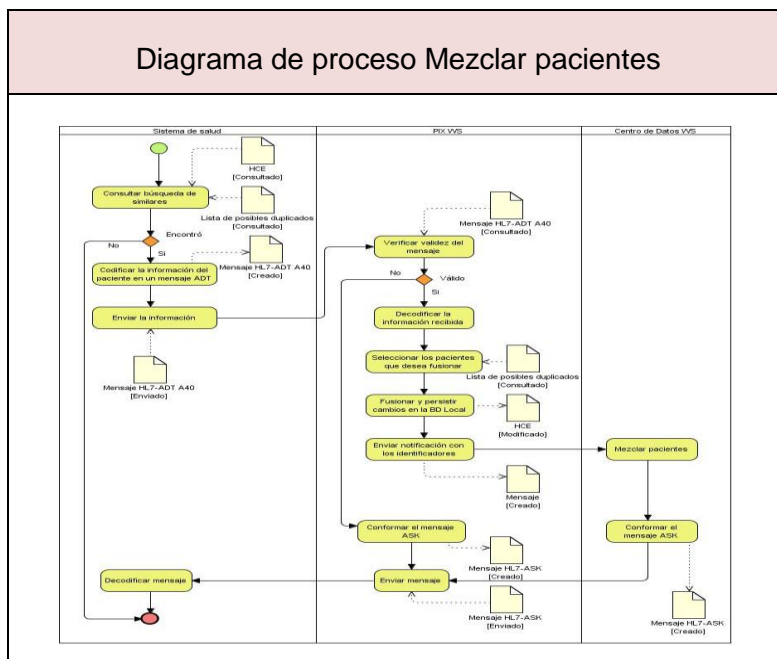


Figura 4 Diagrama de proceso Mezclar pacientes

Anexo 5. Diagrama de proceso Sincronizar paciente.

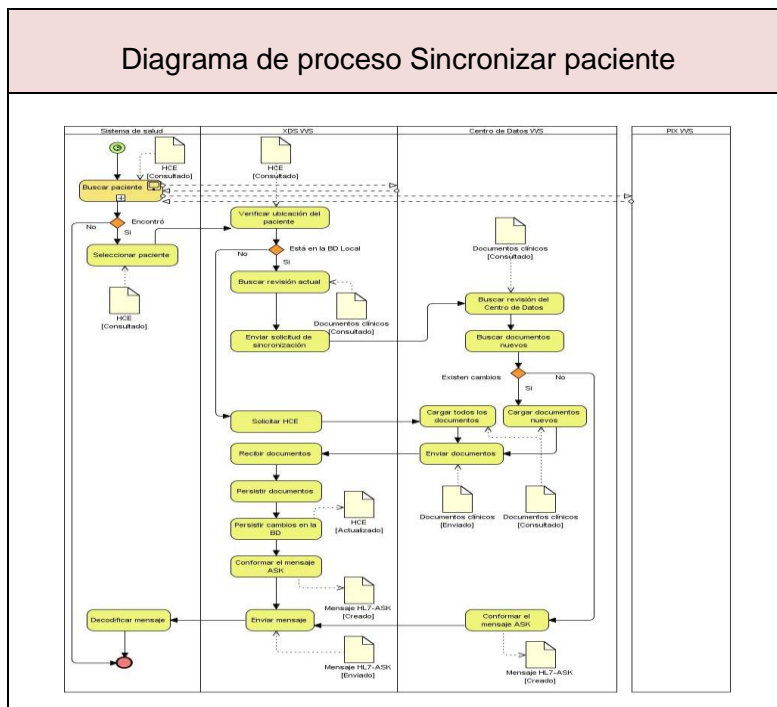


Figura 5 Diagrama de proceso Sincronizar paciente

Anexo 6. Diagrama de proceso Subir cambios.

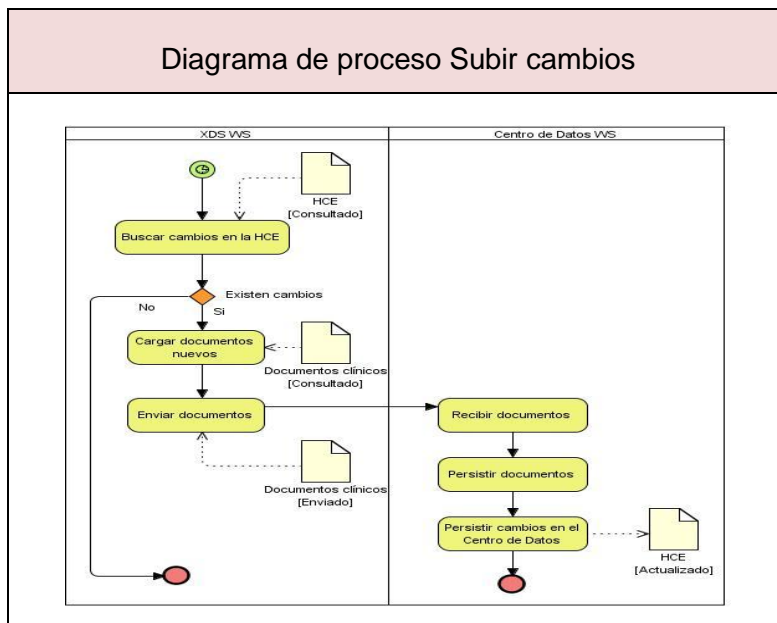


Figura 6 Diagrama de proceso Subir cambios