



Universidad de las Ciencias Informáticas

Facultad 2

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

**Título: Propuesta de integración basada en servicios terminológicos para los
productos del Centro de Informática Médica**

Autores: Dianiset Remedios Reynaldo

Dariel Conde Villa

Tutor: MSc. Karel Fernández Cedeño

Consultante: MSc. Yovannys Sánchez Corales

Ing. José Manuel Torres Amador

La Habana, junio 2014

“Año 56 de la Revolución”



“Entre los grandes placeres que nos brinda la vida, está el superar las dificultades paso a paso, escalando uno a uno los peldaños del éxito, concibiendo nuevos deseos y viendo los realizados”

Samuel Johnson

Declaración de Autoría

Declaramos ser los únicos autores del presente trabajo, autorizando a la Universidad de las Ciencias Informáticas (UCI) y al Departamento de Tecnologías, Integración y Estándares, los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmamos la presente a los X días del mes de X del año 2014.

Dianiset Remedios Reynaldo

Dariel Conde Villa

Autor

Autor

MSc. Karel Fernández Cedeño

Tutor

Datos de Contacto

MSc. Karel Fernández Cedeño (kfernandez@uci.cu). Graduado de Ingeniero en Ciencias Informáticas del año 2008 en la Universidad de las Ciencias Informáticas de Cuba. Ha impartido las asignaturas de Técnicas de programación, Ingeniería de Software I, Ingeniería de Software II, Software libre, entre otras con la categoría de Profesor Instructor. Se desempeña actualmente como Analista Principal del Centro de Informática Médica.

Resumen

La utilización de computadoras en la administración de la información clínica de los pacientes ha puesto en evidencia la complejidad del manejo del vocabulario médico. Los Servidores de Terminologías, son herramientas de software que brindan servicios a otras aplicaciones sanitarias para centralizar el control del vocabulario médico, permitiendo que la información clínica se almacene de una forma estándar.

En el presente trabajo se propone el uso de servicios terminológicos como una alternativa de interoperabilidad entre las aplicaciones del Centro de Informática Médica. Para lograrlo se propone implementar el perfil Compartir Conjuntos de Valores que describe la Integración de Empresas Sanitarias para integrar el servidor de terminologías médicas Apelon DTS a las soluciones del Centro de Informática Médica.

Para comprobar su utilidad y pertinencia se escogió el proceso de codificación en la persistencia de información relevante de los pacientes en la Arquitectura de Documentos Clínicos, logrando así alcanzar el Nivel 3 dentro de la calificación que emite el *Health Level Seven*. Para lograrlo se desarrolló una interfaz de comunicación capaz de proporcionarle a los sistemas del Centro de Informática Médica el acceso a las terminologías necesarias para realizar la persistencia de información codificada. El proceso de validación no solo permitió demostrar la validez de la investigación, sino también, la obtención de la interfaz de programación Apelon_DTS para la comunicación de las aplicaciones del Centro con el servidor de terminologías propuesto.

Palabras claves: codificación, interoperabilidad, terminología, vocabulario.

1 Índice

Introducción1

Capítulo 1: Fundamentos teóricos-metodológicos relacionados con la investigación.....5

 1.1 Análisis del Sistema de Información en Salud5

 1.2 Estándares de terminologías médicas8

 1.3 Especificaciones de servidores de términos médicos10

 1.4 Perfiles de Integración IHE.....16

Capítulo 2: Diseño de la solución.....20

 2.1 Modelo de Dominio.....20

 2.2 Propuesta del sistema.....21

 2.3 Especificación de los requisitos de la solución.23

 2.4 Diagrama de Interacción.....27

Capítulo 3: Validación de la propuesta de solución29

 3.1 Implementación de la API de comunicación.....29

 3.2 Validación de la propuesta de solución37

Conclusiones43

Recomendaciones44

Referencias Bibliográficas45

Bibliografía48

Glosario de Términos-.....50

Introducción

El vertiginoso desarrollo de la informática y las comunicaciones ha impactado de forma positiva en los sectores de la sociedad moderna. El uso de la computación en la medicina forma parte de las aplicaciones más comunes e importantes desde hace tres décadas. Esto ha permitido al sector de la salud, contar no solo con métodos novedosos y eficaces de gestión administrativa en consultas, hospitales y centros de investigación biomédica, sino también disponer de complejos *software* que reducen la posibilidad de error en el diagnóstico de las enfermedades, y que aceleran su formulación (1).

A pesar de tener un componente tecnológico de informática e ingeniería, el estudio del conocimiento médico y su aplicación en el manejo del paciente a través de sistemas de información requiere de una extensa base de formación médica y clínica. A esto se suman las posibilidades que brinda la tecnología de distribuir esta información en forma rápida y efectiva.

Los Sistemas de Información en Salud (SIS) requieren que se registren los datos clínicos con el nivel de detalle adecuado para los fines asistenciales, de gestión e investigación. (2) El almacenamiento de texto, en forma de lenguaje natural, es poco útil al momento de recuperar y analizar la información contenida a pesar de ser el medio más expresivo y sencillo de utilizar para ingresar la información. La utilidad clínica de un texto permanece limitada por el hecho que mucha información de gran valor se pierde por la dificultad de procesar la parte narrativa (3), en este contexto nace la codificación médica.

La codificación de términos médicos es el proceso de transformación de las descripciones de los diagnósticos y procedimientos médicos en códigos médicos estandarizados. Los diagnósticos y procedimientos generalmente se toman a partir de una variedad de fuentes en el registro de cuidado de la salud, como la transcripción de las notas del médico, resultados de laboratorio, resultados radiológicos y otras fuentes de información (4).

Los códigos de diagnóstico se utilizan para identificar las enfermedades y otros problemas de salud en los pacientes. Por ejemplo, en la Clasificación Estadística Internacional de Enfermedades y Problemas Relacionados con la Salud, el código I47.1 es la taquicardia supra ventricular; sin embargo, hay otros

conceptos clínicos que también se clasifican aquí, entre ellos se encuentran la taquicardia paroxística auricular, taquicardia de la unión paroxística, taquicardia auricular y taquicardia nodal (5).

Cada terminología médica abarca un dominio y está diseñada para el uso en un ámbito distinto. No existe una que sea universal, ya que tampoco existe una estructura que permita representar completamente todo el conocimiento, incluso si se limita a tratar únicamente con el vocabulario médico. Es necesario diseñar estrategias de control del vocabulario para que la información clínica almacenada en el SIS se pueda reutilizar, ya sea con fines estadísticos, de gestión o en herramientas de soporte a la toma de decisiones orientadas a mejorar la calidad de la atención de los pacientes.

Una solución muy efectiva la constituye el uso Servidores de Terminologías (ST) que se definen como una herramienta de software que brinda servicios a otras aplicaciones para centralizar el control del vocabulario. La utilización de estos “Servicios terminológicos” comunes a todos los niveles del sistema de información hospitalaria permite que la información clínica se almacene de forma estándar y reutilizable, no importa el punto de origen de la misma (6).

Actualmente en el CESIM coexisten varias aplicaciones sanitarias y cada una utiliza los codificadores que necesita para el registro de la información clínica en el área que es desplegada. Cuando un paciente es atendido en diferentes áreas de una institución de salud, el estado, sus patologías y las técnicas que se le aplican, son reflejados de forma distinta e independiente utilizando las diferentes terminologías médicas con las que interactúan las aplicaciones, provocando que no se entiendan entre sí cuando intercambian esos datos.

Una solución a este problema sería que cada aplicación almacenara los codificadores utilizados por otros sistemas para poder entender la información que se genera fuera de su ámbito, pero esto trae como consecuencia una sobrecarga de la BD y la duplicidad innecesaria de codificadores. A esta situación también se suma que a pesar de que dos aplicaciones coincidan en el uso de un codificador no existe garantía de que también utilicen la misma versión de este, lo que trae como consecuencia que en un momento dado no puedan entender la información que comparten debido a que en el cambio de una versión a otra puede variar la estructura o la cantidad de términos médicos del codificador que utilizan.

Por otra parte el constante crecimiento del vocabulario médico como consecuencia de los nuevos hallazgos conlleva a la necesidad de actualizar los codificadores con nuevos términos. Por lo que si en el CESIM se pretendiera actualizar la versión de un codificador sería necesario realizar esta operación tantas veces como sistemas que utilicen el codificador.

Aun cuando fuera muy costoso, en términos de tiempo y esfuerzo de desarrollo, esto podría realizarse y sería posible establecer intercambios efectivos de comunicación con otras aplicaciones del Centro, no siendo así en cuanto a la integración con una aplicación de terceros se refiere, ya que la misma tendría que incorporar los codificadores utilizados por todas las soluciones de centro para poder comunicarse de manera efectiva con las mismas.

Teniendo en cuenta la situación anterior se define como **problema científico**: la inclusión de los codificadores sanitarios internacionales en cada solución del Centro de Informática Médica, afecta la interpretación de la información que intercambian. Definiéndose como **objeto de estudio**: la interoperabilidad semántica entre Sistemas de Información en Salud. Enfocándose en el **campo de acción**: la persistencia de información en los registros de datos clínicos. Para dar solución al problema planteado se toma como **objetivo general**: aumentar las posibilidades de interoperabilidad semántica de las aplicaciones del Centro de Informática Médica.

Para lograr el objetivo propuesto se dará cumplimiento a las siguientes tareas de la investigación:

1. Levantamiento de información relacionada con el uso de codificadores internacionales en el desarrollo de las aplicaciones del CESIM.
2. Realización de estudio del estado del arte sobre servicios de terminologías médicas y tendencias tecnológicas actuales.
3. Elaboración de la propuesta de solución a los problemas de interoperabilidad semántica actuales en los productos del CESIM.
4. Validación de la propuesta de solución.

Entre los beneficios que se esperan con el desarrollo de la presente investigación se encuentran:

1. Mejorar el rendimiento de las aplicaciones a partir de uso de un servicio centralizado.

2. Mejorar la interoperabilidad semántica entre las aplicaciones del CESIM basada en la homogenización de los codificadores internacionales usados.
3. Proporcionar una herramienta de consulta de términos médicos.
4. Permitir la gestión de los codificadores.

La estructura del presente documento se compone de 3 capítulos en los que se recoge la información obtenida desde el punto de vista investigativo y la propuesta del sistema. Dichos capítulos son:

Capítulo 1. Fundamentos teóricos-metodológicos relacionados con la investigación: en este capítulo se abordan los diferentes aspectos teóricos, que son la base para el desarrollo de la solución, se realiza un estudio del arte relacionado con el trabajo; así como una breve explicación de las herramientas y tecnologías que se utilizan para la realización de la propuesta de solución.

Capítulo 2. Diseño de la solución: en este capítulo se describe en términos de dominios la situación actual describiendo los conceptos del marco y las relaciones que existen entre ellos. Se presenta la propuesta de solución, especificando los requerimientos funcionales, no funcionales y el modelo de casos de usos de la solución propuesta.

Capítulo 3. Validación de la propuesta de solución: en este capítulo se refleja la validación de la solución propuesta a partir de la implementación de la persistencia de los datos de evolución de un paciente utilizando como codificador el servicio propuesto como parte de la solución. Para ello se asumió la línea base del Sistema de Gestión Hospitalaria modificando las clases necesarias para la generación de documentos CDA, y se desarrolló por completo una interfaz de comunicación intermedia entre las aplicaciones y el servicio terminológico. Para la validación se utilizaron los mecanismos propuestos por HL7 para la comprobación estructural de dichos documentos.

Capítulo 1: Fundamentos teóricos-metodológicos relacionados con la investigación

El objetivo de este capítulo es abordar los diferentes aspectos teóricos y metodológicos, que son la base conceptual para el desarrollo del trabajo. En el mismo se describen los conceptos principales para comprender el basamento de la investigación, se realizará un análisis del Sistema de Información en Salud y los principales servidores terminológicos más utilizados a nivel internacional. Además, se describen las tecnologías requeridas para su desarrollo, así como los lenguajes y técnicas a emplear.

1.1 Análisis del Sistema de Información en Salud

Un sistema de información es aquel que está constituido por un conjunto de instrucciones organizadas, sistematizadas y lógicas que se relacionan entre sí por medio de un lenguaje informático. Persigue el fin de obtener información, analizarla, relacionarla y generar nueva información para satisfacer las necesidades de las áreas administrativas y operativas de una organización en general (7). Provee comunicación entre los miembros del equipo de salud y da soporte organizacional a la gestión de la información para realizar operaciones, planeamiento, atención del paciente y el registro de sus actos. Están compuestos por múltiples componentes, no solo están conformados por piezas de software, sino también por recursos humanos y tecnológicos y deben ser considerados como subsistemas en el sistema de información en organizaciones de salud. Los componentes de los Sistemas de Información en Salud son:

- **Componente Computacional:** representa todo el soporte tecnológico por medio del cual el sistema de información puede llevar a cabo sus funciones.
- **Componente de Sistemas Administrativos:** garantiza el soporte administrativo de los procesos asistenciales y la facturación de los actos médicos.
- **Componente de Sistemas Departamentales Clínicos:** representa los sistemas para registrar y dar soporte a la realización de exámenes complementarios utilizados en las áreas clínicas auxiliares.
- **Componente de Interoperabilidad:** garantiza la comunicación y la integración entre los sistemas de información mediante estándares.
- **Componente de Servicios Terminológicos:** encargado de dar servicios terminológicos que permiten lograr el equilibrio adecuado entre la libertad del ingreso de textos narrativos y los beneficios del ingreso estructurado de datos en los sistemas de información.

Capítulo 1: Fundamentos teóricos-metodológicos relacionados con la investigación

- **Componente de Registro Clínico Electrónico:** representa el sistema que utilizan los miembros del equipo de salud para registrar su quehacer asistencial. Es el lugar primario para la carga de toda la información clínica.
- **Componente de Registro Personal de Salud:** plantea la generación de un “Portal Personal de Salud” donde la información de todos los componentes se muestra desde la perspectiva y necesidades del paciente, con el objetivo de brindar herramientas de potenciación a los mismos.
- **Componente de Seguridad:** representa el proceso de firma electrónica/digital de los documentos clínicos y se encarga también de administrar los permisos necesarios para el acceso a la información clínica.
- **Componente de Soporte para la Toma de Decisiones:** constituido por los sistemas diseñados para ayudar al profesional de la salud en la toma de decisiones clínicas.
- **Componente de Agregación de la Información:** se encarga de administrar la información de un grupo de personas enroladas según diferentes criterios (patologías crónicas, neoplasias, enfermedades infectocontagiosas, etc.) para posteriormente generar múltiples intervenciones.
- **Componente Organizacional:** está dedicado a los aspectos socio-organizacionales de los sistemas de información en salud. Este componente engloba al resto. (2)

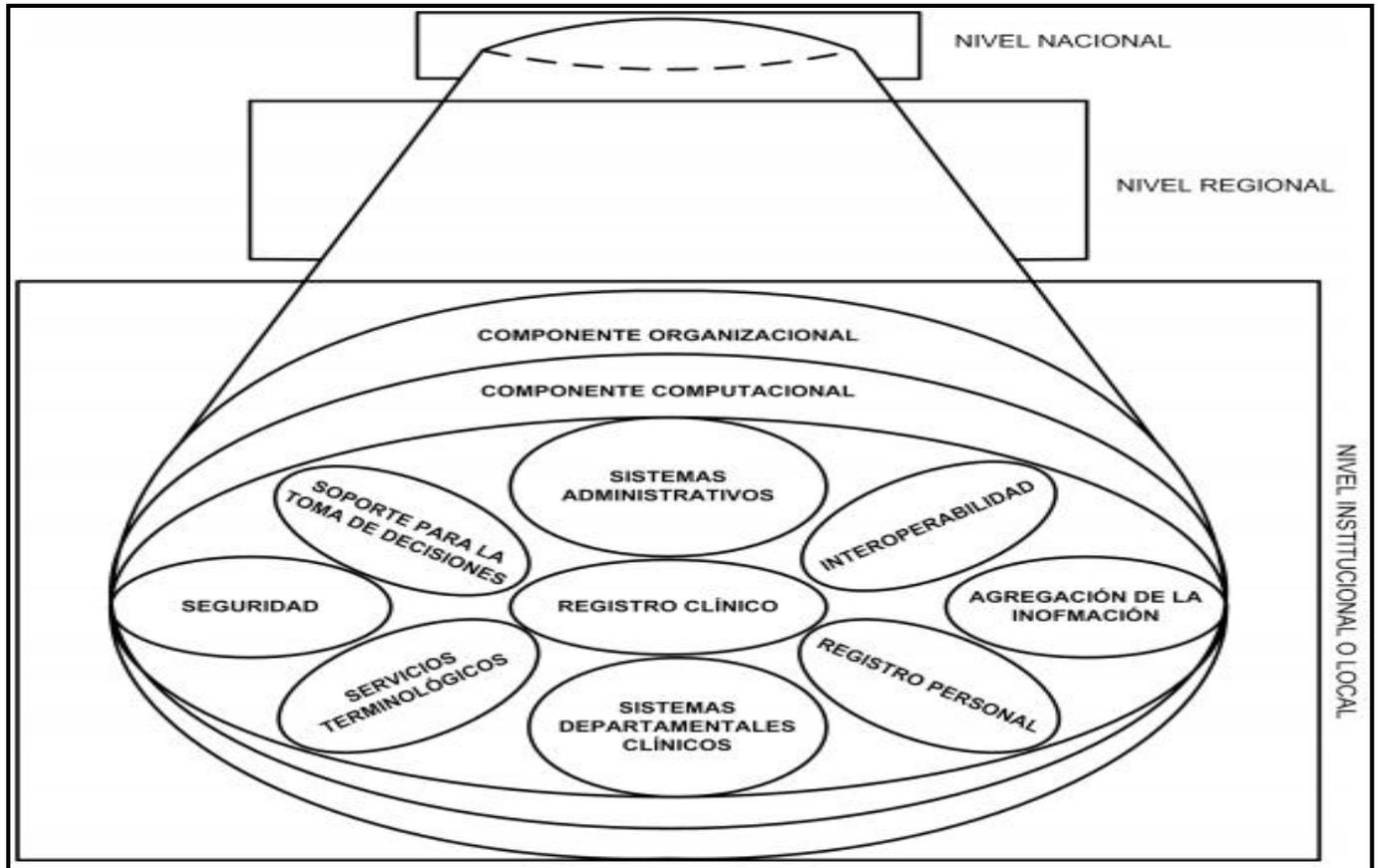


Figura 1.1. Componentes de un sistema de información en salud.

Los componentes de interoperabilidad y de servicios terminológicos son los que poseen mayor relación con el objeto de estudio de esta investigación. Por esta razón a continuación se aborda con más profundidad acerca de ellos.

Componente de Interoperabilidad

Se entiende por interoperabilidad la capacidad de dos o más sistemas de intercambiar y utilizar información entre ellos, de esta definición se desprenden dos tipos de interoperabilidades: la sintáctica (operativa o funcional) y la semántica.

El componente de **interoperabilidad sintáctica** posibilita que los diferentes sistemas se comuniquen entre sí, pero no asegura que los mismos puedan utilizar la información que intercambian. El **componente semántico** es el encargado de almacenar, administrar, integrar y unificar los diccionarios o vocabularios comunes que utilizan los sistemas que deben ser integrados (2).

Componente de Servicios Terminológicos

Los miembros del equipo de salud registran su actividad asistencial de forma narrativa, la cual mantiene gran cantidad de información contextual necesaria para la comunicación con sus pares, pero la misma puede ser ambigua. Varios conceptos pueden estar representados por un mismo término (por ejemplo, el término *polisemia*, puede ser tanto ósea como mamaria) o un mismo concepto representado por varios términos (por ejemplo: *sinonimia* también se conoce como fiebre, temperatura elevada, hipertermia, entre otros).

El componente de servicios terminológicos se encarga de representar las terminologías en un formato consistente, donde se logre un adecuado equilibrio entre la libertad de los textos narrativos y los beneficios del ingreso estructurado de datos (2).

Actualmente la descentralización de la atención médica en redes asistenciales ha generado la necesidad de conectar múltiples sistemas informáticos. Conseguir un historial clínico electrónico que incluya todos los datos existentes sobre un paciente con independencia del sistema que lo haya generado o dónde se encuentre almacenada y mantenida la información, requiere la interoperabilidad semántica de los SIS, por lo que el enlace terminológico entre los datos y las terminologías clínicas es un requisito previo para conseguirla. El intercambio de datos debe hacerse de forma significativa, evitando cualquier posibilidad de error o mala interpretación, asegurando que la información conserve al ser recibida, el significado asignado cuando fue generada, por lo que el uso de estándares se hace necesario.

1.2 Estándares de terminologías médicas

Hoy en día es imprescindible el uso de estándares médicos para lograr integrar, consolidar y coordinar la información de los pacientes por medio de la creación de redes que brinden cuidado médico. Estos constituyen las reglas que gobiernan el modo en que la información del paciente es intercambiada de forma electrónica (8). Son la base de la interoperabilidad, sin ellos no es posible construir sistemas interoperables. Existen seis clasificaciones de estándares en el sector sanitario.

Capítulo 1: Fundamentos teóricos-metodológicos relacionados con la investigación

Los estándares de terminologías médicas proveen códigos específicos para los conceptos médicos, ejemplo: enfermedades, traumatismos, resultados de laboratorios, entre otros. El desarrollo actual del CESIM implica la inclusión de varios de estos codificadores como son:

La **Clasificación Internacional de Enfermedades (CIE)** Emitido en 1893 por la Organización Mundial de Salud (OMS), la cual está diseñada para promover la comparabilidad internacional en la recopilación, procesamiento, clasificación y la presentación de las estadísticas de mortalidad. El CIE en su novena edición (CIE 9), clasifica las enfermedades, afecciones y causas externas de enfermedades y traumatismos, con el objetivo de recopilar información sanitaria útil relacionada con defunciones, enfermedades y traumatismos (mortalidad y morbilidad). Por su parte, en la versión 10 (CIE 10) se determinan la clasificación y codificación de las enfermedades y una amplia variedad de signos, síntomas, hallazgos anormales, denuncias, circunstancias sociales y causas externas de daños y/o enfermedad (9).

Por su parte la **Clasificación Internacional de Atención al Paciente (CIAP)** (10) es un estándar creado por la Sociedad Mundial de Médicos de Familia (WONCA - por sus siglas en inglés) exclusivamente para ser utilizada en atención primaria. Permite codificar las actividades y elementos del episodio de atención al paciente. Está compuesta por alrededor de 700 códigos, con una granularidad (detalle) muy apropiada para su uso en el ámbito ambulatorio por médicos generalistas, ya que permite la recolección y análisis de tres importantes componentes de la consulta médico-paciente: la razón de la consulta, el problema atendido y el proceso de atención. Es utilizada tanto para actividades clínicas, como docentes y de investigación.

Para la identificación de observaciones de laboratorio médico, se crea en 1994 los **Códigos y Nombres de Identificadores Lógicos de Observación** (LOINC – por sus siglas en inglés). Este estándar incluye no solo la codificación de las prácticas en laboratorio, sino también del diagnóstico en enfermería, de las intervenciones de enfermería, de la clasificación de altas, o de la información del cuidado del paciente (11). En el caso de la **Asociación Norte Americana de Diagnósticos de Enfermería** (NANDA- por sus siglas en inglés), es un estándar de codificación para el diagnóstico de enfermería, tiene como objetivo desarrollar, mejorar y promover la terminología que refleje con precisión el juicio clínico emitido en enfermería para mejorar todos los aspectos de la práctica de la misma. Además permite contribuir a la seguridad del paciente integrando la terminología basada en la evidencia en los procesos clínicos y en la toma de decisiones, mejorando la atención a la salud de las personas en el ámbito de la enfermería (12).

Otros estándares incluidos son la Clasificación Anatómica, Terapéutica y Química (ATC – por sus siglas en inglés) y el **Colegio Americano de Radiología** (ACR – por sus siglas en inglés). El ATC es un sistema de codificación de fármacos y medicamentos, según su efecto farmacológico, sus indicaciones terapéuticas y su estructura química. Almacena el sistema u órgano en el que actúa el fármaco, el efecto farmacológico, las indicaciones terapéuticas y la estructura química de dicho fármaco (13). Por su parte ACR definen los principios del suministro de servicios de exámenes por imágenes y terapéuticos de alta calidad.

Utilizar estándares de terminología trae grandes beneficios en las diversas áreas sanitarias, por lo que las soluciones más apropiadas a utilizar son aquellas que promueven el uso de los mismos. El uso de servicios terminológicos para la codificación permite a los SIS crecer en complejidad sin perder la información anterior y adaptando los desarrollos previos sin tener que diseñar nuevos sistemas.

1.3 Especificaciones de servidores de términos médicos

Para el desarrollo de la solución se realizó una investigación de los servidores de terminologías médicas más utilizados a nivel internacional, para analizar las principales características y funcionalidades con el objetivo de seleccionar el sistema que será utilizado para dar solución al problema científico. Durante el análisis de los sistemas encontrados se tuvieron en cuenta aspectos como la coherencia con el problema científico, soporte, si es modificable o no, si es multiplataforma, entre otros.

Uno de estas especificaciones es la que se describe como **Servicios de Consultas de Léxicas** (LQS - por sus siglas en inglés), también conocido como Servicios de Consultas Terminológicas (TQS - por sus siglas en inglés), servidor de terminología que propone 13 módulos y especifica un conjunto de métodos comunes y de solo lectura para acceder a contenido de terminologías médicas. LQS fue concebido para los siguientes escenarios de uso:

- **Adquisición de información:** usar los servicios de terminologías para ayudar en el proceso de introducción de datos codificados.
- **Visualización de información:** utilizar los servicios de terminologías para traducir información codificada a formas de lenguaje humano o legible por la máquina.
- **Mediación:** utilizar los servicios de terminología para transformar mensajes o registros de información de una forma o representación a otra.

- **Indexación e inferencia:** utilizar los servicios de terminología para obtener información acerca de asociaciones las cuales pueden o no pertenecer a varios elementos de datos y ayudar en la localización de varios conjuntos de registros de datos, que pueden contener información relevante para el tema específico o entidad.
- **Navegación:** utilizar los servicios de terminología para determinar la estructura y significado de un sistema de terminología.

Entre las limitaciones de este servidor de terminologías se encuentran:

- El soporte del proveedor es mínimo.
- Su especificación es considerada demasiado densa y compleja.
- Solo permite acceso a las terminologías y no su modificación. Ha sido diseñado para la navegación y no para la inserción de los conceptos.
- No se conoce implementación de este servidor (14).

Otros servicios analizados fueron los que se describen como: **Servicios de Terminologías Comunes** (CTS- por sus siglas en inglés) que definen una interfaz de programación de aplicaciones (API - por sus siglas en inglés) que puede ser utilizada por un software basado en la versión 3 de *Health Level Seven* (HL7- por sus siglas en inglés) cuando accede a contenido terminológico. Fue desarrollado como una alternativa a una estructura de datos común. En lugar de especificar a lo que una terminología externa debe parecerse, CTS identifica las características funcionales comunes que este tipo de vocabulario debe ser capaz de proporcionar. Por ejemplo, un servicio compatible con HL7 tendrá que ser capaz de determinar si un código de concepto dado es válido dentro de un recurso particular. Entre los beneficios que tiene el uso de CTS se encuentran:

- El software que use CTS se puede implementar sin la necesidad de comprender la base de datos de la terminología de cada proveedor.
- Oculta gran parte de la complejidad inherente a los sistemas modernos de la terminología.
- Los requerimientos funcionales de los servicios de terminologías se especifican claramente.
- La implementación puede estar basada en bases de datos y software existentes.

Capítulo 1: Fundamentos teóricos-metodológicos relacionados con la investigación

- Es punto de entrada común a los contenidos de la terminología.
- Es utilizado para traducir conceptos de código de un sistema a otro.

CTS no está diseñado para hacer lo siguiente:

- No se pretende que CTS sea un servidor de terminología completo.
- Su alcance se limita a la funcionalidad necesaria para diseñar, implementar y desplegar un paquete de software compatible con la versión 3 de HL7.
- Tampoco se pretende que CTS sea un lenguaje de propósito general de consulta. Su objetivo es especificar sólo los servicios específicos necesarios en la implementación de HL7.
- CTS no especifica cómo el servicio debe implementarse. Una implementación de CTS usará la arquitectura subyacente que sea la más apropiada para las circunstancias de ejecución dadas (15).

Otras soluciones encontradas pertenecen al grupo de **Servidores de Terminologías Distribuida** (DTS—por sus siglas en inglés, plataforma de servidor código abierto para la gestión de terminologías de salud. Actualmente es utilizada en varias organizaciones líderes de salud, pues proporciona soporte para los estándares de datos tanto nacionales como internacionales, así como los vocabularios locales, bases de datos necesarios para la información de salud comparable y compatible. Las aplicaciones típicas para DTS incluyen la entrada de datos clínicos, resultados de examen, gestión de código de conjunto, la creación de guías, apoyo a la decisión y la recuperación de información. Las principales características de DTS son:

- **Alto rendimiento:** accesos simultáneos a múltiples terminologías interconectadas.
- **Integral:** base de conocimientos de terminología extensa con un modelo de objetos unificado y coherente.
- **La normalización de datos:** coincidencia de entrada de texto para los términos y conceptos estandarizados a través de análisis de orden de las palabras, palabra derivada, la corrección ortográfica.
- **Código traducción:** mapeo de datos clínicos de los sistemas de codificación estándar, como CIE-9.

- **Navegación semántica:** *browsing*¹ de un rico conjunto de relaciones jerárquicas y no jerárquicas entre conceptos para mejorar la calidad en la entrada de datos y recuperación de la información.
- **Clasificación semántica:** la creación, la gestión, y la comparación de las extensiones de síntesis que sean consistentes con los modelos semánticos formales.
- **Subconfiguración:** creación de subconjuntos individuales de terminologías que usan técnicas de lógica de Boole avanzadas.
- **Localización:** la adición de conceptos locales, sinónimos, códigos, propiedades y asociaciones inter - concepto para conectar el contenido local a terminologías estándar (16).

Dentro de estas especificaciones existen varias soluciones relevantes como son:

El **Cazador de Términos Clínicos** (C.T Hunter – por sus siglas en inglés) (R) desarrollado en la Escuela Técnica Superior de Ingeniería informática de Valencia. El mismo es un servidor destinado solamente a trabajar con el estándar de terminología, SNOMED-CT, por lo que sus métodos son específicos para realizar consultas sobre este vocabulario. Cuenta con una API que permite el uso de sus funcionalidades, dividida en dos partes, una para los menús y realizar las búsquedas y la otra para mostrar los resultados. Su aspecto está diseñado para parecerse a un navegador web, con la posibilidad de navegar por las pestañas, cerrar aquellas búsquedas que ya no sean útiles, y acceder fácilmente a un concepto. C.T Hunter tiene como desventajas que las funciones propias que implementa no están disponibles para descarga y son solo de búsquedas, no de edición (17).

LexGrid es un modelo de datos y una implementación de la Clínica Mayo de EE.UU para almacenar de forma estándar vocabularios controlados y ontologías. Define un formato y una representación para vocabularios y representa con exactitud una gran variedad de terminologías y otros recursos léxicos. Este modelo permite caracterizar las partes fundamentales de las terminologías de manera que todas pueden descomponerse y conceptualizarse para ser traducidas al modelo de datos.

La comunidad de *Cancer Biomedical Informatics Grid* (caBIG) refinó esta implementación de la Clínica Mayo para dar lugar a la herramienta **LexBIG**, servidor terminológico diseñado para almacenar y acceder a

Proceso interactivo en el que se visualizan grandes cantidades de información.

terminologías. Su objetivo es lograr que los vocabularios puedan ser accedidos mediante una API bien estructurada, capaz de acceder y distribuir recursos. Posee una interfaz gráfica que permite tanto la carga de terminologías como la búsqueda de conceptos empleando sus propias API. LexBIG presenta como desventaja que al centrarse en funciones de búsqueda única para todos los estándares de terminologías que pretende soportar, carece de funciones para aprovechar las características específicas de cada uno. Además, los pasos a seguir para realizar una búsqueda son complicados y poco intuitivos (18).

Otro servidor relevante es el desarrollado por la empresa informática Apelon dedicada a la estandarización de datos y la interoperabilidad Apelon, desarrolla el servidor terminológico Apelon DTS, solución de código abierto que permite la gestión y despliegue de terminologías normalizadas. Ofrece una API para la navegación, búsqueda de términos médicos. Además, ofrece soporte de software comercial y los servicios de suscripción de contenido de terminología para la plataforma. Entre las ventajas del uso del Apelon DTS como servidor terminológico se encuentran:

- Alto rendimiento.
- Acceso simultáneo a múltiples terminologías.
- Amplia base de conocimientos terminológica que soporta los estándares de codificación internacionales.
- Integrada visualización, navegación y búsqueda a través de los sistemas de código y subconjuntos.
- Mapeo de datos clínicos de los sistemas de codificación estándar (16).

Teniendo en cuenta las funcionalidades y características de los sistemas mencionados anteriormente, se definieron las siguientes variables para seleccionar el sistema más viable para dar solución a la problemática actual de los productos del CESIM. Estas variables tienen varios resultados posibles y a cada resultado le corresponde una puntuación. A continuación una breve descripción de cada una de ellas.

- **Coherencia con el problema científico:** las funcionalidades del sistema dan solución al problema científico. El grado de satisfacción de esta variable recibe una puntuación entre 1 y 3, donde 3 es el valor máximo
- **Es modificable:** en caso de ser necesario el sistema es flexible a algunos cambios. El grado de satisfacción de esta variable recibe una puntuación entre 1 y 3, donde 3 el valor máximo.

- **Soporte:** el proveedor brinda soporte para mantener el sistema y actualizarlo. El grado de satisfacción de esta variable recibe una puntuación entre 1 y 3, donde 3 es el valor máximo.
- **Multiplataforma:** el sistema puede ser utilizado tanto en Sistema operativo Windows como en Linux. El grado de satisfacción de esta variable recibe una puntuación de 2 si es Si y 1 si es No.
- **Nuevas terminologías:** el sistema permite adicionar nuevos sistemas de codificación en su base de conocimientos. El grado de satisfacción de esta variable recibe una puntuación de 2 si es Si y 1 si es No.
- **Viabilidad de las tecnologías:** las tecnologías con las que ha sido desarrollada la solución se encuentran dentro del marco de desarrollo del CESIM. El grado de satisfacción de esta variable recibe una puntuación entre 1 y 3, donde 3 es el valor máximo.

En la siguiente tabla se muestra el resultado de la comparación de cada una de las especificaciones de servidores estudiadas a partir de las variables anteriormente definidas:

Variables	CTS	DTS	TQS/LQS
Coherencia con el problema científico	2	3	2
Es modificable	2	2	1
Soporte	1	3	1
Multiplataforma	2	2	2
Nuevas terminologías	2	2	1
Viabilidad de las tecnologías	3	3	2
	12	15	9

Tabla1. Comparación de los sistemas estudiados.

Luego de la investigación realizada y el posterior análisis de cada especificación analizada, así como soluciones más relevantes, se decide seleccionar el DTS como la especificación de servidores terminológicos que será utilizada en la propuesta de solución, por permitir la integración, la distribución y la utilización de las terminologías ofreciendo servicios terminológicos que simplifican la integración de estándares de datos en estas aplicaciones, y reducen los costes de desarrollo. Además, se selecciona como servidor de terminología médica a utilizar el Apelon DTS.

Una vez analizadas las potencialidades y dificultades de las soluciones más destacadas se hace necesario describir como acceder a las funcionalidades que brinda un ST, para ello se identificó como alternativa más

oportuna seguir la especificación que propone ***Integration Healthcare Enterprise (IHE)*** para interactuar con servicios de este tipo.

1.4 Perfiles de Integración IHE

Los Perfiles de Integración IHE describen una necesidad clínica de integración de sistemas y la solución para llevarla a cabo. Cada Perfil IHE tiene el objetivo de organizar y aprovechar las capacidades de integración que se pueden lograr mediante la aplicación coordinada de las normas de comunicación expuestas en los diferentes estándares. Ellos proporcionan definiciones precisas de cómo las normas pueden aplicarse para satisfacer las necesidades específicas de una situación clínica determinada. Los perfiles de integración definen claramente cómo deben encajar todas las piezas basándose en estándares aceptados globalmente. El dominio de Infraestructura de la Tecnología de la Información (ITI) define 21 perfiles entre los cuales se identificó el perfil Compartir Conjuntos de Valores (SVS – por sus siglas en inglés) (19).

El SVS proporciona un medio para producir o consumir datos clínicos o administrativos (tales como: equipos de diagnóstico por imagen, sistemas de información de laboratorio, o sistemas de registro de salud nacional) de forma tal que se pueda recibir una nomenclatura uniforme. Este perfil describe las transacciones para la recuperación de conjuntos de valores (*Value Sets*) de un repositorio por un consumidor de conjunto de valores (*Value Set Consumer*), estableciendo un dominio de un conjunto coherente y uniforme de las nomenclaturas.

Un Conjunto de Valores es un conjunto único de identificación de las representaciones válidas de un concepto. Un Conjunto de Valores puede ser una simple lista plana de códigos conceptuales extraídas de un único sistema de códigos, o podría ser constituido por expresiones extraídas de los sistemas de códigos múltiples (Un sistema de código es un sistema que consiste en las designaciones y significados, por ejemplo LOINC, SNOMED -CT, la CIE- 10).

Un concepto válido es un concepto que sería lógicamente representativo del valor establecido que pertenece a, por ejemplo para el valor – “Enfermedades del sistema nervioso”, “Parálisis cerebral y otros síndromes paralíticos” sería un concepto válido (tomado de la CIE).

Capítulo 1: Fundamentos teóricos-metodológicos relacionados con la investigación

Para un mejor entendimiento se describen como **Actores involucrados** las diferentes partes que interactúan en cada transacción.

Actor Consumidor de Conjunto de Valores: obtiene un Conjunto de Valores (terminologías médicas) a través de una solicitud realizada al Repositorio de Conjunto de Valores. Este es el actor principal en el proceso de recuperación de Conjunto de Valores.

Actor Repositorio de Conjunto de Valores: es el encargado de almacenar y proporcionar a los consumidores el acceso a las terminologías médicas solicitadas.

La interacción entre estos actores se realiza a través de los mensajes enviados en las transacciones que este perfil define:

Transacción Retrieve Value Set: el Consumidor de Conjunto de Valores recupera un conjunto de valores del Repositorio de Conjunto de Valores.

Transacción Retrieve Multiple Value Sets: el Consumidor de Conjunto de Valores recupera múltiples conjunto de valores del Repositorio de Conjunto de Valores (20).

Herramientas y tecnologías utilizadas para la elaboración de la propuesta de solución

En este epígrafe se expondrán las tecnologías y herramientas que se utilizarán durante el transcurso de la investigación. Las mismas fueron definidas por el CESIM para el desarrollo de sus aplicaciones y se encuentran descritas en el documento Arquitectura Vista de Entorno de Desarrollo Tecnológico.

Como lenguaje de programación se utilizará el Java por ser multiplataforma, orientado a objetos, funciona perfectamente en red. Tiene una gran funcionalidad gracias a sus librerías y genera aplicaciones con pocos errores posibles. Una de las ventajas más importantes del uso de Java es que los programas “ejecutables”, creados por su compilador, son independientes de la arquitectura. Se ejecutan indistintamente en una gran variedad de equipos con diferentes microprocesadores y sistemas operativos.

También será usada la máquina virtual de Java porque es el entorno donde se ejecutan los programas java. Cuenta con características tales como: la portabilidad, eficiencia y seguridad. Cumple con reservar espacio en memoria para los objetos creados, además libera la memoria no utilizada, y acude al sistema huésped

Capítulo 1: Fundamentos teóricos-metodológicos relacionados con la investigación

para ciertas funciones que permiten la ejecución de las instrucciones, y la realización de las demás funcionalidades que brinda (21).

Como Entorno de Desarrollo Integrado (IDE – por sus siglas en inglés), se seleccionó el Eclipse Ganymede por ser una plataforma robusta, donde se pueden realizar la integración con diferentes marcos de trabajos (*framework*) como por ejemplo JBoss Seam 2.1.1.

JBossSeam2.1.1 es un *framework* que ha sido diseñado para simplificar al máximo el desarrollo de aplicaciones, donde se integra la capa de presentación (JSF – por sus siglas en inglés) con la capa de negocios y persistencia (EJB).

Como servidor de aplicaciones se utilizará el Jboss 4.2.2, por ser el servidor de aplicaciones de código abierto más ampliamente desarrollado del mercado. Por ser una plataforma certificada 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* (20).

Como sistema gestor de base de datos (SGBD) se seleccionó el PostgreSQL 8.4 (22), sistema de base de datos relacional perteneciente al ámbito del software libre, que se destaca por tener robustez, escalabilidad y cumplimiento de los estándares SQL. Además, muestra un enfoque orientado a la adición de características (por ejemplo, la autenticación, el control, la reutilización del espacio), y agrega las capacidades definidas en las normas de SQL posteriores.

Entre sus ventajas se encuentra la instalación ilimitada, permitiendo que nadie pueda demandarlo por violar acuerdos de licencia, puesto que no hay costo asociado a la licencia del software. Es multiplataforma. Permite ahorros considerables en costos de operación ya que ha sido diseñado y creado para tener un mantenimiento y ajuste mucho menor que los productos de los proveedores comerciales, conservando todas las características, estabilidad y rendimiento (22).

Se seleccionó como herramienta de ingeniería de Software Asistida por Computadora (CASE – por sus siglas en inglés) el Visual Paradigm 8.0, Esta herramienta utiliza Lenguaje Unificado de Modelado (UML – por sus siglas en inglés) como lenguaje de modelado, con el uso del acercamiento orientado al objeto. Además, genera productos de calidad, soporta aplicaciones web y es fácil de instalar y actualizar. En sentido general, reduce significativamente los esfuerzos en todas las etapas del ciclo de vida de desarrollo de software (23).

Capítulo 1: Fundamentos teóricos-metodológicos relacionados con la investigación

La metodología de desarrollo seleccionada es el Proceso Unificado de Desarrollo (*RUP*- por sus siglas en inglés), por ser una metodología adaptable, es decir, se puede modificar para adaptarla al sistema concreto que se va a desarrollar en cada momento. Es una metodología que sigue un proceso iterativo e incremental, está centrada en la arquitectura y guiada por casos de uso. Incluye artefactos, que son los productos tangibles obtenidos mediante su aplicación y trabajadores, personas que juegan un rol determinado para la obtención de estos artefactos (24).

En este capítulo se valoraron los principales conceptos relacionados con el campo de la investigación, lo que permitió una mejor comprensión de los mismos. Se realizó un análisis crítico y valorativo de los sistemas informáticos de servicios terminológicos existentes a nivel internacional, puntualizando varios conceptos fundamentales que permiten un mejor entendimiento del problema científico. Además se presenta la justificación para la elección de los lenguajes de programación, las tecnologías y las herramientas a utilizar, evaluándose cada una de ellas por las características y ventajas que aportan facilidad para el desarrollo de la aplicación.

Capítulo 2: Diseño de la solución

En este capítulo se explican los conceptos fundamentales abordados en el desarrollo del Modelo de Dominio facilitando la comprensión de estos. Se especifican los requisitos funcionales que deben ser automatizados y se describen las características del sistema a través de los requisitos no funcionales. Además, se plantea la propuesta de solución.

2.1 Modelo de Dominio

El Modelo de Dominio es una representación visual de clases conceptuales o de objetos reales en un dominio de interés, donde se representan los conceptos más importantes y significativos en el desarrollo de un sistema. Su principal objetivo es ayudar a comprender los conceptos que utilizan los usuarios, las definiciones con las que trabajan y con las que deberá trabajar la aplicación (25).

Definición de las clases del Modelo de Dominio

Para brindar una mayor comprensión del diagrama de Modelo de Dominio, a continuación se realiza una breve descripción de los conceptos encontrados en el problema

Repositorio: servidor terminológico donde se almacenan los términos médicos.

Términos: términos médicos que se almacenan en el repositorio y están compuestos por un código y la descripción del término

Consumidor: sistemas del CESIM que solicitan los servicios del repositorio.

Documentos Clínicos: historia clínica del paciente que contiene términos médicos.

Diagrama del Modelo de Dominio

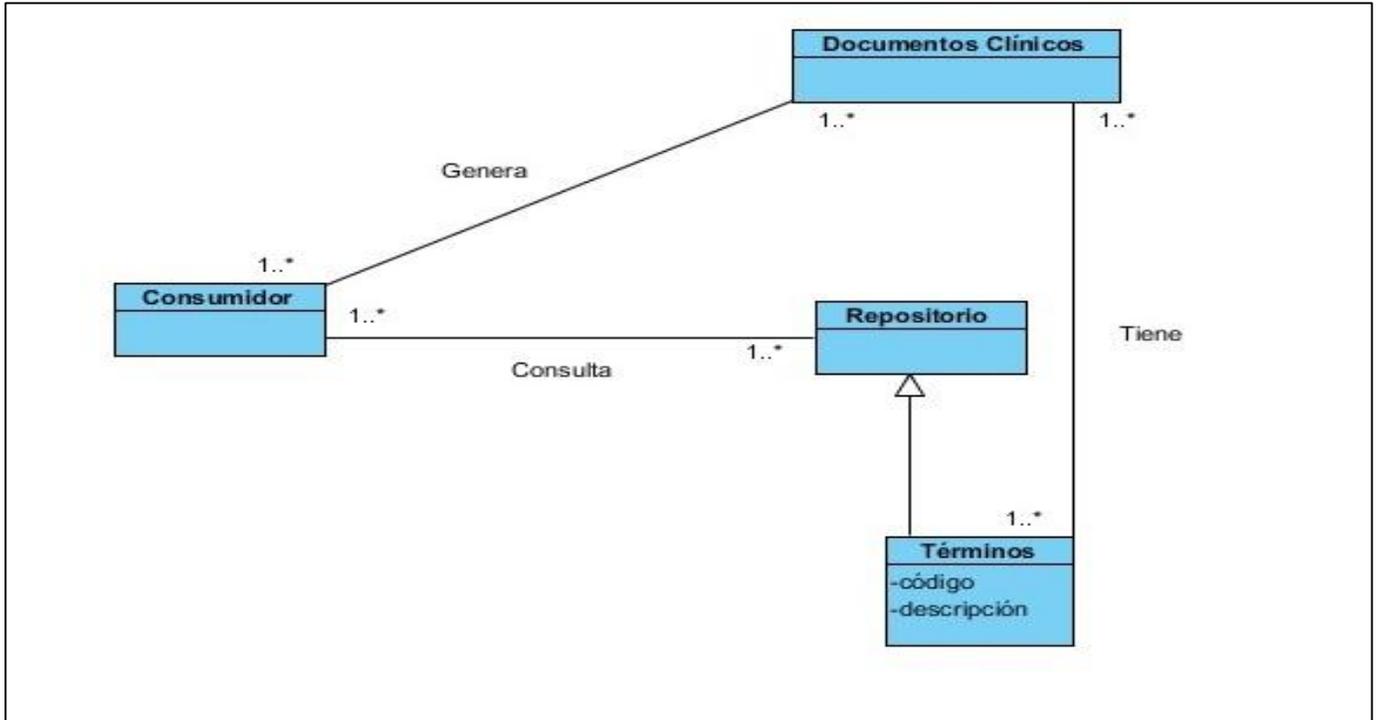


Figura 2.1 Modelo de Dominio

2.2 Propuesta del sistema

Se propone la utilización del Servidor de Terminologías Distribuidas (Apelon DTS) con el objetivo de elevar la interoperabilidad semántica de los sistemas existentes en el CESIM. El servidor almacenará todas las terminologías utilizadas por el centro (CIE, NANDA, LOINC, entre otras), Para realizar la persistencia de la información clínica de los pacientes utilizando el Apelon DTS se propone el desarrollo de una **API de comunicación** que se encargará de conectarse al servidor de terminologías para lograr la comunicación de los sistemas del CESIM con el mismo. A través de la API se realizarán consultas y operaciones sobre la distribución terminológica. Esto permitirá a los médicos realizar búsquedas de conceptos, seleccionar de una lista los conceptos médicos necesarios para emitir un diagnóstico y eliminar conceptos de una selección. Para lograr la integración de la solución con las aplicaciones de CESIM se proponen los siguientes escenarios:

Integración de forma directa: las aplicaciones del CESIM realizarán las solicitudes al servidor de terminologías.

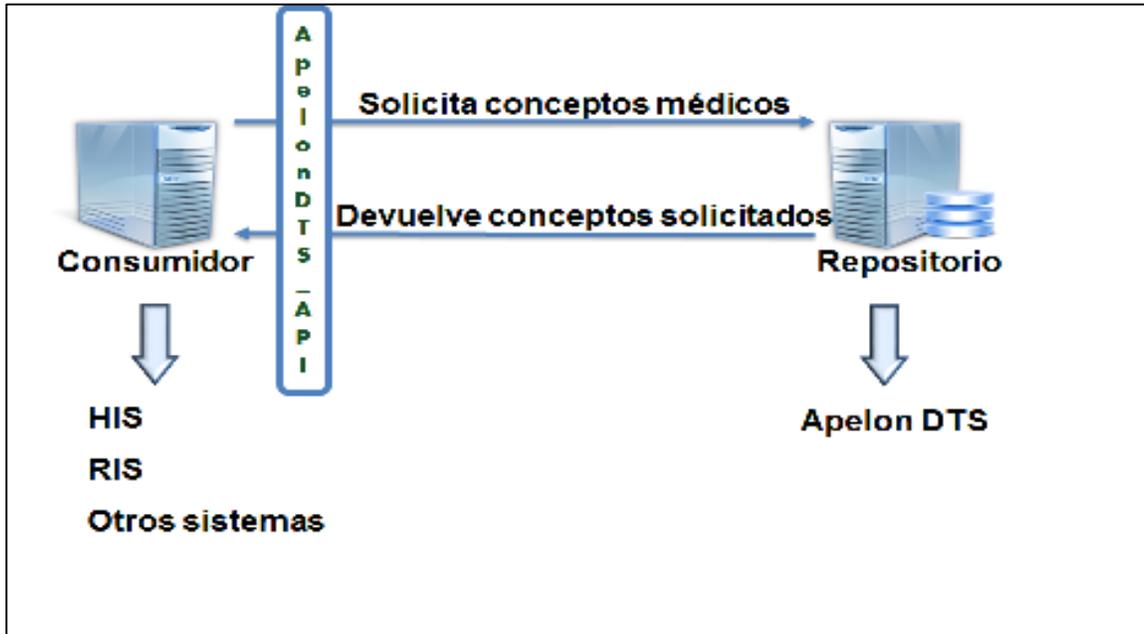


Figura 2.2 Escenario 1 de integración.

A través de una plataforma de integración: la comunicación entre los distintos actores se realiza a través del canal de comunicaciones WSO2 ESB. El servidor de terminologías estará conectado al Canal de Servicios Empresariales del CESIM, el mismo funciona como un intermediario entre el servidor de terminologías y las aplicaciones del Centro. Esta capa intermedia administrará las peticiones realizadas desde las aplicaciones hacia el Apelon y también las respuestas del servidor de terminologías a estos sistemas.

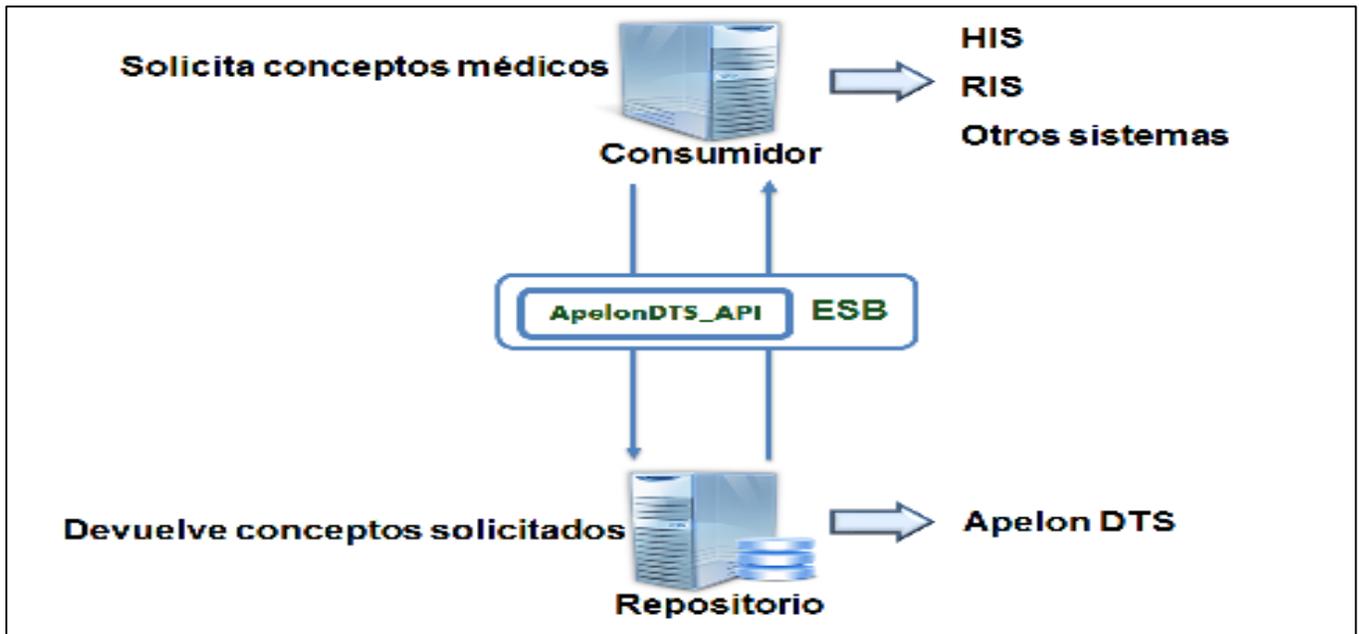


Figura 2.2 Escenario 2 de integración.

2.3 Especificación de los requisitos de la solución.

Los requerimientos de un sistema son la descripción de los servicios proporcionados por el sistema y sus restricciones operativas. Estos requerimientos reflejan las necesidades de los clientes de un sistema que ayude a resolver algún problema como el control de un dispositivo, hacer un pedido o encontrar información. Los requerimientos de software tienen varias clasificaciones, las más comunes son los requisitos funcionales (RF) y los no funcionales (RNF) (26).

Requisitos Funcionales

Los RF de un sistema describen lo que el sistema debe hacer. Son declaraciones de los servicios que debe proporcionar el sistema, de la manera en que este debe reaccionar a entradas particulares y de cómo se debe comportar en situaciones particulares. A continuación se listan los RF del sistema propuesto.

RN1 Realizar sincronización: sincroniza el servidor con las aplicaciones del CESIM.

RF2 Listar Namespaces: se listan los estándares de terminologías disponibles en el Apelon DTS.

RF3 Realizar conexión: realiza la conexión con el servidor.

RF4 Buscar concepto: se buscan conceptos que se encuentran en los estándares terminológicos disponibles en el Apelon DTS.

RF4 Listar conceptos: se listan todos los conceptos que se encuentran en los estándares terminológicos disponibles en el Apelon DTS.

RF5 Generar CDA: Genera los CDA a nivel 3 codificando los mismos con las terminologías almacenadas en el servidor.

Requisitos No Funcionales (RNF)

Son restricciones de los servicios o funciones ofrecidos por el sistema. Incluyen restricciones de tiempo, sobre el proceso de desarrollo y estándares. A continuación se describen los RNF definidos para la solución a realizar.

Usabilidad

RNF1 Debe garantizar el acceso fácil, garantizando que pueda ser utilizado por usuarios con pocos conocimientos informáticos.

RNF2 Solo podrá utilizarse por los usuarios definidos para trabajar en el sistema.

Fiabilidad

RNF3 Estará disponible todo el tiempo permitiendo el trabajo de los sistemas cada vez que lo requieran.

RNF4 Debe ser estable, fiable y la velocidad de respuestas será rápida durante la utilización del mismo.

Eficiencia

RNF5 La eficiencia dependerá de la rapidez de las consultas que se hagan a la base de datos, y el buen aprovechamiento de los recursos disponibles.

Soporte

RNF6 Se podrá trabajar con él desde cualquier sistema operativo.

RNF7 Será compatible con los sistemas desarrollados en el Centro de Informática Médica.

Restricciones de diseño

RNF8 Utilizar las pautas del diseño creadas por el CESIM para la modelación.

RNF9 Se utilizarán patrones para la implementación descritos en el documento Arquitectura de Software del CESIM.

RNF10 Para el diseño de la solución sistema se utilizarán la metodología RUP, usando el UML como lenguaje de modelado, para modelar la herramienta Visual Paradigm.

Interfaz

RNF11 Deben ser visibles las opciones en la interfaz principal.

RNF12 Tendrá colores agradables a la vista del usuario, fácil de manejar y una buena organización en el contenido.

RNF13 Los errores deben ser visibles para el usuario, al igual que envíe mensajes o avisos entendibles.

Interfaz Software

RNF14 Se requiere de un navegador para levantar el sistema.

2.3 Modelo de Casos de Uso

Los casos de uso del sistema permiten que los desarrolladores y a los clientes llegar a un acuerdo sobre las condiciones y posibilidades que debe cumplir una solución informática. Se utilizan para modelar cómo funciona la misma y proporcionan la entrada fundamental para el análisis, el diseño y la implementación en el desarrollo de la solución (27).

Actor	Descripción
Sistemas del CESIM	Es el sistema que realiza la solicitud al servidor para obtener las terminologías que le sea necesaria conocer.

Tabla 2: Actor del sistema

Diagrama de Casos de Uso.

Los diagramas de casos de uso describen las relaciones y las dependencias entre un grupo de casos de uso y los actores participantes en el proceso. Sirven para facilitar la comunicación con los futuros usuarios

de un sistema, y con el cliente, y resultan especialmente útiles para determinar las características necesarias que tendrá la solución. En otras palabras, los diagramas de casos de uso describen qué es lo que debe hacer la solución.

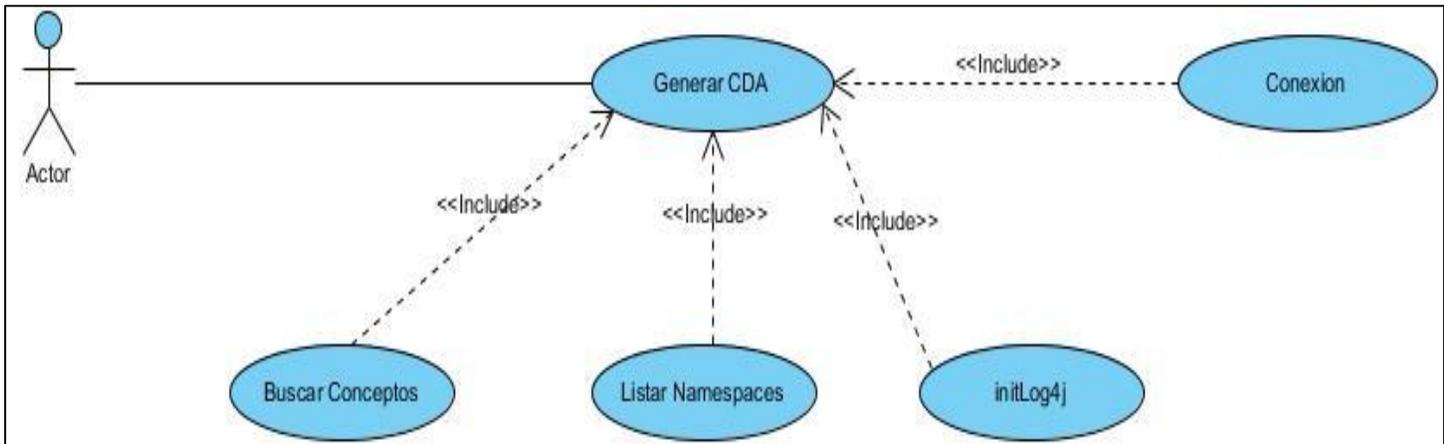


Figura 2.2 Diagrama CU: Generar CDA.

Descripción de los Casos de Uso

CU-1	Realizar sincronización
Actor	Sistemas Externos
Descripción	Establece la sincronización de la consola del servidor con el cliente con el objetivo de llevar todas las trazas de las operaciones realizadas sobre el servidor.
Referencia	RF 1

Tabla 2: Descripción del caso de uso initLog4j.

CU-2	Listar Namespaces
Actor	Sistemas Externos
Descripción	Devuelve todas las terminologías almacenadas en el servidor.
Referencia	RF 2

Tabla 3: Descripción del caso de uso Listar Namespaces.

CU-3	Conexión
Actor	Sistemas Externos
Descripción	Establece la conexión con el servidor y devuelve una instancia de conexión.
Referencia	RF 3

Tabla 4: Descripción del caso de uso Conexión.

CU-4	Buscar Conceptos
Actor	Sistemas Externos
Descripción	Realiza una búsqueda sobre las terminologías almacenadas en el servidor haciendo selección previamente de la terminología donde se realizará la búsqueda. Devuelve una lista de conceptos asociados con la búsqueda
Referencia	RF 4

Tabla 5: Descripción del caso de uso Buscar Conceptos.

CU-5	Generar CDA
Actor	Sistemas Externos
Descripción	Genera los CDA a nivel 3 codificando los mismos con las terminologías almacenadas en el servidor y haciendo uso de los casos de uso descritos anteriormente
Referencia	RF 5

Tabla 6: Descripción del caso de uso Buscar Conceptos.

2.4 Diagrama de Interacción.

Los diagramas de interacción se utilizan para representar los aspectos dinámicos de un sistema. (36). Pueden servir para visualizar, especificar, construir y documentar los aspectos dinámicos de una sociedad particular de objetos, o pueden ser usados para modelar un flujo particular de control de un caso de uso. Los diagramas de interacción están conformados por los diagramas de secuencia y los diagramas de colaboración.

Los diagramas de colaboración pueden ser utilizados para ilustrar la ejecución de una operación, una ejecución de un caso de uso o un escenario de interacción dentro del sistema. Las relaciones se dibujan con líneas y se puede agregar una etiqueta para un mensaje y un número que define la secuencia de estas. (28)

Diagrama de colaboración

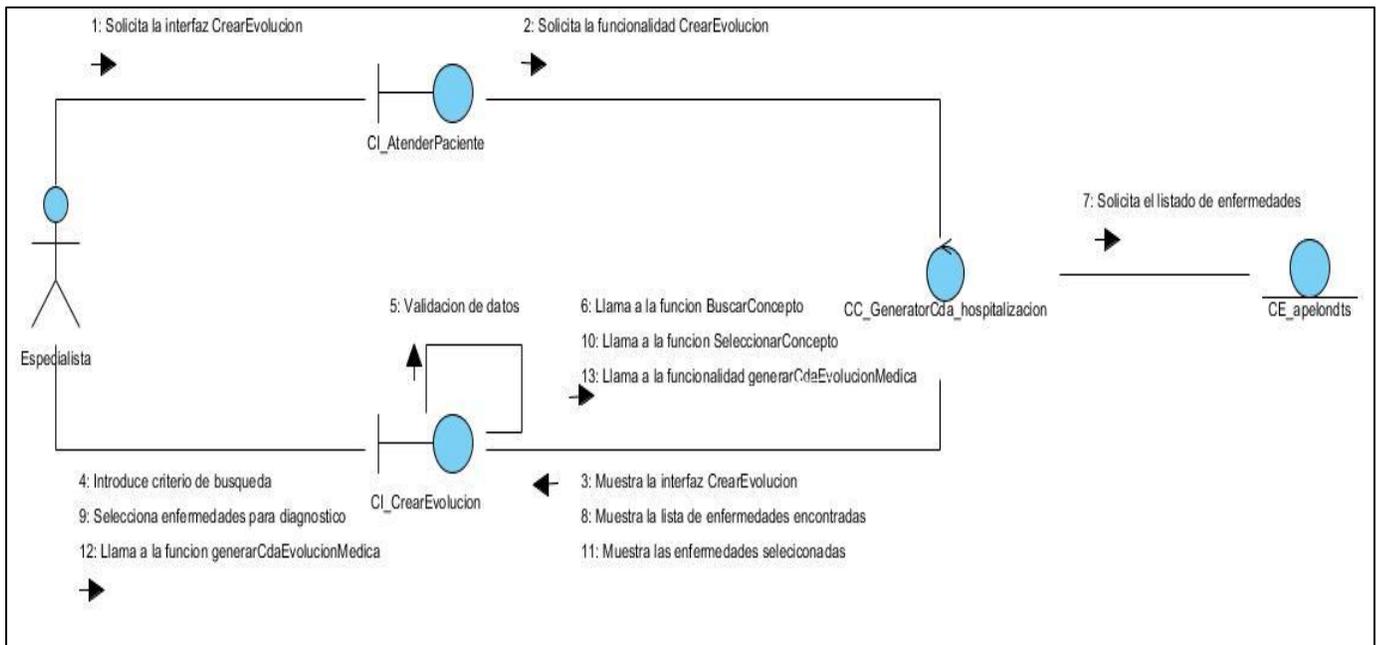


Figura 2.3 Diagrama de colaboración.

En el presente capítulo se abordaron los requisitos funcionales a través de los cuales se definieron las funcionalidades a implementar. Se identificó al actor que interviene, además del diagrama de casos de uso de la solución, logrando una representación detallada de cada proceso. Teniendo en cuenta estos aspectos se posibilita un mejor entendimiento de la solución propuesta.

Capítulo 3: Validación de la propuesta de solución

Después de realizar el diseño de la solución propuesta, se tienen creadas todas las condiciones para comenzar la implementación de la misma. En este capítulo se exponen los componentes de la solución, así como la forma en la que interactúan para posibilitar su correcto funcionamiento. Se brinda una explicación como se realiza el tratamiento de errores, la seguridad y los estándares y estilos utilizados. Además se dejarán evidenciados los resultados de la validación de la propuesta de solución y la pertinencia de su uso.

3.1 Implementación de la API de comunicación

Una API es un conjunto de funciones que permite al programador acceder a servicios de una aplicación a través del uso de un lenguaje de programación. Ofrece al programador un cierto nivel de abstracción que enmascara la complejidad de acceso a un sistema o aplicación, proponiéndole un conjunto de funciones de las cuales sólo se conocen los parámetros y los valores devueltos.

La implementación comienza una vez que se tiene terminado o casi terminado la parte del diseño, por lo que, cada parte del modelo del diseño se implementa en componentes. El diagrama de despliegue que se representa en este epígrafe, es un artefacto que se genera en este flujo de trabajo, conformando el modelo de implementación.

Diagrama de Despliegue

Un Diagrama de Despliegue modela la arquitectura en tiempo de ejecución de un sistema. Esto muestra la configuración de los elementos de hardware (nodos) y muestra cómo los elementos y artefactos del software se trazan en esos nodos. (29) El diagrama representa un grafo, donde sus nodos se unen por conexiones de comunicación.

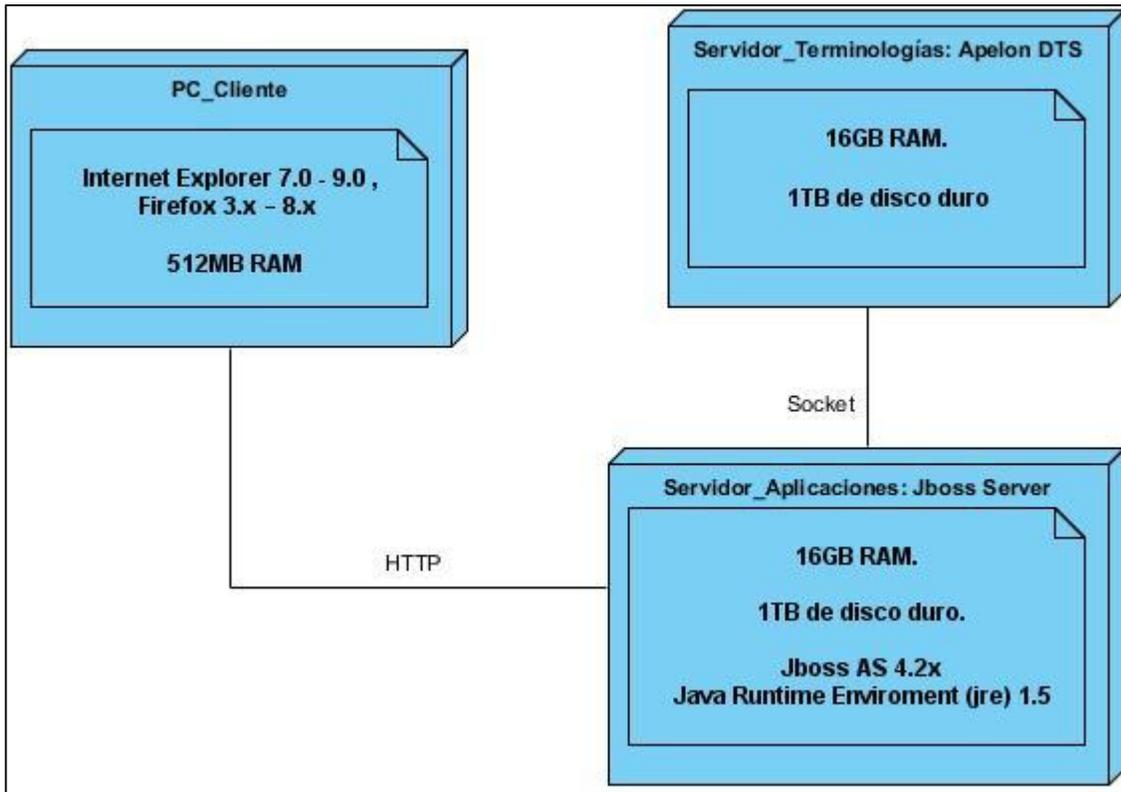


Fig. 3.1 Diagrama de despliegue.

Nodo PC_Cliente: Representa la computadora donde el usuario interactuará con un sistema.

Nodo Servidor de Aplicaciones (Jboss Server): Representa el servidor donde estará el sistema de Administración. Este tendrá comunicación con el nodo donde estará la el servidor de base de Datos.

Nodo Servidor de Terminologías Apelon DTS: Representa el servidor donde estarán almacenadas las terminologías médicas.

Descripción de las funcionalidades de la clase Apelon_DTS

Para realizar la codificación de un documento clínico se implementó una API de comunicación que permitió el acceso a Apelon_DTS y provee al usuario una serie de librerías para interactuar con las terminologías

almacenadas en el servidor. En la presente investigación se seleccionaron las siguientes para dar cumplimiento al desarrollo de la API de comunicación con el servidor DTS:

- 1) Common.
- 2) Dtsapps.
- 3) Dtscore.
- 4) Dtsmc.
- 5) Apelonbeans.
- 6) Apelonmessage.
- 7) Apelonserver.

Con las librerías seleccionadas anteriormente se implementó la API de comunicación Apelon_DTS mostrada en la siguiente tabla.

Apelon DTS
+ publicApelon_DTS
+ void initLog4j
+ ServerConnectionSocketConexion(String servidor,intpuerto)
+ Namespace[] ListarNamespaces(ServerConnectionSocket servidor_dts)
+ DTSConceptBuscarConcepto(ServerConnectionSocket dts, int namespace, String concepto)

Tabla 6.Descripción de la API de comunicación.

- Apelon_DTS: constructor de la clase utilizada solamente para instanciar la clase y poder utilizar los métodos que ofrece.
- initLog4j: funcionalidad en la cual se logra el acceso a la consola del servidor donde se puede observar las trazas de las operaciones realizadas por el cliente, cabe señalar que todo cliente basado en esta tecnología debe implementar esta funcionalidad de carácter obligatorio.
- Conexión: funcionalidad en la cual se logra la comunicación con el servidor mediante sockets, el servidor debe ser inicializado de forma independiente para lograr su correcto funcionamiento, los parámetros a pasar son solamente la dirección del servidor y el puerto por el cual está corriendo el servicio de sockets, y en su retorno se obtiene una instancia de conexión al servidor.

- **ListarNamespaces:** funcionalidad en la cual se listan todos los namespaces del servidor DTS entendiéndose por namespaces cada terminología registrada en el servidor DTS, recibe como parámetros una instancia de conexión al servidor DTS, y en su retorno se obtiene un arreglo con todos los namespaces registrados en el servidor
- **BuscarConcepto:** funcionalidad en la que se listan todos los conceptos asociados a una búsqueda previa realizado por el usuario, cabe destacar que el usuario no necesita conocer el nombre exacto del concepto a buscar, recibe como parámetro una instancia de conexión al servidor y el identificador del namespace seleccionado y se obtiene un arreglo de conceptos asociados a la búsqueda.

Estándares de diseño a utilizar

Para el diseño de la vista de la API de comunicación se tomaron las pautas del Diseño de Aplicaciones Web del CESIM, permitiendo igualdad en todos los productos del centro. A continuación se muestran algunas de las pautas por las que se ha regido el diseño de la misma.

El área de trabajo será la zona principal de la aplicación, donde se mostrarán los contenidos o formularios con los que se encuentre trabajando el usuario.

Etiquetas

- Las etiquetas estáticas que aparezcan irán en negro.
- El contenido de las etiquetas no debe contener preposiciones innecesarias.

Diagnóstico

Aceptar Cancelar

Criterios de búsqueda

Concepto:

Buscar

Resultados de búsqueda

Listado de enfermedades

Código	Descripción	Estructura
--------	-------------	------------

Selección de diagnóstico

Código	Descripción
--------	-------------

Fig. 3.2 Área de trabajo.

Mensajes

- El contenido de los mensajes se mostrará alineado a la izquierda, terminarán con punto final.
- En las ventanas de Información y Error se mostrará solo un botón de “Aceptar”, centrado en la parte inferior.
- Las ventanas de Advertencia utilizarán dos botones colocados en la parte inferior y tendrán los textos: “Si” y “No”. El botón “Si” estará a la izquierda y el “No” a la derecha.

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 a un sistema.

Utilizar técnicas de codificación sólidas y realizar buenas prácticas de programación, es de gran importancia para la calidad del software. La aplicación de estándares de codificación además posibilita que el software que se obtiene sea fácil de comprender y de mantener en el tiempo.

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 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.

Elementos de los estándares de codificación

Se empleará notación CamellCasing para denotar variables y parámetros, esta especifica que la palabra de inicio del identificador comienza con minúscula. Si el identificador está compuesto por más de una palabra entonces éstas deben comenzar con mayúsculas.

Para definir una robusta estructura y organización del código, se deben definir algunos estándares para su posterior entendimiento y cumplir con las buenas prácticas establecidas en la Ingeniería de Software. A continuación se resumen algunas de las convenciones tomadas en relación a estos aspectos.

Identación: Lograr una estructura uniforme para los bloques de código así como para los diferentes niveles de anidamiento.

Se recomienda dejar dos espacios en blanco desde la instrucción anterior para el inicio y fin de bloque `{}`. Lo mismo sucede para el caso de las instrucciones **if**, **else**, **for**, **while**, **do while**, **switch**, **foreach**.

Comentarios, separadores, líneas, espacios en blanco y márgenes: Establecer un modo común para comentar el código de forma tal que sea comprensible con sólo leerlo una vez.

Se recomienda comentar al inicio de la clase o función especificando el objetivo de la misma así como los parámetros que utiliza (especificar tipos de dato, y objetivo del parámetro) entre otras cosas.

Se recomienda dejar una línea en blanco antes y después de la declaración de una clase o de una estructura y de la implementación de una función.

Se recomienda utilizar espacios en blanco entre los operadores para lograr una mayor legibilidad en el código. Ejemplo: `producto = nomproducto`.

Para nombrar las funciones se debe tratar de utilizar verbos que denoten la acción que hace la función. Se empleará notación PascalCasing*. Ejemplo: `function BuscarUnidad()`. Si son funciones que obtienen un dato se emplea el prefijo **get** y si fijan algún valor se emplea el prefijo **set**.

El nombre empleado para las clases, objetos, atributos y funciones debe permitir que con sólo leerlo se conozca el propósito de los mismos.

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 (30). Cuando dicho error ocurre dentro de un método Java, automáticamente se crea un objeto '**Exception**' el cual es tratado en el sistema de ejecución. Este objeto contiene información sobre la excepción, incluyendo su tipo y el estado del programa cuando ocurrió el error. 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.

En el caso de Java se utilizan bloques **try - catch - finally**, **throw** y **throws**, y las subclases de **java.lang.Exception**. Un método lanzará (mediante un **throw**) una excepción (una clase que hereda **java.lang.Exception**) provocando que se abandone el flujo de ejecución normal y cediendo el control al bloque en el que se capturará (en un bloque **try - catch - finally**). En la declaración del método se incluye **throws** para avisar de que puede lanzar una excepción. En terminología java, crear un objeto **exception** y manejarlo por el sistema de ejecución se llama lanzar una excepción.

En la solución propuesta se propone el tratamiento de excepciones principalmente en las regiones críticas de código, que son los fragmentos donde se manipulan datos que son insertados por el usuario. El manejo de las excepciones o errores, en las clases controladoras de procesos, utilizará el bloque **try** para detectar cuando ocurra algún fallo y un bloque **catch** donde se manejarán dichas excepciones, mediante mensajes que se muestran en la interfaz de usuario.

Ejemplo de uso:

```
try{  
  
//declaración que causa la excepción  
  
}  
  
catch  
  
(NombredeExcepcion obj)  
  
{  
  
//código para tratar el error  
  
}
```

Seguridad.

La seguridad es un tema de gran importancia para cualquier Sistema de Información y toma mayor relevancia cuando se gestiona información médica. Para garantizar la seguridad de la solución desarrollada se sostendrá un control a nivel de usuarios y contraseñas, permitiendo el acceso por tipo de usuario logrando

así la visibilidad sólo a las áreas establecidas de acorde a la función que realizan. Las contraseñas solo podrán ser cambiadas por el usuario o por el administrador del sistema.

Otra cuestión es lograr la fiabilidad en las estaciones de trabajo, para lograr esto se define un segundo nivel de seguridad a nivel de estaciones de trabajo lo que posibilita la ejecución sólo de las aplicaciones que hayan sido definidas para la estación en cuestión.

3.2 Validación de la propuesta de solución

La validación es la evaluación del *software* al final del proceso de desarrollo del *software* para determinar su conformidad. Implica la valoración de los productos de trabajo para determinar el apego a las especificaciones, incluyen las especificaciones de requisitos, la documentación del diseño, diversos principios generales de estilo, estándares del lenguaje de instrumentación, estándares del proyecto, estándares organizacionales y expectativas del usuario, al igual que las meta especificaciones para los formatos y notaciones utilizadas en la especificación de productos diversos.

Los objetivos de las actividades de validación son valorar y mejorar la calidad de los productos del trabajo generados durante el desarrollo y modificación del software. Los atributos de la calidad deben ser la corrección, la perfección, la consistencia, la confiabilidad, la utilidad, la eficacia, el apego a los estándares y la eficacia de los costos totales.

Para comprobar la utilidad y pertinencia de la **API de comunicación** se escogió el proceso de codificación en la persistencia de información relevante de los pacientes en la **Arquitectura de Documentos Clínicos** (CDA – por sus siglas en inglés), estándar de marcado *Extensible Markup Language* (XML) de documentos que especifica la estructura y la semántica de "documentos clínicos" con el propósito de intercambio de información, puede incluir texto, imágenes, sonidos y otros contenidos multimedia y pueden ser transmitidos en mensajes HL7 diseñados para transferir documentos clínicos. Entre sus características fundamentales se encuentran:

- Puede existir fuera del contexto de mensajes.
- No especifica la creación o la gestión de documentos, sólo su marcado para intercambio.
- Debe ser legible usando navegadores ampliamente disponibles, utilizando una hoja de estilos genérica en un lenguaje de hojas de estilo estándar.

- Debe ser posible renderizar todos los documentos con una sola hoja de estilos.
- Los documentos CDA pueden ser agregados a documentos ya existentes.

Un documento CDA contiene una cabecera y un cuerpo. La cabecera sigue una estructura común, fácilmente consultable, que identifica y clasifica el documento, provee información acerca de la autenticación, el encuentro, el paciente, autor y autores involucrados. Al seguir una estructura común, bien definida, la consulta de estos campos de una forma automatizada es fácil. A continuación se muestran los elementos de la cabecera de un documento CDA.

- **typeid:** es una referencia al *release 2* de la especificación CDA y debe ser evaluado de la siguiente forma:
root = "2.16.840.1.113883.1.3"
extension = "POCD_HD000040" identificador único para CDA, Release Two Hierarchical Description.
- **class:** valor por defecto DOCCLIN, documento de observaciones clínicas y servicios.
- **Id:** identificador único del documento.
- **code:** especifica un tipo particular de documento (resumen de alta, nota de procedimiento).
- **title:** título del documento, a menudo se toma del tipo de documento.
- **effectiveTime:** fecha y hora de creación del documento.
- **ConfidentialityCode:** código de confidencialidad.
 - N (normal) (codeSystem 2.16.840.1.113883.5.25)
 - R (restricted) (codeSystem 2.16.840.1.113883.5.25)
 - V (very restricted) (codeSystem2.16.840.1.113883.5.25)
- **setId:** representa un identificador común a todos los documentos que pertenecen a actualizaciones de un mismo documento.
- **versionNumber:** valor entero correspondiente a las sucesivas actualizaciones de un documento.

El cuerpo del documento puede contener tres niveles de implementación:

- **El nivel 1** es aquel que se transmite en el cuerpo del mensaje un bloque de datos sin ninguna estructura definida (un "blob" que puede ser cualquier cosa, un texto, una imagen, un archivo PDF, entre otros.)

- **El nivel 2** sigue una estructura XML bien definida con secciones de información identificadas, pero el contenido es texto libre. Por ello se conoce como "bloque" narrativo. Esta estructura permite la presentación de este texto en diferentes formatos mediante transformaciones XML, ya que las secciones y su contenido son bien definidas.
- **El nivel 3** agrega a cada sección, y a cada dato dentro de esas secciones (diagnósticos, unidades de medición, medicamentos, entre otros) una codificación de vocabulario estricta (*LOINC*, *SNOMED*, *CIE 9*, entre otros o la propia del centro), con el fin de ser procesable computacionalmente. Este nivel trae muchas ventajas, la verdadera interoperabilidad semántica que permite que los documentos sean altamente procesables, interoperables y sin ambigüedades (31).

Actualmente los CDA del CESIM se encuentran en un nivel 2 de implementación. Con el objetivo de proporcionarle a los sistemas del CESIM el acceso a las terminologías necesarias para realizar la persistencia de información codificada fue seleccionado como objeto de prueba el documento clínico "**Hoja de evolución médica**". Este documento se genera en la atención del paciente dentro del módulo de Hospitalización del Sistema de Información Hospitalaria (HIS – por sus siglas en inglés) del CESIM. Recoge todos los acontecimientos ocurridos al paciente durante su estancia hospitalaria o en las sucesivas revisiones ambulatorias. Además, registra los juicios clínicos y la justificación de las decisiones terapéuticas. No presenta gran complejidad y contiene una sección de diagnósticos que puede ser codificada.

Para llevar a cabo la validación de la propuesta de solución fue necesario realizar modificaciones en la clase **GeneratorCDA_hospitalizacion** y crear una interfaz visual en la línea base adoptada, a través de la cual los usuarios (en este caso los médicos) podrán buscar y posteriormente seleccionar las enfermedades que van a conformar el diagnóstico final que será creado en la **Hoja de evolución** médica del paciente.

Modificaciones de la clase GeneratorCDA_hospitalizacion

A continuación se muestran las modificaciones realizadas a la clase GeneratorCda_hospitalización.

Se añadieron los siguientes atributos:

Atributos	Descripción
Private apelon_dts server	Crea un objeto de tipo Apelon_DTS que es a su vez el API de comunicación que se propone en la solución.
Private String illselected	Guarda la enfermedad que se va a seleccionar para que no se repita en el diagnóstico.
Private String problema_search	Guarda el concepto por el cual se realizará la búsqueda.
Private List<Enfermedad>enfermedades_search	Guarda los conceptos arrojados por la búsqueda en esta variable.
Private List<Enfermedad>enfermedades_selected	Guarda los conceptos seleccionados para establecer un diagnóstico.

Tabla 7: Atributos añadidos a la clase GeneratorCda_hospitalización.

Se añadieron los siguientes métodos:

Método	Descripción
public void SeleccionarConcepto()	Se utiliza cuando ya se haya realizado una búsqueda y el atributo enfermedades_search. Se encarga de verificar que el concepto que se quiera añadir ya no esté guardado en enfermedades_search, si es así el concepto se añade al atributo enfermedades_selected.
public void EliminarConcepto(String codigo)	Opera sobre el atributo enfermedades_selected y garantiza que si el usuario no quiere tener ese concepto en la lista de seleccionados lo elimina de la misma
public void BuscarConcepto()	Opera sobre el atributo enfermedades_search y es el encargado de realizar la búsqueda del concepto solicitado.

<code>private void BuscarAux(DTSConcept[] con)</code>	Convierte el resultado de la búsqueda a objetos que guardan las enfermedades, pues por este tipo de objeto se establece el diagnóstico.
---	---

Tabla 8: Métodos añadidos a la clase GeneratorCda_hospitalizacion.

Componentes del cuerpo de la Hoja de evolución médica

El área de diagnósticos permitirá la entrada de los padecimientos del paciente y estará constituida por:

- Un elemento <component> el cual contendrá un elemento <section> correspondiente a la sección de Diagnóstico.
- Un elemento <code> donde se registra el nombre y el código del estándar de codificación utilizado.
- Un elemento <title> donde se establece el título de la sección.
- Un elemento <text> que contendrá un elemento <list> dentro del cual se encontrarán los elementos <item> en los cuales se producirá la entrada de los conceptos seleccionados por el usuario.

3.3 Validación CDA Hoja de evolución médica generada

La validación del documento Hoja de evolución médica generado se utilizaron fuentes de validación *online* propuestas en el sitio oficial de HL7, disponible en el Lantana Consulting Group (www.lantanagroup.com). Es una organización que se centra en el diseño e implementación de sistemas compatibles con CDA, incluida la validación, gestión de documentos, edición, conversión de datos y servicios web para el intercambio de información con amplia experiencia y profundidad en XML, terminologías de atención de la salud y la gestión de proyectos. Este validador *online*, se encarga de:

1. Verificar el correcto formato XML del documento CDA a validar.
2. Verificar el esquema del documento CDA a validar contra *CDA Schema*.

En el capítulo anteriormente presentado se concreta la implementación de la solución siguiendo lo establecido en la Especificación de Requisitos de Software, además se modela el diagrama de despliegue, que permite representar como estará ubicado el servidor de terminologías. Se describieron los elementos que fueron necesarios para realizar persistencia en la Arquitectura de Documentos Clínicos para comprobar

el correcto funcionamiento de la solución. De esta forma queda conformada y validada una solución que posibilita el acceso a las terminologías del servidor Apelon DTS.

Conclusiones

Con el presente trabajo de diploma se ha cumplido con el objetivo general propuesto, así como con las tareas de la investigación, obteniéndose las siguientes conclusiones:

- En los desarrollos actuales del CESIM se utilizan los codificadores: CIE 9 CIE 10 LOINC, NANDA ACR, CIAP y ATC.
- Se seleccionó el Servicio de Terminología Distribuida Apelon DTS como alternativa más efectiva ante la solución planteada.
- Se propone la integración del Apelon DTS a la implantación de los sistemas del CESIM como Servicios de Terminología Distribuidas y para ello se implementó el API de comunicación Apelon_DTS con las funcionalidades necesarias para el acceso al servicio.
- Se implementó como caso de prueba la persistencia del CDA “Evolución del paciente” utilizando como codificador el servicio propuesto. Los resultados al comprobar en los validadores online fueron satisfactorios.

Recomendaciones.

La solución propuesta puede ser enriquecida con algunas funcionalidades que pueden aportar eficiencia.

Por esta razón se recomienda a los desarrolladores del Centro:

- Realizar la codificación a nivel 3 de los restantes documentos clínicos utilizados en el CESIM.
- Implementar un componente de apoyo a la toma de decisiones que utilice los servicios de Apelon DTS.
- Mantener un seguimiento de las versiones posteriores de servidor de terminologías médicas Apelon DTS.

Referencias Bibliográficas

1. Marín, ME. informatica2013. [En línea] 2013. [Citado el: 15 de Enero de 2014.] <http://www.informatica2013.sld.cu/index.php/informaticasalud/2013/paper/viewFile/>.
2. Hospital Italiano de Buenos Aires. Componentes de un sistema de Información en salud. [Documento] Buenos Aires : s.n.
3. SB, Johnson. Anelectronic health record based on structured. [Documento] USA : J Am Med Inform Assoc, 2008.
4. Rivas, Karla. Slidershare. [En línea] 11 de Noviembre de 2008. [Citado el: 10 de Marzo de 2014.] <http://www.slidershare.net/krivas/codificacion-medica-ctpicdpi>.
5. Slidershare. [En línea] 11 de Noviembre de 2008. [Citado el: 10 de Marzo de 2014.] <http://www.slidershare.net/krivas/codificacion->.
6. González, Alberto Bradshaw. Servidor de Terminología Médica. [Documento] La Habana : UCI, 2012.
7. Ratzan, L. Understanding information systems : what they do and why we need . [Documento] Chicago : American Library Association, 2004.
8. González, Reinier Alosnso. Modelo de producción de software para el CESIM. [Documento] Ciudad de la Habana : UCI, 2010.
9. Gambarte, M.L. A practical approach to advanced terminology services in health information systems. [Documento]
10. Libicki, M y Brahmakulam, I. The Costs and Benefits of Moving to the ICD-10 Code Sets. [Documento] California : RAND Corporation, 2004.
11. Tecnologías de la información al servicio de una gestión y salud de calidad, . [En línea] 2010. [Citado el: 15 de Marzo de 2014.] <http://www.informaticamedica.cl/2012/07/que-es-loinc.html>.
12. NANDA. NANDA Internacional. [En línea] 2014. [Citado el: 15 de Marzo de 2014.] <http://www.nanda.org/>.

13. Infomed. Glosario de terminos farmacologicos. [En línea] 29 de Abril de 2011. [Citado el: 7 de Junio de 2014.] <http://glosario.sld.cu/terminos-farmacologicos/2011/04/29/clasificacion-anatomica-terapeutica-y-quimica-atc/>.
14. Object Management Group. Lexicon Query Service Specification. [Documento] s.l. : Object Management Group, 2000.
15. Health Level Seven. Health Level Seven International. [En línea] 2014. [Citado el: 20 de Marzo de 2014.] http://www.hl7.org/implement/standards/product_brief.cfm?product_id=10.
16. Apelon. Apelon DTS. [En línea] Apelon, 2014. [Citado el: 20 de Marzo de 2014.] <http://www.apelondts.org>.
17. Sánchez, Jesús Portolés. Búsqueda Semántica en Repositorios de Conceptos Biomédicos Estandarizados. [Documento] Valencia : s.n., 2010.
18. Common Terminology Services . [En línea] 2014. <http://informatics.mayo.edu/LexGrid/index.php?page=ctsspec>.
19. IHE Internacional. IHE_ITI_TF. [Documento] 2013.
20. Hackelare. [En línea] 2014. [Citado el: 25 de Febrero de 2014.] <http://hackelare.wordpress.com/2010/06/17/hibernate-para-nonos-introduccion>.
21. WebTaller. [En línea] 2014. [Citado el: 01 de Abril de 2014.] <http://www.webtaller.com/construccion/lenguajes/java/lecciones/que-es-java.php>.
22. PostgreSQL investigacion. [En línea] 2014. [Citado el: 02 de Abril de 2014.] <http://www.scribd.com/doc/36570462/postgreSQL-investigacion>.
23. Visual Paradigm para UML. [En línea] 2014. [Citado el: 02 de Abril de 2013.] <http://www.software.com.ar/visual-paradigm-para-uml.html..>
24. USMP. [En línea] 2014. [Citado el: 24 de Febrero de 2014.] <http://www.usmp.edu.pe/publicaciones/boletin/fia/info49/articulos/RUP%20vs.%20XP.pdf>.

25. Synergix. Tecnología y Synergix. [En línea] 2014. [Citado el: 04 de Junio de 2014.] <http://synergix.wordpress.com/2008/07/10/modelo-de-dominio/>.
26. Rafa. ¿Qué es un requerimiento? . clases3gingsof.wetpaint.com. [En línea] [Citado el: 07 de Junio de 2014.] <http://clases3gingsof.wetpaint.com/page/%C2%BFQu%C3%A9+es+un+requerimiento%3F..>
27. Definición de Casos de uso. Definición de Casos de uso. [En línea] [Citado el: 4 de junio de 2014.]
28. Diagramas de Interaccion de Objetos. . Diagramas de Interaccion de Objetos. . [En línea] [Citado el: 1 de junio de 2014.]
29. Tema 9: Diseño de Software, Departamento de Lenguajes y Sistemas Informáticos II. Tema 9: Diseño de Software, Departamento de Lenguajes y Sistemas Informáticos II. [En línea] [Citado el: 18 de mayo de 2014.]
30. Tratamiento de excepciones. Tratamiento de excepciones. [En línea] [Citado el: 20 de mayo de 2014.] <http://elvex.ugr.es/decsai/builder/intro/6.html>.
31. Ramírez, Rodney Ledo. *Conferencia HL7/CDA*. [Documento] La Habana : Universidad de las Ciencias Informáticas.

Bibliografía

1. Apelon. Apelon DTS. [En línea] Apelon, 2014. [Citado el: 20 de Marzo de 2014.] <http://www.apelondts.org>.
2. Bautista, Alfredo. Scribd. [En línea] 2014. [Citado el: 15 de Enero de 2014.] <http://www.scribd.com/doc/56428211/QUE-ES-TERMINOLOGIA-MEDICA>.
3. Corneta, Ronald y de Keizera, Nicolette F. Overcoming Barriers to Evaluation of. [Documento] s.l.: IMIA, 2004.
4. Hackelare. [En línea] 2014. [Citado el: 25 de Febrero de 2014.] <http://hackelare.wordpress.com/2010/06/17/hibernate-para-nonos-introduccion>.
5. Health Level Seven. Health Level Seven International. [En línea] 2014. [Citado el: 20 de Marzo de 2014.]
6. HL7. Health Level Seven International CDA. [En línea] 2014. [Citado el: 20 de Marzo de 2014.]
7. Hospital Italiano de Buenos Aires. Componentes de un sistema de Información en salud. [Documento] Buenos Aires: s.n.
8. IHE International. IHE IT Infraestructure (ITI) Technical Framework. [Documento] 2013.
9. IHE Internacional. IHE_ITI_TF. [Documento] 2013.
10. Instituto Universitario del Hospital Italiano. Curso de introducción a la Informática Biomédica: Componentes de un sistema de información de la salud. [Documento] Buenos Aires: s.n.
11. Marín, ME. informatica2013. [En línea] 2013. [Citado el: 15 de Enero de 2014.] <http://www.informatica2013.sld.cu/index.php/informaticasalud/2013/paper/viewFile/>.
12. NANDA. NANDA Internacional. [En línea] 2014. [Citado el: 15 de Marzo de 2014.] <http://www.nanda.org/>.
13. Object Management Group. Lexicon Query Service Specification. [Documento] s.l.: Object Management Group, 2000.

14. PostgreSQL investigación. [En línea] 2014. [Citado el: 02 de Abril de 2014.] <http://www.scribd.com/doc/36570462/postgreSQL-investigacion>.
15. Rivas, Karla. Slidershare. [En línea] 11 de Noviembre de 2008. [Citado el: 10 de Marzo de 2014.] <http://www.slidershare.net/krivas/codificacion-medica-ctpicdnpi>.
16. Ratzan, L. Understanding information systems: what they do and why we need. [Documento] Chicago: American Library Association, 2004.
17. Radiologyinfo. [En línea] 2014. [Citado el: 15 de Marzo de 2014.] http://www.radiologyinfo.org/sp/about/index.cfm?pg=abt_acr.
18. Ramírez, Rodney Ledo. Conferencia HL7/CDA. [Documento] La Habana: Universidad de las Ciencias Informática.
19. Rivera. Sistemas WDS para la Administración remota de servidores. [En línea] 2014. [Citado el: 02 de Febrero de 2014.] <http://es.scribd.com>.
20. SB, Johnson. Anelectronic health record based on structured. [Documento] USA: J Am Med Inform Assoc, 2008.
21. Slidershare. [En línea] 11 de Noviembre de 2008. [Citado el: 10 de Marzo de 2014.] <http://www.slidershare.net/krivas/codificacion-medica-ctpicdnpi>.
22. Tecnologías de la información al servicio de una gestión y salud de calidad. [En línea] 2010. [Citado el: 15 de Marzo de 2014.] <http://www.informaticamedica.cl/2012/07/que-es-loinc.html>.
23. USMP. [En línea] 2014. [Citado el: 24 de Febrero de 2014.] <http://www.usmp.edu.pe/publicaciones/boletin/fia/info49/articulos/RUP%20vs.%20XP.pdf>.
24. Visual Paradigm para UML. [En línea] 2014. [Citado el: 02 de Abril de 2013.] <http://www.software.com.ar/visual-paradigm-para-uml.html>. 25WebTaller. [En línea] 2014. [Citado el: 01 de Abril de 2014.] <http://www.webtaller.com/construccion/lenguajes/java/lecciones/que-es-java.php>.

Glosario de Términos-

Actor: es una entidad externa al sistema que se modela y que puede interactuar con él. Puede ser una persona o un grupo de personas homogéneas, otro sistema, o una máquina. Los actores son externos al sistema que vamos a desarrollar. Por lo tanto, al identificarlos, estamos comenzando a delimitar el sistema y a definir su alcance.

Actores IHE: son sistemas de información o componentes de un sistema de información que producen, gestionan y actúan sobre la información asociada a actividades operativas de la empresa de salud. Cada uno de los actores recibe un nombre específico que los identifica de forma no ambigua.

Administración de requisitos: la administración de todos los requisitos recibidos o generados por el proyecto, incluyendo requisitos técnicos y no técnicos, así como esos requisitos levantados en el proyecto por la organización.

API: Interfaz de programación de aplicaciones (IPA) o API (del inglés *Application Programming Interface*) es el conjunto de funciones y procedimientos (o métodos, en la programación orientada a objetos) que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

Caso de Uso: secuencias de acciones que el sistema puede llevar a cabo interactuando con sus actores, incluyendo alternativas dentro de las secuencias.

CDA: estándar de marcado XML de documentos que especifica la estructura y la semántica de "documentos clínicos" con el propósito de intercambio de información, puede incluir texto, imágenes, sonidos y otros contenidos multimedia y pueden ser transmitidos en mensajes HL7 diseñados para transferir documentos clínicos.

Codificación de términos médicos: es el proceso de transformación de las descripciones de los diagnósticos y procedimientos médicos en códigos médicos universales.

Estándar: serie de reglas y definiciones que especifican cómo llevar a cabo un proceso o producir un producto.

Framework: estructura conceptual y tecnológica de soporte definida, normalmente con artefactos o módulos de software concretos, con base en la cual otro proyecto de software puede ser organizado y desarrollado.

HL7: es una de las organizaciones más importantes en estándares de intercambio de información en salud. Surgió en los Estados Unidos a finales de la década del 80 como una iniciativa para lograr un Sistema de Información Hospitalario (HIS) integrado, interconectando distintos sistemas específicos.

Interoperabilidad: capacidad de dos o más sistemas de intercambiar y utilizar información entre ellos, de esta definición se desprenden dos tipos de interoperabilidades la sintáctica (operativa o funcional) y la interoperabilidad semántica.

Interoperabilidad sintáctica: se encarga de comunicar diferentes sistemas preexistentes o nuevos desde el punto de vista operativo o funcional.

Interoperabilidad semántica: capacidad de los sistemas informáticos de comunicar, incorporar y utilizar información generada por sistemas externos.

Perfil de integración IHE: se puede definir un Perfil de Integración IHE como la agrupación de actores, transacciones y vocabulario común para realizar una tarea típica de flujo de trabajo en un servicio o dominio correspondiente.

Requisito: condición o capacidad que debe poseer un producto o componente de producto para satisfacer un contrato, estándar, especificación, u otros documentos obligatorios formales.

Reglas de Negocio: las reglas del negocio describen las políticas, normas, operaciones, definiciones y restricciones presentes en una organización y son de vital importancia para alcanzar sus objetivos.

Servicios terminológicos: permiten que la información clínica se almacene de una forma estándar y reutilizable, no importa el punto de origen de la misma.

Sistema de Información en Salud: es aquel que está constituido por un conjunto de instrucciones organizadas, sistematizadas y lógicas que se relacionan entre sí por medio de un lenguaje informático. Persigue el fin de obtener información, analizarla, relacionarla y generar nueva información para satisfacer las necesidades de las áreas administrativas, operativas de una organización en general. Este provee

comunicación entre los miembros del equipo de salud y da soporte organizacional a la gestión de la información para realizar operaciones, planeamiento, atención del paciente y el registro de sus actos.

Terminología médica: la terminología médica tiene el propósito de expresar en términos precisos los complejos conceptos e ideas del mundo de la medicina. También tiene como propósito la unificación de criterios. Cada término debe poseer un significado único aceptado por la comunidad científica, facilitando, así, el intercambio de información a nivel internacional.

Transacción: son interacciones entre actores que se transfieren la información requerida a través de la utilización de mensajes basados en estándares.

Usabilidad: la usabilidad, hace referencia, a la rapidez y facilidad con que las personas llevan cabo sus tareas propias a través del uso del producto objeto de interés.