

**Universidad de las Ciencias Informáticas**

**Facultad 7**



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

# **COMPONENTE REPOSITORIO DE DATOS CLÍNICOS DE LA HISTORIA CLÍNICA ELECTRÓNICA PARA EL CENTRO DE INFORMÁTICA MÉDICA**

---

**Autores:**

Ismary Frontela Cabrera

Lorenzo Angulo Díaz

**Tutor:**

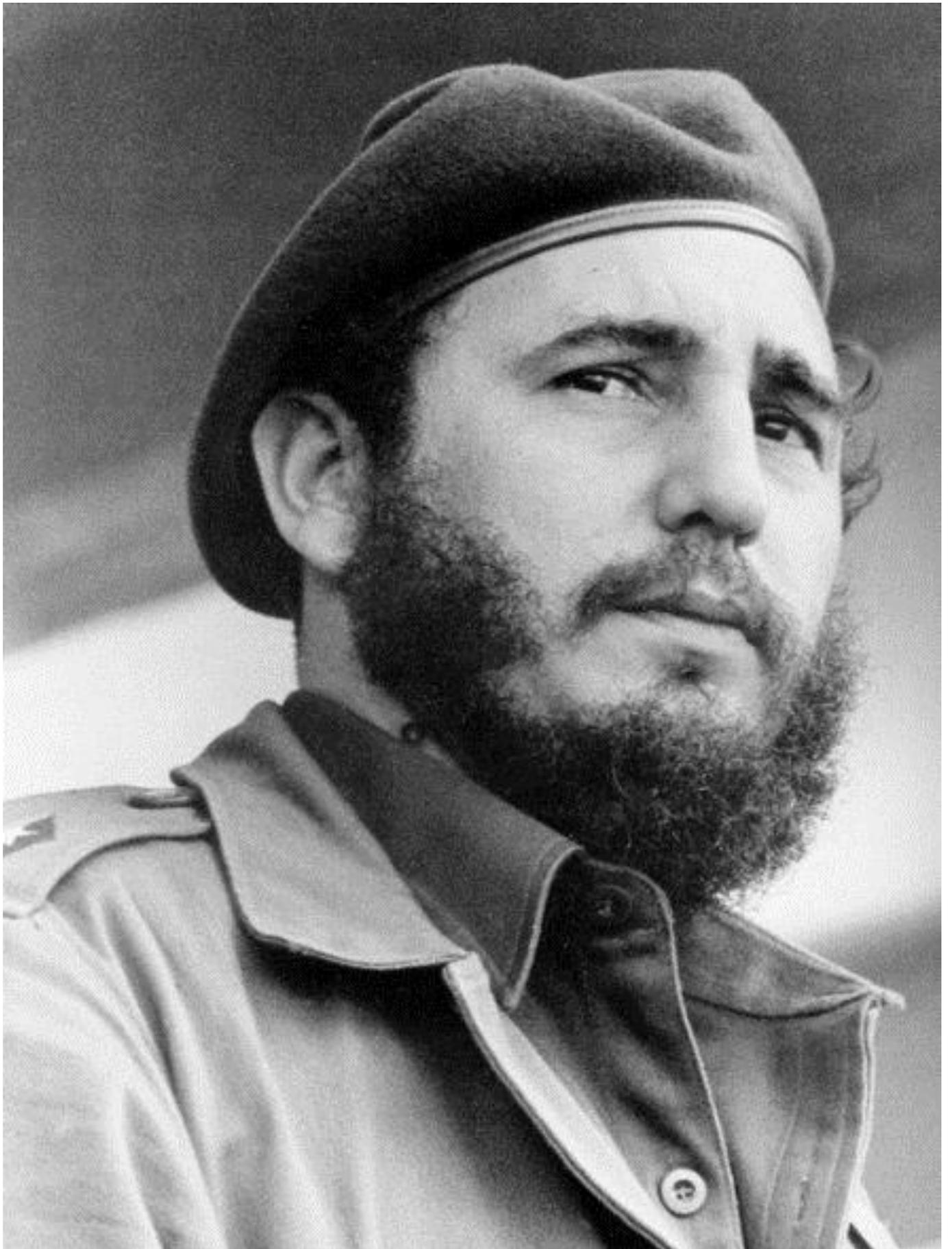
Ing. Raymundo Rodríguez García

**Co-tutor:**

Ing. Nuria Isabel Betanco Alvarez

**Año 55 de la Revolución**

**La Habana 2013**



## DATOS DE CONTACTO

### **Ing. Raymundo Rodríguez García**

Especialista. Graduado de Ingeniero en Ciencias Informáticas, egresado de la UCI en el 2010. Se desempeña actualmente como Jefe de Proyecto y Administrador de la configuración del Departamento de Tecnologías, Integración y Estándares del Centro de Informática Médica (CESIM), en la Universidad de las Ciencias Informáticas (UCI).

Correo electrónico: rrdguez@uci.cu

### **Ing. Nuria Isabel Betanco Alvarez**

Profesor en adiestramiento. Graduada de Ingeniero en Ciencias Informáticas, egresado de la UCI en el 2011. Se desempeña actualmente como Gestor del Conocimiento del Departamento de Tecnologías, Integración y Estándares del CESIM, en la UCI.

Correo electrónico: nibetanco@uci.cu

### AGRADECIMIENTOS

A mi padre Marino Frontela Reyes que ha sido mi inspiración durante toda mi vida. Gracias por ser el mejor padre del mundo, por apoyarme en todo momento, por depositar esa confianza en mí, por tu dedicación, amor, ternura y comprensión. Gracias por tu paciencia y entrega, por estar siempre a mi lado, gracias por guiarme hacia el camino del éxito y convertirme en la persona que soy hoy, te quiero mucho.

A mi abuelita Zenaida Cabrera Pita que le debo todo lo que soy hoy, gracias por tu amor y por quererme tanto como me quieres y por estar ahí siempre que te necesito. Te amo.

A Idalmis Barrios Machado que le debo tanto, que ha sido para mi madre, compañera y amiga, que siempre me ha apoyado estando a mi lado, gracias por tu cariño incondicional, por escucharme siempre, gracias por tu amor. Te quiero con la vida.

A mi mamá Iliana Cabrera Cabrera que a pesar de estar lejos de mí, sé que me quiere mucho y que confía plenamente en mí, gracias por tu amor, te quiero mucho.

A mi novio Ariel que ha sido de gran apoyo para mí durante estos 2 años, gracias por tu dedicación y cariño, que me han ayudado a seguir adelante y romper cualquier obstáculo que se ha puesto en el camino, gracias por tu apoyo. Te amo

A mis hermanos Yoan, Yoney y Rainel por su apoyo y por su amor, gracias por ser los mejores hermanos del mundo, son mi vida, los amo mucho a todos.

A mi familia de Pinar del Río y a la de la Lisa, por sus consejos, apoyo, dedicación y amor. Los quiero mucho y siempre los llevaré en mi corazón y presente en mi pensamiento.

A mis tutores Raymundo Rodríguez García y Nuria Isabel Betanco Alvarez quienes con su paciencia y entrega me han guiado en la elaboración de este trabajo, sin la ayuda de ellos nunca lo hubiera logrado, gracias por su apoyo y dedicación.

A mis amistades a Humberto, Laura, Elizandra, Ariel, Diego, Lorenzo a todos los que en un momento u otro me han ayudado y siempre han estado cuando los necesito. Los quiero mucho.

*A la Revolución Cubana, a Fidel, a Raúl, a mis profesores, a todos los que de una forma u otra me han dado la posibilidad de realizar este sueño, gracias.*

*A todos ustedes muchas gracias.*

**Ismary**

*Agradezco con todo mi corazón:*

*A mi familia, que siempre me ha apoyado y ha velado por mi formación como persona.*

*A la Revolución Cubana, que gracias a ella estoy en esta Universidad.*

*A mis tutores Nuria Isabel Betanco Alvarez y Raymundo Rodríguez García que siempre han estado al tanto de nuestro trabajo, apoyándonos en todo momento y transmitiéndonos todos sus conocimientos en cualquier instante.*

*A mi compañera de tesis por todo el trabajo incansable que realizó junto conmigo.*

*A mis compañeros y amigos que han estado a mi lado durante estos 5 años brindándome su apoyo, entre los que vale destacar a Raidel, Yoandy, Randy, Manuel, Mercedes, a mis compañeros de grupo durante todos estos años y tantos otros, que me sería imposible mencionar en tan poco espacio.*

*A todos los profesores que de una forma u otra han contribuido a mi desarrollo, ya sean de la Universidad como de enseñanzas anteriores.*

**Lorenzo**

**DEDICATORIA**

*A mi abuelita del alma Zenaida, quien me ha cuidado y me ha protegido tanto durante toda mi vida.*

*A mi papá Marino y a Idalmis quienes a lo largo de mi carrera han velado por mi bienestar y educación siendo mi apoyo en todo momento, depositando su entera confianza en cada reto que se me presentaba sin dudar ni un solo momento en mi inteligencia y capacidad. Es por ello que he llegado a ser lo que soy ahora.*

*A mi mamá linda Iliana, que siempre supo que yo podía lograr cualquier propósito y me ha apoyado tanto.*

*A mis hermanos, a mi novio, a mis tíos, a mis tías, a mi otra abuelita Adelina, a mis primos, a toda mi familia, que es para mí la mejor del mundo.*

*Los amo con mi vida.*

**Ismary**

*A mi familia, a mis padres que siempre han sabido darme un buen consejo para todo, que siempre dedicaron todo su esfuerzo para que hoy pudiera estar ante ustedes.*

*A mi hermano que va a comenzar la vida como universitario.*

*A mis abuelos que a pesar de no conocer mucho de esta rama siempre han estado motivándome a estudiar, esforzarme, y prepararme cada días más, a todos ellos que han vuelto a ser estudiantes y se gradúan hoy junto conmigo.*

*A mi Revolución Cubana y a todos aquellos que han consagrado su vida a ella.*

*A los comandantes Fidel Castro Ruz y Raúl Castro Ruz, que han creado esta Universidad de Luz. A todos los que de una forma u otra han contribuido a mi formación como profesional y han hecho posible que hoy pueda defender mi Trabajo de Diploma.*

**Lorenzo**

## RESUMEN

A partir de la necesidad de crear una Historia Clínica Electrónica, en el Centro de Informática Médica se decide crear un repositorio que almacene la información con que se trabaja en los diferentes sistemas del mismo. Estos sistemas no cuentan con un almacenamiento único de datos clínicos, guardan sus datos específicos en repositorios propios e independientes que generalmente presentan algunas deficiencias: los datos que existen se encuentran dispersos, duplicados y en muchos casos no son accesibles. Por ello se plantea como objetivo general desarrollar un Repositorio de Datos Clínicos que proporcione el almacenamiento de la información clínica gestionada por los sistemas que conforman la Historia Clínica Electrónica del Centro de Informática Médica.

Para el desarrollo del componente se utilizó como metodología de desarrollo RUP regido por las normas de calidad de CMMI en su nivel 2, UML en su versión 2.1 como lenguaje de modelado, Enterprise Architect en su versión 7.5 como herramienta para modelar, Java como lenguaje de programación, utilizando como IDE de desarrollo Eclipse en su versión 3.1 y como gestor de base de datos PostgreSQL en su versión 9.1.

Un Repositorio de Datos Clínicos, centralizado y estandarizado que gestione e integre toda la información clínica contenida en los sistemas del Centro de Informática Médica, permitirá un almacenamiento único de la información y contribuirá con la creación de una Historia Clínica Electrónica que contará con su propio repositorio de datos.

**Palabras clave:** *Datos Clínicos, Historia Clínica Electrónica, HL7, Repositorio de Datos Clínicos, Servicio de Mensajería HL7.*

## ÍNDICE

INTRODUCCIÓN .....	1
CAPÍTULO 1: FUDAMENTACIÓN TEÓRICA DE REPOSITORIO DE DATOS CLÍNICOS .....	6
1.1 Conceptos fundamentales.....	6
1.2 Análisis de las soluciones existentes a nivel internacional.....	6
1.3 Análisis de las soluciones existentes a nivel nacional .....	9
1.4 Estándares relacionados con la solución .....	12
1.5 Descripción de tecnologías, herramientas y metodología propuestas para el proceso de desarrollo .....	14
CAPÍTULO 2: INGENIERÍA DEL SISTEMA .....	21
2.1 Propuesta de sistema .....	21
2.2 Servicio de Mensajería HL7 .....	24
2.3 Modelo de Dominio .....	25
2.4 Especificación de requerimientos de software.....	26
2.4.1 Requerimientos funcionales.....	27
2.4.2 Requerimientos no funcionales .....	32
2.5 Casos de uso del sistema.....	34
2.6 Descripción textual de algunos casos de uso .....	37
2.7 Arquitectura en Capas .....	42
2.8 Patrones de diseño .....	43
2.9 Diagramas de Clases del Diseño .....	45
2.10 Diagramas de Interacción.....	46
CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBA .....	49
3.1 Modelo de datos.....	49
3.1.1 Descripción textual de algunas entidades importantes del modelo de datos .....	50
3.2 Modelo de implementación .....	54
3.2.1 Diagrama de componentes.....	54
3.2.2 Diagrama de despliegue .....	56
3.3 Tratamiento de errores .....	57
3.4 Estrategias de codificación. Estándares y estilos a utilizar .....	57

<b>3.5 Seguridad informática .....</b>	<b>60</b>
<b>3.6 Análisis de resultados .....</b>	<b>61</b>
<b>3.6.1 Pruebas del Software.....</b>	<b>61</b>
<b>CONCLUSIONES.....</b>	<b>65</b>
<b>RECOMENDACIONES.....</b>	<b>66</b>
<b>REFERENCIAS BIBLIOGRÁFICAS.....</b>	<b>67</b>
<b>BIBLIOGRAFÍA.....</b>	<b>70</b>
<b>WEBGRAFÍA.....</b>	<b>70</b>
<b>GLOSARIO DE TÉRMINOS .....</b>	<b>73</b>

## INTRODUCCIÓN

Con el auge y desarrollo de las Tecnologías de la Información y las Comunicaciones (TIC) a nivel mundial se ha evidenciado un gran interés por vincularlas con las diferentes esferas y áreas del conocimiento de la sociedad. Una de las áreas que mayor desarrollo ha adquirido es la informática, casi la totalidad de las disciplinas científicas se valen de ella. La informática revolucionó la era de las comunicaciones, un fenómeno que ha dotado a la humanidad de medios importantes para la expansión del conocimiento, la cultura y la sabiduría.

El uso de la informática en la medicina es una de las aplicaciones más comunes e importantes desde hace varias décadas. La misma ha permitido al sector de la salud, no solo contar con métodos novedosos, sencillos 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). Previo al desarrollo de la informática en la salud, lo que se realizaba en los hospitales eran tareas administrativas, sistemas para admisión y altas, toma de órdenes, revisión de resultados y control de inventarios. Para tener una mejor organización de la información del paciente, se utiliza la historia clínica que es un documento legal que surge al documentar cada episodio clínico de un paciente.

Los profesionales de la salud recogen la información necesaria de los pacientes como: la causa por la que el paciente asiste a la consulta, resultados de exámenes, diagnósticos, tratamientos, seguimiento, entre otros, tanto escritas como gráficas, para la correcta atención y evolución clínica de un paciente a lo largo del proceso asistencial. Todo esto con el objetivo de mejorar la calidad y rapidez del servicio brindado en el sector de la salud. La historia clínica convencional o en papel, presenta algunas dificultades ya que la información se encuentra dispersa y en muchos casos duplicada, existe falta de información y en algunas ocasiones resulta ilegible (2). El volumen de papel que se acumula con el almacenamiento de dichas historias puede constituir una limitante para consultar toda la información escrita que existe, hacer comparaciones, llegar a conclusiones para las que se necesita analizar un gran número de ellas, protegerlas de deterioro, humedad y manipulación.

La Historia Clínica Electrónica (HCE) es un sistema de información, que contiene un conjunto de sistemas mantenidos electrónicamente, con los datos de salud de un paciente, guardados de tal manera que sirvan

para múltiples usuarios del registro médico. Esta permite crear, guardar, organizar y editar la información clínica de un paciente en una computadora, contribuyendo además a mejorar la calidad y seguridad en el cuidado de la salud. La adopción a nivel mundial de las HCE, ha demostrado beneficios que incluyen la disminución de errores en medicina, mejoras a nivel de costo-efectividad y posibilidad de brindar un rol activo a los pacientes en la toma de decisiones clínicas. Es el centro de cualquier sistema de información en salud. (3)

En HCE es de vital importancia la creación de un Repositorio de Datos Clínicos (RDC), que es una base de datos que contiene una variedad de fuentes clínicas y presenta una visión unificada de un solo paciente. Algunos de los datos que se encuentran a menudo en un RDC incluyen: resultados de las pruebas de laboratorio, datos demográficos del paciente, información de farmacia, informes de radiología e imágenes, informes de patología y la admisión hospitalaria.(3)

Los RDC se relacionan con el registro clínico, pero también están relacionados con el resto de los componentes a través de la Historia Clínica Electrónica. Todas estas ventajas de la HCE hacen que se cree como una meta, extender su uso a todos los centros de salud y brindar los servicios con mayor eficiencia, logrando así una mayor satisfacción de los pacientes.

Cuba forma parte de este mundo en constante evolución y no se ha mantenido ajena al desarrollo de las tecnologías y la informática. La Universidad de las Ciencias Informáticas (UCI), y específicamente el Centro de Informática Médica (CESIM), han jugado un papel importante en cuanto a la informatización de la salud, debido a que se han desarrollado diversas aplicaciones como apoyo al desarrollo de la atención sanitaria y tiene entre sus tareas prioritarias la creación de una HCE. El CESIM se encarga de desarrollar productos, servicios y soluciones informáticas para la optimización del trabajo y mejoramiento de la calidad de la atención médica. (4)

El Centro de Informática Médica cuenta con un conjunto de sistemas que almacenan información relativa a la atención sanitaria. Esta información se encuentra distribuida entre los sistemas especializados provocando que se incurra en errores como la falta de coordinación entre los niveles de atención y los lugares donde se ofrecen servicios, la duplicación de la información, así como incoherencias en el manejo de vocabularios clínicos y codificadores de terminología médica.

Una de las aplicaciones desarrolladas en este centro es el Sistema de Almacenamiento, Transmisión y Visualización de Imágenes Médicas (alás PACS) que contribuye al proceso de diagnóstico para los pacientes en el área de imagenología. Este sistema almacena todo lo relacionado a estudios clínicos del paciente como id de estudio, fecha de estudio, datos del médico que lo atendió, entre otros y guarda además datos personales del mismo.

El Sistema de Información Hospitalaria (alás HIS), es una institución médica donde se genera diariamente un gran volumen de información de todo tipo, tanto administrativa como asistencial. Con el objetivo de gestionar esta información y planificar la actividad hospitalaria surgen los HIS, permitiendo registrar toda la información clínica generada (5). El alás HIS contiene datos de todo lo relacionado con el paciente, como diagnóstico médico, antecedentes prenatales, desarrollo, alimentación, entre otros.

El Sistemas de Información Radiológica (alás RIS) gestiona la actividad clínica y administrativa del departamento de radiología, maneja la información demográfica de los pacientes, programan las citas y entregan reportes de diagnóstico. Reducen significativamente los errores asociados al registro manual de la información, aumentando de este modo el nivel de calidad en la atención al paciente y de seguridad de la información. Informatiza la gestión de la información referente a la actividad radiológica relativa a un paciente, desde la petición del estudio, hasta el informe del mismo pasando por la recogida de las incidencias y consumos que conlleva la realización de la exploración (6). El alás RIS almacena todo lo referente a datos personales, consultas y datos clínicos del paciente.

Los mecanismos de almacenamiento de estos sistemas provocan que la información se encuentre duplicada y que en algunos casos los datos estén dispersos, provocando problemas de accesibilidad a ellos. La falta de un almacenamiento único provoca la duplicidad, redundancia en las respuestas a las peticiones que se realizan, deficiencia al acceso de los datos y resulta engorrosa la coordinación entre los niveles de atención y los lugares donde se ofrecen servicios. La presencia de información incompleta y la no existencia de datos en un formato utilizable para las personas que trabajan con ellos, hacen ineficaz el trabajo de integración y colaboración entre estos sistemas y por tanto acarrea problemas en la gestión médica.

Los sistemas desarrollados en el CESIM, almacenan información relativa a los eventos de atención sanitaria de forma independiente, diseñan a su forma y estilo la HCE que van utilizar, sin tener en cuenta

mecanismos de interoperabilidad. La interoperabilidad es la capacidad de dos o más sistemas de intercambiar y utilizar información entre ellos (7).

Por lo antes expuesto se determina como **problema a resolver**: ¿Cómo centralizar la información clínica gestionada por los sistemas que conforman la Historia Clínica Electrónica en el Centro de Informática Médica?

Para la investigación se define como **objeto de estudio** los sistemas de almacenamiento de información clínica y el **campo de acción** queda determinado por los Repositorios de Datos Clínicos de las Historias Clínicas Electrónicas.

Para dar respuesta a la problemática anterior se propone como **objetivo general**: Desarrollar un Repositorio de Datos Clínicos que proporcione el almacenamiento centralizado de la información clínica gestionada por los sistemas que conforman la Historia Clínica Electrónica del Centro de Informática Médica.

Se definen como **tareas de la investigación**:

1. Realizar el estudio del estado del arte del tema tratado.
  - 1.1 Evaluar las tendencias y mejores prácticas en cuanto al desarrollo de repositorios de información.
  - 1.2 Estudio de los estándares relacionados con la implementación de RDC.
  - 1.3 Valorar la posible reutilización de componentes ya existentes que puedan enriquecer la solución propuesta.
  - 1.4 Analizar la información clínica que almacenan los sistemas desarrollados en el CESIM así como la estructura de la misma.
  - 1.5 Estudiar la arquitectura propuesta por el CESIM para el desarrollo de sistemas con tecnologías Java.
2. Implementar el componente Repositorio de Datos Clínicos para el CESIM.
  - 2.1. Desarrollar mediante la metodología definida en el CESIM, los artefactos generados en los flujos de trabajo de “Modelado de Negocio”, “Administración de Requisitos”, “Análisis y Diseño” e “Implementación”.

**Este documento está estructurado con 3 capítulos:**

**Capítulo 1: Fundamentación teórica de Repositorio de Datos Clínicos:** En este capítulo se realiza un estudio profundo del estado del arte, se exponen los estándares utilizados y las herramientas para la modelación del análisis y diseño del repositorio y la implementación del mismo.

**Capítulo 2: Ingeniería del sistema:** En este capítulo se realiza una descripción de la propuesta del sistema y sus principales funcionalidades, así como las características del sistema, análisis y diseño. Describe la construcción del componente, a través del diseño, enfocado a cómo el sistema cumple sus objetivos teniendo en cuenta los requisitos funcionales y no funcionales. Se realizan los casos de uso del sistema, se detalla la modelación de los diagramas de clases del diseño y diagramas de secuencia del diseño. Se describe la construcción además el modelo de dominio y la arquitectura del software que se utilizará.

**Capítulo 3: Implementación y prueba:** En este capítulo se realizan diagramas de despliegue, colaboración y modelo de datos. Se diseñan y ejecutan los casos de pruebas para probar el correcto funcionamiento del sistema implementado.

## CAPÍTULO 1: FUDAMENTACIÓN TEÓRICA DE REPOSITORIO DE DATOS CLÍNICOS

En este capítulo se expone la base teórica para el posterior desarrollo de un repositorio de datos clínicos. Se abordará sobre algunos sistemas que utilicen un repositorio de datos clínicos, así como algunos estándares relacionados con la implementación de dicho repositorio. Se analizará la información clínica que almacenan los sistemas desarrollados en el CESIM, así como la estructura de la misma y la arquitectura propuesta por el centro para el desarrollo de sistemas con tecnologías Java.

### 1.1 Conceptos fundamentales

**Clínico:** disciplina de la medicina que se ocupa de la observación directa del paciente y de su posterior tratamiento en caso de ser necesario. A partir de la serie de datos e informaciones que el médico clínico va recolectando a través de la entrevista al paciente, acompañado todo esto por los signos que resultan de la exploración física o de exploraciones complementarias de laboratorio o pruebas de imagen.(8)

**Información clínica:** informe detallado del diagnóstico, tratamiento y seguimiento de un paciente en particular. La información clínica también contiene alguna información demográfica acerca del paciente (por ejemplo, sexo, edad).(9)

**Repositorio:** depósito o archivo donde se almacena de forma centralizada información digital, es una base de datos fundamental para el diseño; no solo guarda datos, sino también algoritmos de diseño y, en general, elementos software necesarios para el trabajo de programación.(10)

**Repositorio de datos clínicos:** es una base de datos, comúnmente actualizada en tiempo real, que recibe información clínica de varias fuentes y la guarda de forma unificada y estandarizada.(11)

**Historia Clínica Electrónica:** un sistema o conjunto de sistemas que posibilitan la creación y almacenamiento digital de los episodios médicos del paciente.(11)

### 1.2 Análisis de las soluciones existentes a nivel internacional

#### Repositorio Clínico Hitachy

Hitachy Data Systems Corporation, es una corporación desarrollada el 22 de febrero de 2011 en Japón, propiedad de Hitachy, que ha presentado una nueva solución para proveedores de servicios de salud que

mejora la toma de decisiones clínicas y la atención al paciente. Hitachy Clinical Repository (HCR) es una solución de gestión de la información que transforma los datos médicos y las imágenes en bruto, en información interpretable, independientemente de las aplicaciones de donde proceda.(12)

Mediante la utilización de las tecnologías líderes de gestión de datos y almacenamiento de la compañía, HCR crea un repositorio de metadatos basado en cualquier sistema de información y proporciona a los proveedores de servicios de salud una visión consolidada de toda la información relevante de los pacientes.(12)

Hitachi Clinical Repository proporciona la flexibilidad de integrar fácilmente cualquier tipo de sistema clínico en un archivo para toda la organización, creando la base para ofrecer información en todos los usos clínicos y administrativos, y proporcionar así una asistencia mucho mejor al paciente. El repositorio está formado por varias tecnologías de almacenamiento propios de Hitachi, que incluyen:

- La Hitachi Content Platform (HCP): es una solución de almacenamiento de objetos que permite a las organizaciones de las tecnologías de la información y proveedores de servicios almacenar, compartir, sincronizar, proteger , preservar, analizar y recuperar datos de archivos de un solo sistema. Automatiza las operaciones del día a día, tales como la protección de datos, evoluciona rápidamente a los cambios en las tecnologías, el alcance, las aplicaciones, almacenamiento y el servidor durante la vida útil de los datos.
- La Hitachi Data Discovery Suite (HDDS): ayuda a buscar, identificar, recopilar y conservar los datos en toda la infraestructura de almacenamiento de un lugar seguro para la mejora de la gestión de almacenamiento. Permite a los usuarios finales y al personal de las tecnologías de la información para llevar a cabo: búsqueda multiplataforma, recuperación y jerarquización basada en archivos con almacenamiento por niveles existentes.(12)

### **Estación de Trabajo Clínico Regional (CWS) y Repositorio de Datos Clínicos (RCD)**

Es un proyecto desarrollado en julio del 2009 en Nueva Zelanda que exige la aplicación de una plataforma común para los sistemas clínicos a través de cada región. Para ello es fundamental una estación de trabajo clínico (CWS), y un repositorio centralizado de datos (RDC).

El sistema contiene un conjunto de información clínica y datos clínicos adicionales que pueda determinar cada región. Los RDC regionales, sientan las bases para una visión nacional de la salud de un paciente y permitirle la accesibilidad de cualquier paciente fuera de la región. También proporciona vistas nacionales de datos clínicos, los cuales serán utilizados con fines de gestión de los conocimientos, la investigación y la planificación de la salud. Todos los médicos de los hospitales, y proveedores de la comunidad, con el acceso al repositorio, verá la información del paciente (de fuentes de atención secundaria y terciaria). (13)

### **Repositorio de Datos Clínicos del Departamento de Defensa para asuntos de salud de Veteranos**

El Departamento de Defensa de Estados Unidos desarrollado en junio de 2008 cuenta con un Repositorio de datos Clínicos para asuntos de salud de Veteranos, el cual permite al Almacén de Datos Clínicos del Departamento de Defensa y al Almacén de Datos de Salud del Departamento de Asuntos para Veteranos, intercambiar datos de farmacias del enfermo ambulatorio e información de alergia, para lograr la meta de la interoperabilidad calculable entre las agencias. Proporciona una aplicación basada en Web que permite a los usuarios autorizados acceder para marcar o desmarcar los pacientes, lograr intercambio de datos y la disponibilidad de los sistemas del monitor y las estadísticas de rendimiento. (14)

#### Características

- Proporciona los datos para el intercambio bidireccional, calculable y permanente entre los dos almacenes de datos clínicos.
- Habilita el intercambio de cheques de órdenes de apoyo entre el Departamento de Defensa, las prescripciones del Departamento de Asuntos para Veterano y los datos de alergia y drogas.
- Proporciona una aplicación web que le permite a los usuarios autorizados ver sus datos compartidos y supervisa la disponibilidad del sistema y la estadística del mismo.
- Permite acceder a los datos de farmacia y a los datos de alergia.

### **Repositorio de Datos Clínicos de la Universidad de Medicina en Virginia**

Es un almacén de datos manejados por la División de Informática Clínica del Departamento de Ciencias de Salud Pública, que contiene la información sobre los pacientes en la Universidad de Medicina de Virginia en Estados Unidos, proporcionando el acceso directo a investigadores, médicos, y estudiantes, de

los datos clínicos de los pacientes. El RDC contiene por más de 15 años los datos de un mismo paciente actualizado y aproximadamente contiene, 1000,000 de pacientes. Los datos de los pacientes incluyen diagnósticos, resultados del laboratorio, procedimientos, los detalles de visita de enfermo ambulatorio, las medicaciones, entre otras.(15)

Una de sus principales funciones del repositorio es guardar poblaciones en el tiempo y compartirlas con otros usuarios autorizados, recuperar los datos experimentales de las solicitudes de subvención y los análisis exploratorios, identificar oportunidades de mejora de la calidad y explorar experiencia en la universidad para proporcionar información para enseñar a los estudiantes. Desde cualquier navegador web estándar se logra el acceso al repositorio.(15)

Las soluciones analizadas brindan gran fiabilidad y seguridad de la información con que interactúan, proporcionando el acceso directo de sistemas autorizados con necesidad de acceder a la misma. Utilizan tecnologías de almacenamientos propias de cada uno de ellos, almacenando de forma centralizada toda la información sanitaria con que ellos se relacionan. Todas las soluciones brindan una interfaz de comunicación para el trabajo con el repositorio, a la cual se puede acceder mediante cualquier navegador web estándar.

Estos sistemas no pueden ser usados ya que no utilizan estándares validados por ninguna asociación internacional de salud, debido a que la información que manejan están normadas por patrones específicos, de cada uno de ellos en particular. Además la información que se almacena en estas soluciones no está al alcance y acceso de todos los usuarios.

### **1.3 Análisis de las soluciones existentes a nivel nacional**

#### **Hospital Clínico Quirúrgico Hermanos Ameijeiras**

Institución inaugurada el 3 de diciembre de 1982, por el Comandante en jefe Fidel Castro. Creada por la Revolución para brindar al pueblo una atención en el ámbito de los mejores centros de su clase en el mundo. Se caracteriza por un avance ascendente en la incorporación de las tecnologías más modernas y un espíritu creador por parte de ingenieros y técnicos junto a los cuadros científicos para la solución de los problemas tecnológicos más complejos(17).

El objetivo principal del hospital es obtener una atención médica de excelencia y enfrentar la introducción y asimilación de la tecnología más avanzada ha sido la divisa principal de la institución. A partir de enero del año 2000 el hospital "Hermanos Ameijeiras" sustituyó la hoja de egreso por el informe de alta y lo incluyó en el proyecto de automatización del departamento de registros médicos del hospital. Los datos de paciente se incluyen en un repositorio de datos clínicos de la forma siguiente: El número de historia clínica, los datos generales del paciente y la información sobre los servicios en que fue hospitalizado el paciente (17).

Es importante destacar que este mecanismo de almacenamiento que se ha realizado en el Hospital Clínico Quirúrgico Hermanos Ameijeiras aún no arroja un resultado concreto, pero constituyen los primeros pasos para un almacenamiento seguro y centralizado de la información clínica de los pacientes, aunque no puede ser utilizado ya que no cuenta con una estandarización adecuada.

En la Universidad de las Ciencias Informáticas se desarrollan un conjunto de sistemas, se muestran algunos de ellos:

### **alas PACS**

En el Centro de Informática Médica se desarrolló el alas PACS que contribuye al proceso de diagnóstico para los pacientes en el área de imagenología. Este producto está conformado por equipos de adquisición de imágenes, estaciones de visualización, servidores de almacenamiento y la infraestructura de red para comunicar las distintas partes.

Su objetivo principal es permitir el funcionamiento del servicio de imágenes sin la necesidad de la impresión de placas radiológicas ni de papeles para la información clínica asociada a las imágenes, supliendo los altos costos en tiempo y dinero que esto significa. Las imágenes puedan ser vistas desde cualquier estación de trabajo del hospital para su diagnóstico, ya sea individual o simultáneamente, gracias a que son almacenadas en un servidor central.(16)

### **alas RIS**

En este mismo centro se desarrolla el alas RIS que comprende los procesos administrativos del departamento de radiología, incluyendo gestión de citas, admisión y alta de pacientes, reportes de

diagnósticos, ubicación y seguimiento de las placas radiográficas e información general de los pacientes de radiología.

### **alas HIS**

El alas HIS está orientado a satisfacer las necesidades de la gestión de la información en los centros de atención hospitalaria. Permite registrar toda la información clínica generada, además que circule de manera ordenada a través de la red y que la documentación asociada sea almacenada en servidores de gran capacidad, evitando así el deterioro de las mismas.

El sistema utiliza una HCE única, que cuenta con una biblioteca de clases que respalda la creación, lectura, validación y persistencia de documentos CDA v2 y de mensajes HL7 v2.3.1 para el intercambio de estos. Por cada documento que se genere hay una plantilla que se guarda como archivo CDA. Dispone de la posibilidad de realizar firmas digitales a través de un componente desarrollado en el marco del Visor de Historias Clínicas.

Cada uno de estos sistemas cuenta con servidores, los cuales presentan todas las funcionalidades necesarias para almacenar información específica de cada uno de ellos. Pero ninguno de estos servidores almacena de forma centralizada, accesible y completa toda la información que generan dichos sistemas para que pueda ser compartida con otros.

Todos estos sistemas tienen una funcionalidad diferente pero la finalidad de todos ellos se reduce a: mejorar la atención al paciente. Con el fin de almacenar toda la información que generan estos sistemas en el repositorio de datos clínicos, se realizó un estudio a los datos que guardan cada uno de ellos en particular. Luego del estudio realizado se determinó almacenar toda la información referente a los pacientes y sus eventos clínicos.

El alas PACS fue uno de los sistemas analizados, donde se recopiló todo lo referente a los datos de estudio del paciente como id de estudio, si es contrastado o no, fecha del estudio, entre otros e información demográfica del paciente incluidas también en el sistema alas HIS. Del sistema alas HIS se seleccionaron todo tipo de información clínica del paciente como antecedentes prenatales, gineco obstétricos, hemato oncológico y sexuales reproductivos del mismo, además datos del médico que atendió

al paciente, todo lo relacionado a el desarrollo, período neonatal, hoja frontal del paciente, entre otros. Del alas RIS se recopilaron datos referente a los diferentes tipos de citas a las que han asistido los paciente como sencillas, interconsultas y de referencia, datos personales del paciente como dirección, religión, talla, número de historia clínica, entre otros. Todos estos datos que se seleccionaron son con los que contará el repositorio de datos clínicos.

### **1.4 Estándares relacionados con la solución**

La interoperabilidad es la habilidad de los sistemas de información computarizados y las aplicaciones de software, de comunicarse intercambiando datos en forma precisa, efectiva, consistente en tiempo real y de usar esa información intercambiada. Para lograr la interoperabilidad entre los sistemas que gestionan información clínica del paciente dentro de una institución sanitaria es necesaria la adopción de estándares.(7)

La estandarización garantiza la validez de los datos clínicos en el sistema informático y la creación de un sistema universal, accesible, fácil de usar y en el que se puede confiar. Un estándar es un conjunto de reglas y definiciones que sirven como modelo, guía, y son aplicadas a un dominio específico. Los estándares sirven como normas para las interfaces de comunicación contribuyendo de forma fiable y aprobada, la conexión de los componentes sin los cuales los sistemas no podrán funcionar correctamente. (18)

### **Modelo de Arquetipos de OpenEHR**

Es una especificación arquitectónica abierta y estándar para la implementación de la HCE. Los datos de las atenciones médicas son almacenados en un repositorio y están accesibles a los usuarios autorizados. La especificación de OpenEHR alcanza un nivel de desacoplamiento total, lo cual la hace muy flexible, pues los datos clínicos son guardados como tal, especificados a través de un modelo de arquetipos.

El modelo de información de OpenEHR es una especificación orientada a objetos, y es la parte del estándar que debe implementarse dentro del software. La principal característica es que es un modelo genérico, es decir, que permite representar toda la información que se genera en la asistencia médica, pero no especifica la semántica de cada concepto clínico particular, solo contiene conceptos clínicos generales, como son: observaciones, evaluaciones, instrucciones y acciones.(18)

### HL7

HL7 (Health Level Seven) es una organización con base en los Estados Unidos, dedicada al desarrollo de un conjunto de estándares para facilitar el intercambio electrónico de información clínica, que utiliza una notación formal de modelado, el Lenguaje Unificado de Modelado (UML) y un metalenguaje extensible de marcado con etiquetas (XML). Esta organización está acreditada por el American National Standards Institute, autoridad oficial de estandarización americana (ANSI). (19)

HL7 desarrolla estándares cuya sintaxis coherente y extensible permite estructurar información en salud. También permite apoyar los procesos de atención al paciente, para ser intercambiada entre aplicaciones de software, conservando al mismo tiempo la semántica de la información. (19)

Desde 1997, HL7 está desarrollando un nuevo conjunto de estándares, conocido como “versión 3”, o “v3”. Estos estándares están dirigidos principalmente a la definición de mensajes de aplicación, pero están basados en modelos formales, incluyendo el “Modelo de Información de Referencia” RIM.

Estándares de HL7:(19)

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

- CCOW: Es un estándar framework para compartir contexto entre aplicaciones.

### **CDA**

Arquitectura de Documento Clínico (CDA, por sus siglas en inglés), es un subdominio del HL7, basado en XML que describe la estructura y semántica de documentos clínicos con el objetivo de facilitar su intercambio en un entorno de interoperabilidad. Permite una implementación efectiva y eficiente, siendo independiente de plataforma. Soporta especialmente el intercambio de documentos legibles entre sistemas, permitiendo presentar la información de forma adecuada a usuarios con diferentes requisitos o conocimientos.(20)

Los estándares relacionados con la solución son: OpenEHR ya que es un modelo para el almacenamiento de datos clínicos y HL7 que cuenta con un conjunto de estándares que se utilizan para el intercambio electrónico de información clínica, además de ser un estándar reconocido a nivel internacional. Estos estándares son libres de ataduras a un lenguaje específico o plataforma de desarrollo, los cuales pueden ser implantados con la introducción de herramientas libres, permitiendo de esta forma la facilidad de mantenimiento y actualización, así como el evitar la inversión de grandes sumas de dinero en licencias.

Existen muchas personas que se han documentado acerca de estos estándares, lo que trae como beneficio el conocimiento y algunas experiencias acerca de ellos, además de mucha información significativa para la utilización de los mismos. Se utilizará el estándar HL7 v2.3.1 ya que es el servicio de mensajería que permite enviar y recibir mensajes HL7 en dicha versión, para el intercambio electrónico de datos de salud basado en XML, para garantizar la seguridad en cuanto a la comunicación de los sistemas que interactúan con el repositorio, a través de dichos mensajes.

### **1.5 Descripción de tecnologías, herramientas y metodología propuestas para el proceso de desarrollo**

#### **Proceso Unificado de Desarrollo de Software (RUP)**

Es un marco de desarrollo de software iterativo e incremental, que está compuesto por cuatro fases denominadas Inicio, Elaboración, Construcción y Transición. Estas iteraciones ofrecen como resultado un

incremento del producto desarrollado que añade o mejora las funcionalidades del sistema en desarrollo. Es un marco de trabajo extensible que puede ser adaptado a organizaciones o proyectos específicos.(21)

### **Características de RUP**

- Dirigido por los casos de uso: En el Proceso Unificado los casos de uso se utilizan para capturar los requisitos funcionales y para definir los contenidos de las iteraciones. La idea es que cada iteración coja un conjunto de casos de uso o escenarios y desarrolle todo el camino a través de los distintos flujos de trabajo: diseño, implementación, prueba, etc.
- Centrado en la arquitectura: El Proceso Unificado asume que no existe un modelo único que cubra todos los aspectos del sistema. Existen múltiples modelos y vistas que definen la arquitectura de software de un sistema. La analogía con la construcción es clara, cuando construyes un edificio existen diversos planos que incluyen los distintos servicios del mismo: electricidad, fontanería, etc.
- Iterativo e incremental: RUP divide el proceso en cuatro fases, dentro de las cuales se realizan varias iteraciones en número variable según el proyecto, y se definen según el nivel de madurez que alcanzan los productos que se van obteniendo con cada actividad ejecutada. La terminación de cada fase ocurre en el hito correspondiente a cada una, donde se evalúa que se hayan cumplido los objetivos de la fase en cuestión.(21)

### **Lenguaje Unificado de Modelado 2.1 (UML)**

UML es un lenguaje para el modelado orientado a objetos. Se usa para definir un sistema de software; para detallar los artefactos en el sistema; para documentar y construir. Se puede utilizar en una gran variedad de formas para soportar una metodología de desarrollo de software (como RUP) pero no especifica en sí mismo qué metodología o proceso usar.

Las principales funciones de UML son:

- Visualizar: UML permite expresar de una forma gráfica un sistema de forma que otro lo puede entender.
- Especificar: UML permite especificar cuáles son las características de un sistema antes de su construcción.

- Construir: A partir de los modelos especificados se pueden construir los sistemas diseñados.
- Documentar: Los propios elementos gráficos sirven como documentación del sistema desarrollado, sirviendo para su futura revisión.(22)

### **Enterprise Architect 7.5**

Es una herramienta multi-usuarios, diseñada para ayudar a construir software robusto y fácil de mantener. Además, ofrece salida de documentación flexible y de alta calidad. Su interfaz es intuitiva y de alto rendimiento, con una vista del proyecto como un "explorador", además posee un entorno fácil de usar, rápido y flexible.

Las bases de Enterprise Architect usan Perfiles UML para extender el dominio de modelado, mientras que la validación del modelo asegura integridad. Combina los procesos de negocio, información y flujos de trabajo en un modelo usando extensiones gratuitas para la notación de modelado de procesos del negocio.(23)

### **Java**

Lenguaje de programación multiplataforma que adopta el paradigma orientado a objetos, diseñado por la compañía Sun Microsystems Inc., salió a la luz en 1991, Licencia de software: GNU GPL / Java Community Process. Es un lenguaje de programación basado en clases, que fue diseñado específicamente para tener tan pocas dependencias de implementación como fuera posible.

Su intención es permitir que los desarrolladores de aplicaciones escriban el programa una vez y lo ejecuten en cualquier dispositivo, lo que quiere decir que el código que es ejecutado en una plataforma no tiene que ser recompilado para correr en otra. Java es, a partir del 2012, uno de los lenguajes de programación más populares en uso, particularmente para aplicaciones de cliente-servidor de web, con unos 10 millones de usuarios reportados. (24)

### **Eclipse 3.1**

Es un Entorno de Desarrollo Integrado (IDE) abierto y extensible. Un IDE es un programa compuesto por un conjunto de herramientas útiles para un desarrollador de software. Como elementos básicos, un IDE cuenta con en un editor de código, un compilador/intérprete y un depurador. Eclipse sirve como IDE Java y cuenta con numerosas herramientas de desarrollo de software. A la plataforma base de Eclipse se le

pueden añadir extensiones (plugins) para extender la funcionalidad. Tiene características potentes como el completamiento de código, permite mostrar desde varias perspectivas el entorno de cada framework permitiendo la integración de sus funcionalidades y el trabajo en equipo mediante el Subclipse para el manejo de versiones.(25)

### **PostgreSQL 9.1**

Es un sistema de gestión de base de datos relacional orientada a objetos. Es un programa de código abierto, que está dirigido por una comunidad de desarrolladores llamada PGDG (*PostgreSQL Global Development Group*). Entre sus principales características se encuentran alta concurrencia, la amplia variedad de tipos nativos, y diversas funciones más específicas.(26)

Entre las principales ventajas de usar PostgreSQL se tiene:

- Completo soporte para transacciones.
- Soporte para construcciones SQL.
- Orientación a objetos con herencia de tablas.
- Herramientas gráficas de diseño y administración de bases de datos: Existen varias herramientas gráficas de alta calidad para administrar las bases de datos (pgAdmin, pgAccess) y para hacer diseño de bases de datos (Tora, Data Architect).
- Multiplataforma: PostgreSQL está disponible en casi cualquier Unix y en Windows.
- Extensible: El código fuente está disponible para todos sin costo. Si se necesita extender o personalizar PostgreSQL de alguna manera, puede hacerse con un mínimo esfuerzo.
- Replicación (soluciones comerciales y no comerciales) que permiten la duplicación de bases de datos maestras en múltiples sitios de réplica. (26)

### **JBoss 4.2.2**

JBoss Application es el servidor de aplicaciones de código abierto más ampliamente desarrollado del mercado. Por ser una plataforma certificada Java EE, soporta todas las funcionalidades de Java EE 1.4, incluyendo servicios adicionales como clustering, caching y persistencia. JBoss es ideal para aplicaciones Java y aplicaciones basadas en la web. (27)

Entre sus innumerables ventajas, JBoss permite:

- Disminuir la complejidad de desarrollo.
- Mejorar la experiencia del usuario final.
- Solucionar la fricción en los procesos comerciales.

### **Bibliotecas y marcos de trabajo**

#### **Java Server Faces 1.2 (JSF)**

Es un framework de interfaz de componentes de usuarios del lado del servidor para las aplicaciones web basadas en la tecnología Java.(28)

Los principales componentes de la tecnología JSF son:

- Una API para representar componentes de Interfaz de Usuario y gestionar su estado.
- Manejar eventos, validar en el servidor y conversión de datos.
- Definir la navegación de páginas.
- Soporte de internacionalización y accesibilidad.
- Dos librerías de etiquetas Java Server Pages (JSP) personalizadas para expresar componentes en una página JSP y enlazar los componentes a objetos del servidor.(28)

#### **Enterprise JavaBeans 3.0 (EJB)**

Los EJB son una de las API que forman parte del estándar de construcción de aplicaciones empresariales J2EE. Los EJB proporcionan un modelo de componentes distribuido estándar del lado del servidor. El objetivo de los EJB es dotar al programador de un modelo que le permita abstraerse de los problemas generales de una aplicación empresarial (conurrencia, transacciones, persistencia, seguridad, etc.) para centrarse en el desarrollo de la lógica de negocio en sí. El hecho de estar basado en componentes permite que éstos sean flexibles y sobre todo reutilizables. (29)

#### **JBoss Seam 2.1.1**

Seam es un framework de código abierto desarrollado por la empresa Jboss, con el fin de unir diferentes tecnologías y estándares de Java en un solo framework, a la vez que añade algunas funcionalidades no

contempladas por ellos. Con Seam basta agregar anotaciones propias de éste a los objetos entidad y sesión de EJB, logrando con esto escribir menos código Java y XML.

Otra característica importante es que se pueden hacer validaciones en los POJOs (Plain Object Java) y además manejar directamente la lógica de la aplicación y de negocios desde las sesiones beans. Seam también se integra perfectamente con otros frameworks como: RichFaces, ICEFaces (soportan Ajax) MyFaces, Hibernate y Spring. Además integra las tecnologías como JavaScript Asíncrono, JSF, JPA y Business Process Management (BPM). (30)

### **Rich Faces 3.3.1**

Es un framework de código abierto que añade capacidad Ajax dentro de aplicaciones JSF existentes sin recurrir a JavaScript. Rich Faces incluye ciclo de vida, validaciones, conversores y la gestión de recursos estáticos y dinámicos. Los componentes de Rich Faces están construidos con soporte Ajax y un alto grado de personalización del “look-and-feel” que puede ser fácilmente incorporado dentro de las aplicaciones JSF.(31)

Algunas ventajas que aporta la utilización de RichFaces son:

- Al pertenecer RichFaces a un subproyecto de JBoss, su integración con Seam es perfecta.

### **Hibernate 3.3**

Es una herramienta de mapeo objeto/relacional y un generador de sentencias SQL. Permite diseñar objetos persistentes que podrán incluir polimorfismo, relaciones, colecciones, y un gran número de tipos de datos. De una manera muy rápida y optimizada se pueden generar bases de datos en cualquiera de los entornos soportados. (32)

Entre las principales ventajas que se pueden encontrar en Hibernate están:

- Productividad: evita código confuso en la capa de persistencia, lo que permite centrarse en la lógica de negocio.
- Mantenibilidad: por tener pocas líneas de código permite que el código sea más claro. Al dividir la capa de persistencia se pueden identificar los errores muy fácilmente.
- Rendimiento: Hibernate tiene un buen desempeño pero todo depende realmente de como se realicen las consultas y como se configure el Framework.(32)

### **Java Persistence API (JPA)**

Es el estándar para la gestión de la persistencia y provee facilidades para el mapeo objeto / relacional a desarrolladores de aplicaciones, haciendo uso del modelo de dominio de Java para administrar bases de datos relacionales. (33)

### **Ajax4jsf**

Es una librería que se integra totalmente en la arquitectura de JSF y extiende la funcionalidad de sus etiquetas dotándolas con tecnología Ajax de forma limpia y sin añadir código Java script. Mediante este framework es posible variar el ciclo de vida de una petición JSF, recargar determinados componentes de la página sin necesidad de recargarla por completo, realizar peticiones automáticas al servidor, control de cualquier evento de usuario, etc. Permite dotar a las aplicaciones JSF de contenido mucho más profesional con muy poco esfuerzo.(34)

En este capítulo se realizó un estudio de los principales sistemas informáticos que cuentan con un repositorio de datos clínicos a nivel nacional e internacional, llegando a la conclusión que los mismos no dan solución a la problemática planteada. Se determinaron las herramientas necesarias y los estándares requeridos para el desarrollo de la solución propuesta.

## **CAPÍTULO 2: INGENIERÍA DEL SISTEMA**

En el presente capítulo se describe la problemática existente, así como las características del componente a desarrollar. Debido a la poca definición de los procesos del negocio y la ausencia de un cliente se manifiestan los principales conceptos tratados y las relaciones existentes entre ellos mediante un Modelo de Dominio o Modelo Conceptual. Se definen los requisitos funcionales y no funcionales del sistema, los casos de uso del sistema y sus actores y la arquitectura propuesta. Se detalla la modelación de los diagramas de clases del diseño y diagramas de secuencia del diseño.

### **2.1 Propuesta de sistema**

Se propone realizar un Repositorio de Datos Clínicos para almacenar los datos relacionados con el paciente y los estudios o procedimientos que se le hayan realizado al mismo. Con este repositorio van a interactuar sistemas externos, por ejemplo: alas PACS, alas RIS y alas HIS. Estos sistemas para comunicarse con el repositorio, enviarán un mensaje, ya sea para almacenar o recuperar datos del paciente. En consecuencia recibirán otro con la información solicitada o con una confirmación de que el mensaje se ha recibido correctamente y que el almacenamiento se ha realizado sin contratiempos. Para el envío y recepción de los mensajes HL7 el repositorio utilizará el servicio de mensajería HL7, desarrollado en el departamento de Tecnologías, Integración y Estándares del CESIM en el año 2011, que es capaz de enviar y recibir mensajes HL7 en la versión 2.3.1 de dicho estándar. Dichos mensajes van a estar estructurados por un conjunto de segmentos que a su vez van a estar conformados por campos, en los cuales estarán los datos clínicos que el sistema externo desea enviar o recibir. Estos mensajes también serán almacenados íntegramente en el RDC una vez que sean recibidos.

El repositorio permite al administrador seleccionar un conjunto de datos a cifrar de forma tal que esta información se mantenga siempre oculta ante el acceso de cualquier intruso. La información solo se descifrará en el momento de enviársela a un sistema externo y nunca persistirá descifrada, ni siquiera en el proceso de conformación del mensaje. Una vez que esta información sea enviada al sistema externo, ya pasa a ser responsabilidad de este la protección de la misma.

El repositorio permite en todo momento generar archivos log, para tener un registro de todas las operaciones y errores que ocurran en el mismo, así como las evidencias de los sistemas que accedieron

al repositorio ya sea para almacenar o solicitar información del mismo. De esta forma se puede tener un control de todo lo que esté sucediendo dentro del repositorio, así como conocer qué sistema tuvo acceso a determinada información y cuándo. También el repositorio realizará distintas copias de seguridad, ya sea para una base de datos, un backup o un archivo, y así cuando el repositorio sufra de alguna pérdida de datos, contará con su propio mecanismo de recuperación.

El desarrollo de un RCD centralizado, estandarizado, con información fiable, accesible, completa y en un formato utilizable traerá muchos beneficios para el Centro de Informática Médica, como es la creación de una HCE, también le permitirá a los sistemas del centro un repositorio único.

Se seleccionaron los mensajes que almacenan la misma información que el Repositorio de Datos Clínicos utiliza. Algunos datos que el repositorio debe guardar no estaban contenidos en ningún mensaje, por lo que se crearon segmentos regidos por el estándar HL7 en su versión 2.3.1 que pudieran almacenar dichos datos. Los mensajes seleccionados son:

**ADT\_A01:** Insertar información del paciente.

Este mensaje es utilizado para insertar información de los pacientes, además de insertar todos los datos personales del mismo, se insertan datos del hospital de servicio, sobre la ubicación del hospital, desarrollo del paciente, período neonatal, los antecedentes prenatales, sexuales reproductivos, gineco obstétricos y hemato oncológico de los pacientes, entre otros. Algunos segmentos que se añadieron al mensaje son el ZPA que se utiliza para insertar datos adicionales del paciente referente a la talla, el peso, número de la historia clínica, etc. El segmento ZHF es para insertar información de la hoja frontal de los pacientes, el ZDM para insertar datos del diagnóstico médico de los pacientes, entre otros.

**ADT\_A08:** Actualización de información del paciente.

Este mensaje es utilizado para actualizar información de los pacientes, contiene los mismos datos y segmentos adicionales que el mensaje insertar información del paciente.

**ADT\_A28:** Insertar información de persona.

Este mensaje es utilizado para insertar información de las personas como: el identificador de la persona, el nombre, apellidos, fecha de nacimiento, dirección, país, número de teléfono, lugar de nacimiento, nacionalidad, entre otros datos. Los segmentos adicionales del mensaje son el ZFT para insertar la foto de

la persona, ZDP es el responsable de insertar datos de la dirección y el ZME es el segmento encargado de adicionar datos del médico.

**ADT\_ A31:** Actualización de información de persona.

Este mensaje es utilizado para actualizar información de las personas, contiene los mismos datos y segmentos adicionales que el mensaje insertar información de persona.

**CRM\_ C01:** Registro de Estudios Clínicos del paciente.

Este mensaje es utilizado para guardar registros de estudios clínicos del paciente que contiene datos como: fecha del estudio, observaciones, nombre de la institución que realizó el estudio, entre otros. Este mensaje contiene un único segmento adicional, el ZRC que es para insertar información adicional de los registros clínicos.

**RCI\_I05:** Solicitud de información clínica del paciente.

El mensaje permite guardar información de las citas de los pacientes, el mismo contiene todo lo relacionado a la cita, así como las observaciones de dicha cita, qué médico la realizó, el diagnóstico médico, el lugar en que se realizó la cita, entre otros datos. Este mensaje también se envía para recuperar todo tipo de información clínica de los pacientes excepto los datos de estudio. Los segmentos adicionales son ZCR, ZCI y ZCS que se utilizan para insertar información de las citas de referencias, las citas de interconsultas y las sencillas respectivamente. El mensaje cuando es recibido siempre contiene el identificador universal del paciente (uid), y el repositorio conforma un mensaje ADT\_A01 para enviarle al sistema externo la información del paciente deseado. Los segmentos adicionales son campos opcionales dentro del mensaje, si el mismo llega con el id de las citas, el repositorio envía el mismo mensaje con la información de las citas que el sistema externo solicita. Cuando el mensaje viene con el id de las citas y también con información adicional de las mismas, el repositorio guarda la información de las citas correspondiente al (uid) del paciente recibido.

**ZDE** Recuperar Datos de Estudio.

Al no existir un mensaje que permita recuperar datos de estudio de los pacientes, se creó el ZDE regidos por el estándar HL7 en su versión 2.3.1, con un único segmento adicional ZRC que contiene datos de estudio del paciente que el sistema externo solicita. El mensaje es enviado con un id de estudio y el repositorio conforma y envía un CRM\_ C01 con los datos de estudios pedidos.

## 2.2 Servicio de Mensajería HL7

El servicio de mensajería HL7 que se utilizará para la comunicación entre las aplicaciones y el Repositorio de Datos Clínicos, permite recibir y enviar mensajes HL7, a través de estos mensajes es que los sistemas que interactúen con el repositorio pueden enviar o solicitar información al mismo. Las solicitudes pueden ser para recuperar datos referentes a los pacientes o estudios y para almacenarlos. Cuando el repositorio recibe un mensaje, envía otro de aceptación al sistema que los envió y cuando ocurre algún error para satisfacer esta solicitud, envía un mensaje de rechazo. El servicio de mensajería HL7 se desarrolló con el objetivo de alcanzar una comunicación eficiente y segura en los sistemas que coexisten en un entorno hospitalario y así lograr una mejor atención al paciente.

El servicio de mensajería HL7 permite establecer comunicación con otros sistemas y lo hace a través del protocolo MLLP (Minimal Lower Layer Protocol). El objetivo del protocolo MLLP es el de proveer una interfaz de comunicación entre aplicaciones para transmitir información por la red. Los caracteres de control que fija el protocolo para transmitir un mensaje, son los siguientes:(35)

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



**Fig.1.** Estructura del mensaje utilizando MLLP

Para el correcto funcionamiento del Repositorio de Datos Clínicos se decidió utilizar el servicio de mensajería HL7, ya que cuenta con todos los mensajes que el repositorio necesita. El servicio ofrece la posibilidad a los sistemas externos de poder comunicarse con el repositorio.

## 2.3 Modelo de Dominio

Cuando en el entorno del negocio los procesos tienen un bajo nivel de estructuración y no existen flujos de información interconectados y bien definidos, se genera la imposibilidad de llevar a cabo un Modelo de Negocio, haciéndose necesario la realización de un Modelo de Dominio. El Modelo de Dominio, permite mostrar los principales conceptos que se manejan para el dominio del sistema en desarrollo. El diagrama del Modelo de Dominio se presenta en forma de diagrama de clases donde figuran los principales conceptos y roles del sistema analizado. (36)

### Conceptos Fundamentales

**Sistema Externo:** Se encarga de realizar la solicitud, ya sea para recuperar o para guardar los datos del paciente o de estudios.

**Solicitud:** Representan las solicitudes que pueden ser atendidas por el repositorio.

**Solicitud de lectura:** Encargada de solicitar los datos de los pacientes que el sistema externo desee.

**Solicitud de escritura:** Encargada de solicitar el almacenamiento de los datos referente a los pacientes que el sistema externo desee.

**Datos del Paciente:** Representa todos los datos del paciente ya sean datos generales o datos clínicos.

**Datos de Estudio:** Representa los datos de estudios realizados a los pacientes.

**Evidencias:** Representa todos los registros de los eventos del sistema, como las operaciones o los errores y las acciones de los sistemas externos.

**Operaciones:** Encargada de registrar las acciones que se realizan en el repositorio.

**Errores:** Encargada de guardar todos los errores que surgen, como caídas del sistema, etc.

**Operaciones realizadas:** Encargada de registrar las acciones de los sistemas externos en el repositorio.

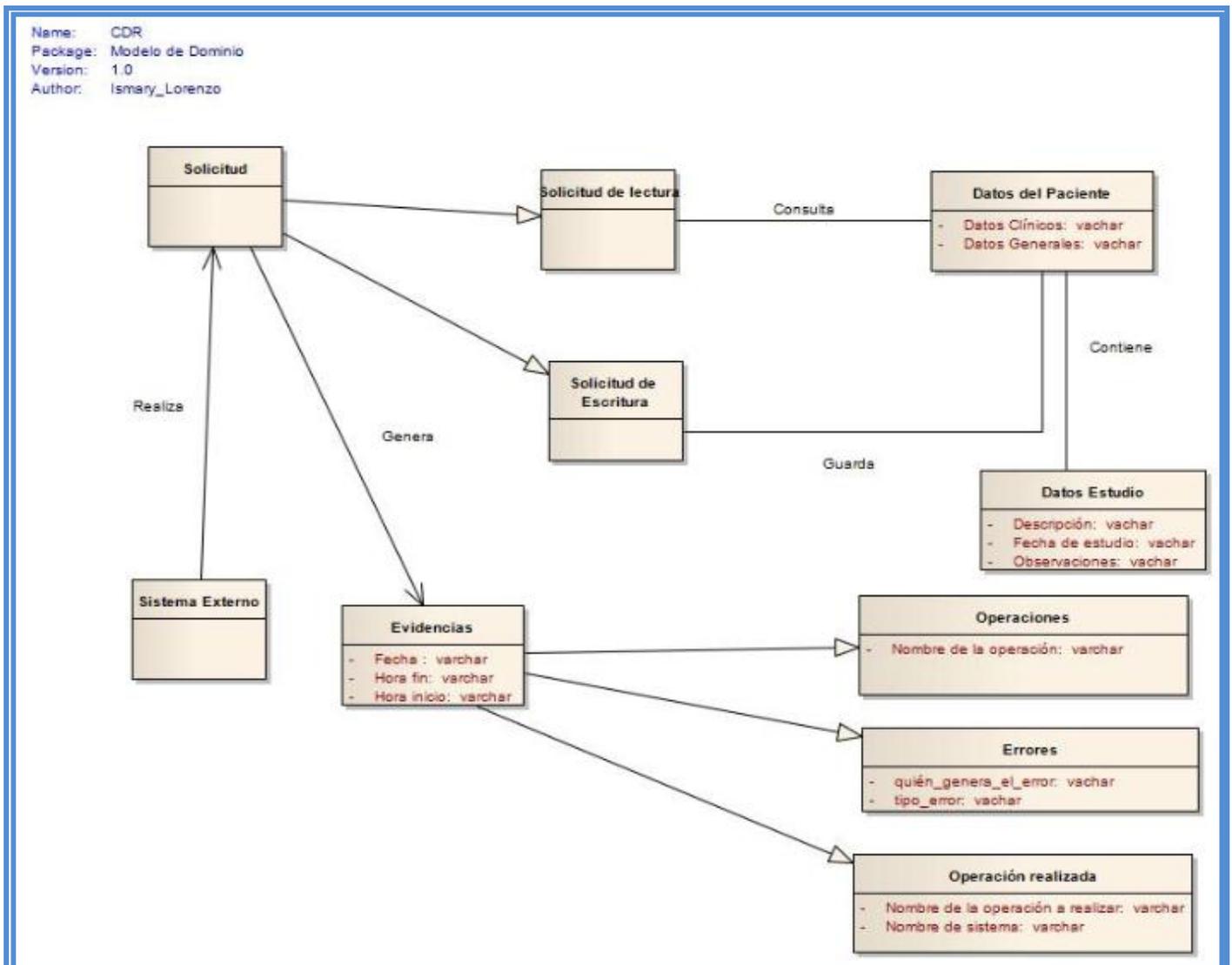


Fig. 2 Diagrama Modelo de Dominio

## 2.4 Especificación de requerimientos de software

La Ingeniería de requerimientos comprende las tareas relacionadas con la determinación de las necesidades o de las condiciones a satisfacer para un software nuevo o modificado, tomando en cuenta los diversos requerimientos de los inversores, que pueden entrar en conflicto entre ellos. Tiene como

propósito definir y validar los requerimientos con los que debe cumplir el software; así como proveer a los desarrolladores del sistema un mejor entendimiento de los requerimientos del sistema.

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

- Funcionales: son los que el usuario necesita que efectúe el software.
- No funcionales: son los "recursos" para que trabaje el sistema de información (redes, tecnología).(37)

#### **2.4.1 Requerimientos funcionales**

Como se define en la ingeniería de requerimientos, los requerimientos funcionales establecen los comportamientos del sistema, son capacidades o condiciones que el mismo debe cumplir. A continuación se listan los requerimientos funcionales del sistema propuesto: (37)

RF1. Modificar contraseña usuario administrador.

RF2. Agregar sistemas externos.

RF3. Modificar datos de sistemas externos.

RF4. Eliminar sistemas externos.

RF5. Asignar identificador de repositorio de datos.

RF6. Configurar el servidor de Bases de Datos donde se encontrará el repositorio.

RF7. Generar Base de Datos del sistema.

RF8. Autenticar administrador.

RF9. Seleccionar datos a cifrar en la base de datos.

RF10. Visualizar trazas del sistema.

RF11. Configurar copias de seguridad.

RF12. Verificar permisos de acceso de sistemas externos.

- RF13. Enviar Mensaje de "(ACK)".
- RF14. Generar Mensaje de "Aceptación (ACK)".
- RF15. Generar Mensaje de "Rechazo (ACK)".
- RF16. Recibir Mensaje de "Información de un paciente".
- RF17. Enviar Mensaje de "Información de un paciente".
- RF18. Generar Mensaje de "Información de un paciente".
- RF19. Enviar Mensaje de "registro de estudio clínicos de un paciente".
- RF20. Recibir Mensaje de "registro de estudio clínicos de un paciente".
- RF21. Generar Mensaje de "registro de estudio clínicos de un paciente".
- RF22. Recibir Mensaje de "Información de persona".
- RF23. Recibir Mensaje de "actualización de información de la persona".
- RF24. Recibir Mensaje de "actualización de información del paciente".
- RF25. Recibir Mensaje de "Solicitud de información clínica del paciente."
- RF26. Insertar información de la persona.
- RF27. Actualizar información de la persona.
- RF28. Buscar información de la persona.
- RF29. Insertar información del paciente.
- RF30. Actualizar información del paciente.
- RF31. Buscar información del paciente.
- RF32. Obtener información del paciente.
- RF33. Insertar información del médico.
- RF34. Actualizar información del médico.
- RF35. Buscar información del médico.
- RF36. Insertar información de la hoja frontal.
- RF37. Actualizar información de la hoja frontal.
- RF38. Buscar información de la hoja frontal.
- RF39. Obtener información de la hoja frontal.
- RF40. Insertar información acerca de la dirección del paciente.
- RF41. Actualizar información acerca de la dirección del paciente.
- RF42. Buscar información de la acerca de la dirección del paciente.

- RF43. Obtener información de la acerca de la dirección del paciente.
- RF44. Insertar información acerca de las Especialidades de los médicos.
- RF45. Actualizar información acerca de las Especialidades de los médicos.
- RF46. Buscar información acerca de las Especialidades de los médicos.
- RF47. Obtener información acerca de las Especialidades de los médicos.
- RF48. Insertar información acerca de los Datos Estudio del paciente.
- RF49. Actualizar información acerca de los Datos Estudio del paciente.
- RF50. Buscar información acerca de los Datos Estudio del paciente.
- RF51. Obtener información acerca de los Datos Estudio del paciente.
- RF52. Insertar información acerca del Diagnóstico Médico.
- RF53. Actualizar información acerca del Diagnóstico Médico.
- RF54. Buscar información acerca del Diagnóstico Médico.
- RF55. Obtener información acerca del Diagnóstico Médico.
- RF56. Insertar información acerca del Desarrollo del paciente.
- RF57. Actualizar información acerca del Desarrollo del paciente.
- RF58. Buscar información acerca del Desarrollo del paciente.
- RF59. Obtener información acerca del Desarrollo del paciente.
- RF60. Insertar información acerca del Período Neonatal del paciente.
- RF61. Actualizar información acerca del Período Neonatal del paciente.
- RF62. Buscar información acerca del Período Neonatal del paciente.
- RF63. Obtener información acerca del Período Neonatal del paciente.
- RF64. Insertar información acerca de las Citas de los pacientes.
- RF65. Actualizar información acerca de las Citas de los pacientes.
- RF66. Buscar información acerca de las Citas de los pacientes.
- RF67. Obtener información acerca de las Citas de los pacientes.
- RF68. Insertar información acerca de las Citas de Referencias de los pacientes.
- RF69. Actualizar información acerca de las Citas de Referencias de los pacientes.
- RF70. Buscar información acerca de las Citas de Referencias de los pacientes.
- RF71. Obtener información acerca de las Citas de Referencias de los pacientes.
- RF72. Insertar información acerca de las Citas de Interconsultas de los pacientes.

- RF73. Actualizar información acerca de las Citas de Interconsultas de los pacientes.
- RF74. Buscar información acerca de las Citas de Interconsultas de los pacientes.
- RF75. Obtener información acerca de las Citas de Interconsultas de los pacientes.
- RF76. Insertar información acerca de las Citas Sencillas de los pacientes.
- RF77. Actualizar información acerca de las Citas Sencillas de los pacientes.
- RF78. Buscar información acerca de las Citas Sencillas de los pacientes.
- RF79. Obtener información acerca de las Citas Sencillas de los pacientes.
- RF80. Insertar información acerca del Estado de los pacientes.
- RF81. Actualizar información acerca del Estado de los pacientes.
- RF82. Buscar información acerca del Estado de los pacientes.
- RF83. Obtener información acerca del Estado de los pacientes.
- RF84. Insertar información acerca de los Antecedentes Hemato Oncológicos de los pacientes.
- RF85. Actualizar información acerca de los Antecedentes Hemato Oncológicos de los pacientes.
- RF86. Buscar información acerca de los Antecedentes Hemato Oncológicos de los pacientes.
- RF87. Obtener información acerca de los Antecedentes Hemato Oncológicos de los pacientes.
- RF88. Insertar información acerca de los Antecedentes Prenatales de los pacientes.
- RF89. Actualizar información acerca de los Antecedentes Prenatales de los pacientes.
- RF90. Buscar información acerca de los Antecedentes Prenatales de los pacientes.
- RF91. Obtener información acerca de los Antecedentes Prenatales de los pacientes.
- RF92. Insertar información acerca de los Antecedentes sexuales Reproductivos de los pacientes.
- RF93. Actualizar información acerca de los Antecedentes sexuales Reproductivos de los pacientes.
- RF94. Buscar información acerca de los Antecedentes sexuales Reproductivos de los pacientes.
- RF95. Obtener información acerca de los Antecedentes sexuales Reproductivos de los pacientes.
- RF96. Insertar información acerca de Antecedentes Gineco Obstétricos de los pacientes.
- RF97. Actualizar información acerca de Antecedentes Gineco Obstétricos de los pacientes.
- RF98. Buscar información acerca de los Antecedentes Gineco Obstétricos de los pacientes.
- RF99. Obtener información acerca de los Antecedentes Gineco Obstétricos de los pacientes.
- RF100. Insertar información acerca de los Flujo Ginecostétrico de los Antecedentes Gineco Obstétricos de los pacientes.

- RF101. Actualizar información acerca de los Flujo Ginecostétrico de los Antecedentes Gineco Obstétricos de los pacientes.
- RF102. Buscar información acerca de los Flujo Ginecostétrico de los Antecedentes Gineco Obstétricos de los pacientes.
- RF103. Obtener información acerca de los Flujo Ginecostétrico de los Antecedentes Gineco Obstétricos de los pacientes.
- RF104. Insertar información acerca de la Ubicación de los pacientes.
- RF105. Actualizar información acerca de la Ubicación de los pacientes.
- RF106. Buscar información acerca de la Ubicación de los pacientes.
- RF107. Obtener información acerca de la Ubicación de los pacientes.
- RF108. Insertar información acerca de la Entidad de la ubicación de los pacientes.
- RF109. Actualizar información acerca de la Entidad de la ubicación de los pacientes.
- RF110. Buscar información acerca de la Entidad de la ubicación de los pacientes.
- RF111. Obtener información acerca de la Entidad de la ubicación de los pacientes.
- RF112. Insertar información acerca del Servicio de los pacientes.
- RF113. Actualizar información acerca del Servicio de los pacientes.
- RF114. Buscar información acerca del Servicio de los pacientes.
- RF115. Obtener información acerca del Servicio de los pacientes.
- RF116. Insertar información acerca de los Servicios Flotantes de los pacientes.
- RF117. Actualizar información acerca de los Servicios Flotantes de los pacientes.
- RF118. Buscar información acerca de los Servicios Flotantes de los pacientes.
- RF119. Obtener información acerca de los Servicios Flotantes de los pacientes.
- RF120. Insertar información acerca del departamento de servicio de los pacientes.
- RF121. Actualizar información acerca del departamento de servicio de los pacientes.
- RF122. Buscar información acerca del departamento de servicio de los pacientes.
- RF123. Obtener información acerca del departamento de servicio de los pacientes.
- RF124. Cifrar datos.
- RF125. Descifrar datos.
- RF126. Realizar copia de seguridad.
- RF127. Restaurar la base de datos desde una copia de seguridad.

- RF128. Generar Logs de seguridad.
- RF129. Generar Logs de errores del sistema.
- RF130. Generar logs de operaciones del sistema.
- RF131. Leer archivo de logs.
- RF132. Verificar procesos pendientes.
- RF133. Ejecutar procesos pendientes.
- RF134. Registrar sistema que se comunican con el repositorio.
- RF135. Recuperar Mensaje "Recuperar datos de estudio".
- RF136. Visualizar trazas de seguridad.
- RF137. Visualizar trazas de operaciones.
- RF138. Visualizar trazas de errores.

#### **2.4.2 Requerimientos no funcionales**

Un requerimiento no funcional es, en la ingeniería de sistemas y la ingeniería de software, un requerimiento que especifica criterios que pueden usarse para juzgar la operación de un sistema en lugar de sus comportamientos específicos, ya que éstos corresponden a los requerimientos funcionales, son propiedades o cualidades que el producto debe tener. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable(37). Son clasificados de la siguiente forma:

##### **Usabilidad**

**RNF1** Solo podrá utilizarse por los sistemas que sean capaces de emitir y recibir mensajes HL7 y que estén registrados en el repositorio como sistemas conocidos.

**RNF2** Se utilizará para el envío y recepción de mensajes HL7, el Servicio de Mensajería HL7 desarrollado en el departamento Tecnologías, Integración y Estándares perteneciente al Centro de Informática Médica.

**RNF3** Se pueden utilizar para la recepción de mensajes los puertos del 500 al 511 seleccionando solo uno entre los disponibles para la escuchar los mensajes de aceptación (ACK) y otro para la escucha de

cualquier tipo de mensaje. El puerto por el que estará escuchando el sistema externo es de total responsabilidad del sistema externo.

**RNF4** Se utilizará para la comunicación con el RDC el protocolo MLLP para la implementación de mensajes.

### **Confiabilidad**

**RNF5** El sistema contará con un registro de archivos logs, que almacenarán todo lo relacionado a las operaciones y errores que suceda en el repositorio, así como las operaciones de los sistemas externos dentro del mismo.

**RNF6** Si existe algún error como interferencia del fluido eléctrico o interrupción de comunicación en la red, el sistema brinda la posibilidad de no dejar operaciones inconclusas luego del acontecimiento.

### **Soporte**

El soporte es un aspecto realmente amplio y abarca todas las acciones a tomar una vez que se ha terminado el desarrollo del software con motivos de asistir a los clientes de este, así como lograr su mejoramiento progresivo y evolución en el tiempo. Puede incluir: pruebas, adaptabilidad y mantenimiento del sistema.(37)

**RNF7** El sistema contará antes de su puesta en marcha con un período de pruebas que permitirá identificar y corregir errores cometidos durante el desarrollo, posteriormente se brindará el servicio de instalación del producto.

**RNF8** Se generará la documentación apropiada para agilizar su mantenimiento y configuración.

### **Seguridad**

**RNF9** El repositorio debe ser capaz de brindar acceso al contenido solo a los sistemas previamente registrados.

**RNF10** El repositorio necesita saber con confianza con quién se está comunicando.

**RNF11** Se garantizará el tratamiento adecuado de las excepciones y validaciones de las entradas de datos proporcionados por el sistema externo.

**FNF12** El repositorio permitirá que parte de la información almacenada pueda estar cifrada.

**RFN13** Se realizarán copias de seguridad periódicas.

### **Hardware**

**RNF13** Se propone una estación de trabajo de 2 TB de almacenamiento como mínimo, al menos 4 GB de RAM, un microprocesador superior a Intel(R) Dual Core(TM) i3-2120 CPU o un equivalente del mismo en la tecnología AMD, para el funcionamiento del repositorio.

**RNF14** Se propone como sistema gestor de base de datos PostgreSQL en su versión 9.1.

### **Software**

**RNF15** El repositorio funcionará en sistema operativo Windows en un servidor IIS (Internet Information Services) o en un servidor apache que incluyas las librerías dotnet.

**RNF16** El repositorio funcionará en sistema operativo Linux con un servidor apache instalado. El mismo deberá tener incluidas las librerías mono para el buen funcionamiento del Servicio de Mensajería HL7.

**RNF17** Para la administración del repositorio es necesario contar con un navegador web que puede ser Internet Explorer o Mozilla Firefox.

## **2.5 Casos de uso del sistema**

Son utilizados para capturar el comportamiento deseado del sistema en desarrollo, sin tener que especificar cómo es implementado este comportamiento. Ofrecen un medio para que los desarrolladores, los usuarios finales del sistema y los expertos del dominio lleguen a una comprensión común del sistema. Un caso de uso describe un proceso de principio a fin, por tanto una secuencia de eventos, las acciones y las transacciones que se requieren para realizarlo. (38)

### Actores del sistema

Los actores del sistema representan roles y son personas (o dispositivos) que juegan como impulsores del sistema. Definido más formalmente, un actor se comunica con el sistema o producto y que es externo al sistema en sí mismo. Es decir un actor es una idealización de una persona externa, que utiliza el sistema o producto.

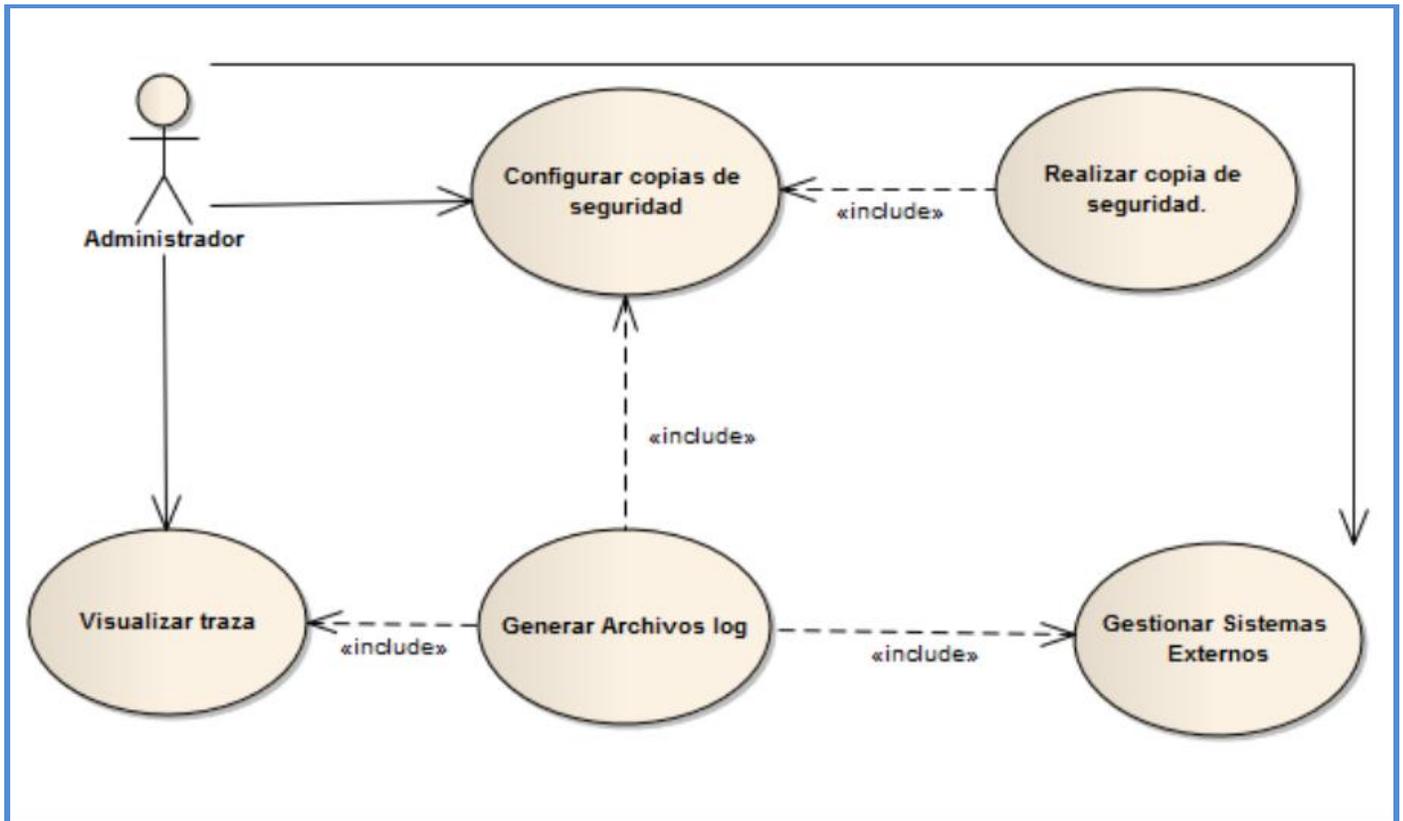
Para el desarrollo del RDC se identificaron los siguientes actores del sistema:

Actor	Descripción
Administrador	Rol responsable de toda la configuración del sistema y la visualización de las trazas de seguridad.
Sistema Externo	Rol ficticio para representar los sistemas que pueden interactuar con el repositorio como inicializadores de algún caso de uso.

**Tabla 1.** Actores del Sistema

Se identificaron 43 casos de uso del sistema, por lo extenso y engorroso que sería representar en un solo diagrama todos los casos de uso, se separaron en paquetes. El paquete administrador contiene todos los casos de uso correspondientes a las configuraciones y visualizaciones que el mismo debe realizar. El paquete sistema externo contiene todos los casos de usos correspondientes a los mensajes que se destinan al repositorio y los que el mismo conforma y envía, también abarca los referentes a todos los datos que el sistema externo desea almacenar o recuperar, ya sean inserciones o modificaciones de información del paciente o datos de estudios del mismo.

A continuación se muestran el diagrama de los casos de usos arquitectónicamente significativos del actor administrador.



**Fig. 3** Diagrama de CU: Administrador

A continuación se muestran el diagrama de los casos de usos arquitectónicamente significativos del actor sistema externo.

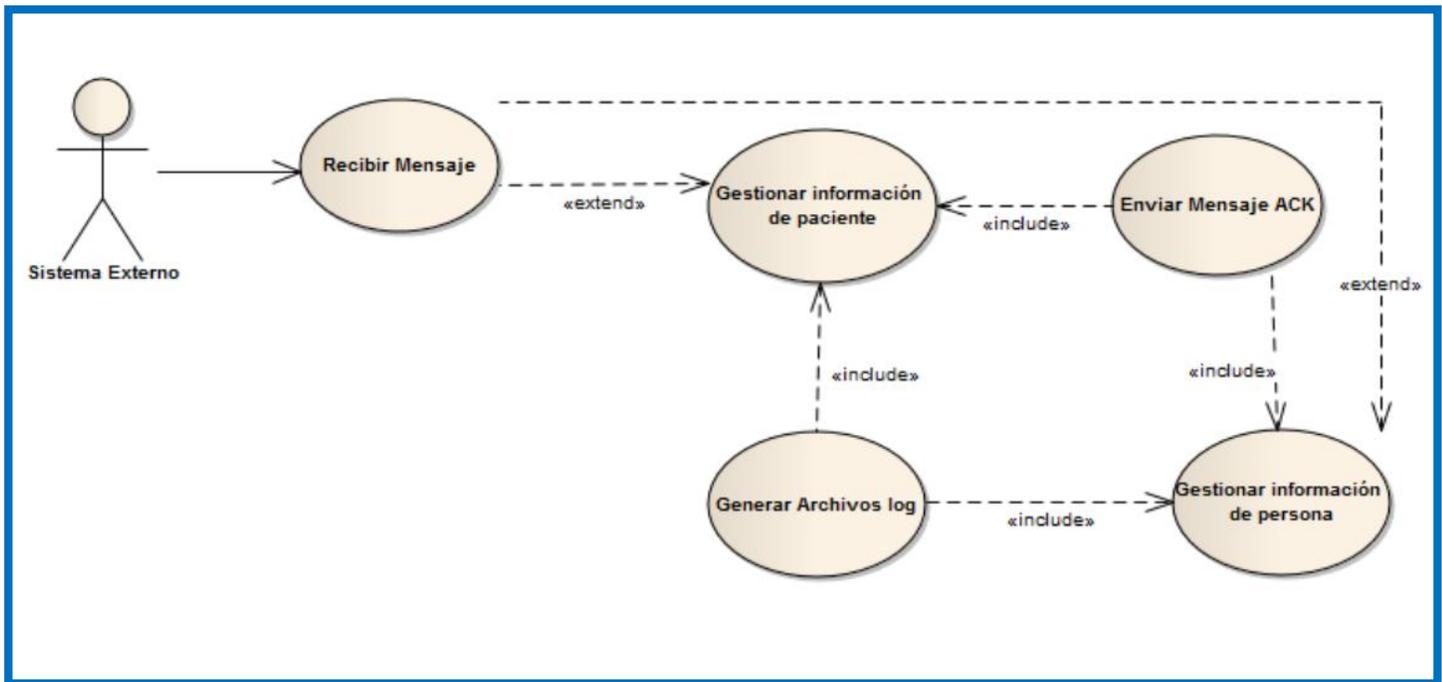


Fig. 4 Diagrama de CU: Sistema Externo

A continuación se muestra la dirección donde se encuentran las especificaciones de los caso de uso.

**(Remitirse a: Expediente. Proyecto\1. Ingeniería\1.2 requisitos, al documento CDR\_TIE\_0114\_Especificación de Casos de Uso)**

### 2.6 Descripción textual de algunos casos de uso

Se especifica la descripción textual del caso de uso gestionar información del paciente, del actor sistema externo, ya que es considerado uno de los casos de uso más importantes.

#### CU: Gestionar información de paciente.

<b>Objetivo</b>	Gestionar información de los datos del paciente.
<b>Actores</b>	Sistema Externo(inicia)

<b>Resumen</b>	El CU inicia cuando el sistema externo envía el mensaje ADT_A01 o ADT__A08.	
<b>Complejidad</b>	Media	
<b>Prioridad</b>	Crítico	
<b>Referencia</b>	RF 29,30,31,32	
<b>Precondiciones</b>	Recibir Mensaje ADT_A01 o ADT_A08.	
<b>Postcondiciones</b>	Se inserta o modifica información del paciente.	
<b>Flujo de eventos</b>		
<b>Flujo básico Insertar datos de paciente</b>		
	<b>Actor</b>	<b>Sistema</b>
1.	Envía mensaje ADT_A01	
2.		Registra operación insertar datos del paciente del sistema externo. <b>VerCU # 14 Generar archivos log</b>
3.		Registra operación inicio insertar datos de paciente. <b>VerCU # 14 Generar archivos log</b>
4.		Verifica si existen los datos.
5.		Registrar operación inicio verificación datos de paciente. <b>VerCU # 14 Generar archivos log</b>
6.		Registrar operación fin verificación datos de paciente. <b>VerCU # 14 Generar archivos log</b>
7.		Valida dato, para saber si el formato es correcto.

8.		Registrar operación inicio validación datos con formato correcto. <b>VerCU # 14 Generar archivos log</b>
9.		Registrar operación fin validación datos con formato correcto. <b>VerCU # 14 Generar archivos log</b>
10.		Verificar si existen campos vacíos.
11.		Registrar operación inicio validación campos repetidos. <b>VerCU # 14 Generar archivos log</b>
12.		Registrar operación fin validación campos repetidos. <b>VerCU # 14 Generar archivos log</b>
13.		Registrar operación inicio conexión con servidor de base datos. <b>VerCU # 14 Generar archivos log</b>
14.		Conectarse con el servidor de base de datos.
15.		Verifica si quiere cifrar datos.
16.		Se inserta datos de paciente, como: <ul style="list-style-type: none"> <li>• Número de historia clínica</li> <li>• Id persona</li> <li>• Peso</li> <li>• Talla</li> <li>• Uid Paciente</li> <li>• Id estudio</li> <li>• Id hoja frontal</li> </ul>
17.		Registrar operación fin conexión con servidor de base datos. <b>VerCU # 14 Generar archivos log</b>
18.		Registra operación fin insertar datos de paciente. <b>VerCU # 14 Generar archivos log</b>
19.		<b>Enviar Mensaje ACK Ver CU # 42</b>

### Flujos alternos

4a Los datos existen		
	Actor	Sistema
1.		Registra operación cancelar insertar datos de paciente. <b>VerCU # 14 Generar archivos log</b>
2.		Registra error insertando datos de paciente. <b>VerCU # 14 Generar archivos log</b>
3.		Los datos existen. <b>Enviar Mensaje ACK Ver CU #:42</b>
7a Saber si el formato es correcto		
	Actor	Sistema
1.		Registra operación cancelar insertar datos de paciente. <b>VerCU # 14 Generar archivos log</b>
2.		Registra error insertando datos de paciente. <b>VerCU # 14 Generar archivos log</b>
3.		El formato de los datos es incorrecto. <b>_Ver CU #:42 Enviar Mensaje ACK</b>
10a Existen campos vacíos		
	Actor	Sistema
1.		Registra operación cancelar insertar datos de paciente. <b>VerCU # 14 Generar archivos log</b>
2.		Se modifican los campos. <b>Ir Sección 1</b>
14a Conectarse con el servidor de base de datos		
	Actor	Sistema
1.		Registra operación cancelar insertar datos de paciente. <b>VerCU # 14 Generar archivos log</b>

2.		Registra error insertando datos de paciente. <b>VerCU # 14 Generar archivos log</b>
<b>15a Verifica si quiere cifrar datos.</b>		
	<b>Actor</b>	<b>Sistema</b>
1.		Registrar operación inicio de cifrado. <b>VerCU # 14 Generar archivos log</b>
2.		Cifrar datos <b>Ver CU #</b>
3.		Registrar operación fin de cifrado. <b>VerCU # 14 Generar archivos log</b>
<b>Sección 1: Modificar datos de paciente.</b>		
	<b>Actor</b>	<b>Sistema</b>
1.	Envía mensaje ADT_A08	
2.		Registra operación inicio modificar datos de paciente. <b>VerCU # 14 Generar archivos log</b>
3.		Actualiza en los campos vacíos los datos nuevos del paciente.
4.		Registra operación fin modificar datos de paciente. <b>VerCU # 14 Generar archivos log</b>
5.		<b>Enviar Mensaje ACK. Ver CU # 42</b>
<b>Relaciones</b>	<b>CU Incluidos</b>	<b>CU # 42 Enviar Mensaje CU # 14 Generar archivos log</b>
	<b>CU Extendidos</b>	

Tabla 2. CU Gestionar información de paciente, Flujo Básico Insertar información de paciente

## 2.7 Arquitectura en Capas

Los estilos arquitectónicos son los que definen la estructura de un componente de software y se componen de subsistemas con sus responsabilidades(39). Una vez concluida la ingeniería de requisitos, donde se definieron las características y restricciones del componente, se definió que la arquitectura a utilizar es en capas ya que el sistema que se implementará presenta varias funcionalidades y esta arquitectura permite darle responsabilidades a cada capa, si se quiere modificar alguna capa no afectará a las demás. Las capas o niveles pueden estar implementadas internamente por objetos o procedimientos(39). Las mismas quedan definidas de la siguiente forma:

### **Capa de Presentación:**

Todo lo relacionado con la interacción entre el repositorio y el usuario administrador: versiones HL7 que van a estar admitiendo, configuración de la BD, estaciones de trabajo con acceso al repositorio, identificador del repositorio. Se comunica con la Capa de Datos y de ser necesario con las demás capas. Gestiona la accesibilidad a la aplicación de configuración.

### **Capa de Comunicación:**

Todo lo relacionado con el envío y recepción de mensajes mediante el Servicio de Mensajería HL7, el chequeo de los mensajes que se reciben, chequeo de los mensajes que se generan. Gestiona lo referente a la comunicación con otras aplicaciones (estaciones que tienen privilegios en el repositorio). Le envía los mensajes recibidos a la Capa de Negocio.

### **Capa de Negocio:**

Interpretación del mensaje que recibe: diferenciar entre peticiones de datos o envío de datos.

- Peticiones de información: preparar información (comunicación con Capa de Datos) y generar el mensaje HL7 para enviarlo a la Capa de Comunicación.
- Envío de información: desglosar la información recibida para su almacenamiento, además tiene que generar un mensaje ACK (mensaje de aceptación o rechazo).

### **Capa de Datos:**

Esta capa se puede comunicar con la Capa de Presentación, ya que el administrador puede configurar la base de datos y las copias de seguridad. Es la capa encargada del cifrado y descifrado de la información en el caso que se requiera. Almacenamiento y extracción de datos.

#### Capa de Respaldo y Trazas:

Trazabilidad de todas las operaciones que se realizan en el repositorio, mediante el registro en logs. Revisar operaciones inconclusas: generar logs de operaciones, logs de errores y logs relacionados a las actividades que realiza el sistema externo.



**Fig. 5** Arquitectura en capas

#### 2.8 Patrones de diseño

Los patrones utilizados en el desarrollo del sistema son los denominados Patrones Generales de Software para Asignación de Responsabilidades (GRASP) estos describen los principios fundamentales de diseño de objetos para la asignación de responsabilidades (40). Constituyen un apoyo para la enseñanza que ayuda a entender el diseño de objeto esencial y aplica el razonamiento para el diseño de una forma sistemática, racional y explicable. Para el diseño del sistema se utilizaron los patrones que vienen implícitos en el marco de trabajo empleado.

- **Experto:** es un patrón que se usa más que cualquier otro al asignar responsabilidades; es un principio básico que suele utilizarse en el diseño orientado a objetos. Expresa simplemente la “intuición” de que los objetos hacen actividades relacionadas con la información que poseen. Se conserva el encapsulamiento, ya que los objetos se valen de su propia información para hacer lo que se les pide. Esto soporta un bajo acoplamiento, lo que favorece el hecho de tener sistemas más robustos y de fácil mantenimiento. (40).
- **Controlador:** Consiste en asignar una responsabilidad de recibir o manejar un mensaje de evento del sistema a una clase. (40).
- **Alta Cohesión:** es la medida de la fuerza que une a las responsabilidades de una clase. Una clase con baja cohesión es aquella que hace muchas cosas no afines o muchas tareas, lo que trae como consecuencias dificultades para entender, reutilizar y conservarla. Son delicadas y las afectan constantemente los cambios. Una clase con alta cohesión mejora la claridad y la facilidad de su uso, su mantenimiento se simplifica y es fácil de reutilizar. A menudo se genera un bajo acoplamiento. (40).
- **Bajo acoplamiento:** es la medida de la fuerza con que una clase está conectada a otras clases. De tal forma que en caso de producirse una modificación en alguna de ellas, se tenga la mínima repercusión posible en el resto de clases, potenciando la reutilización, y disminuyendo la dependencia entre las clases. (40).

Estos patrones se aplicaron en la definición de cada clase del sistema con el objetivo de lograr que las clases cumplan las funciones específicas de cada una sin que se sobrecarguen (experto).

Las relaciones entre clases están limitadas solamente a la dependencia de conceptos, por lo que cada clase solamente se relaciona con aquellas cuyos conceptos dependen completamente de ella asegurando la mínima comunicación posible (bajo acoplamiento). Se definieron las clases controladoras de forma que cada una solo realiza un conjunto de tareas específicas sin suplantar las responsabilidades de otra (alta cohesión).

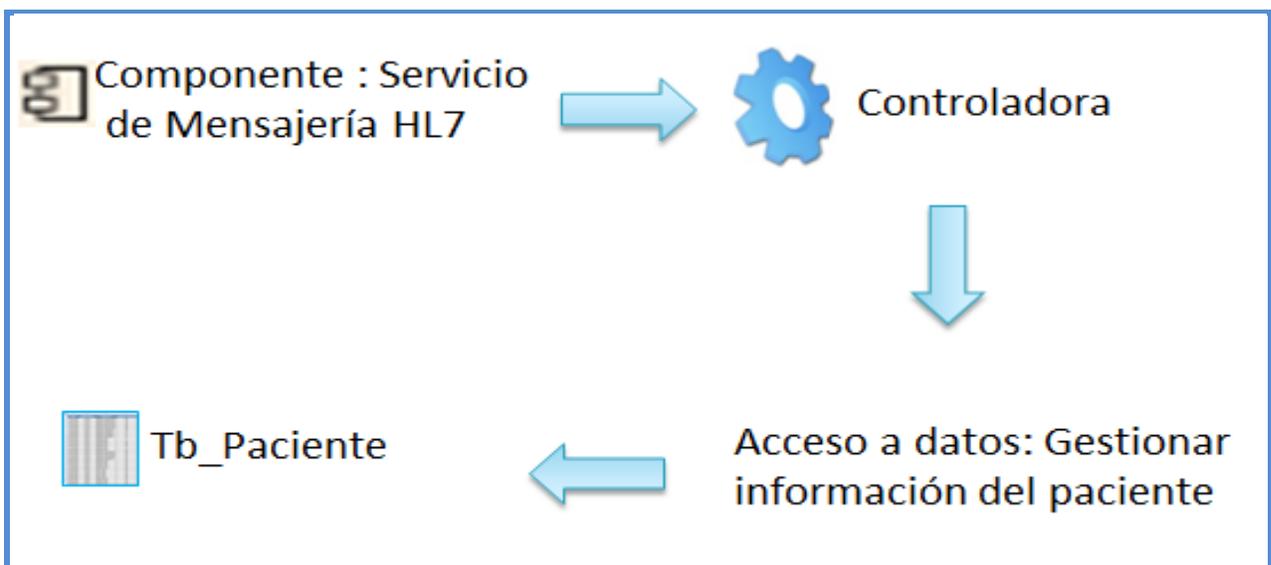
## 2.9 Diagramas de Clases del Diseño

### Descripción de las clases del diseño

El Diagrama de Clases es el diagrama principal en la fase de diseño. Un diagrama de clases presenta las clases del sistema con sus relaciones estructurales y de herencia. La definición de clase incluye definiciones para atributos y operaciones (40).

Se realizaron los diagramas de clase del diseño, a continuación se muestran en imágenes la explicación de los diagrama de clase del caso de uso gestionar información del paciente y configurar copias de seguridad. La especificación de estos diagramas se encuentra en la siguiente dirección.

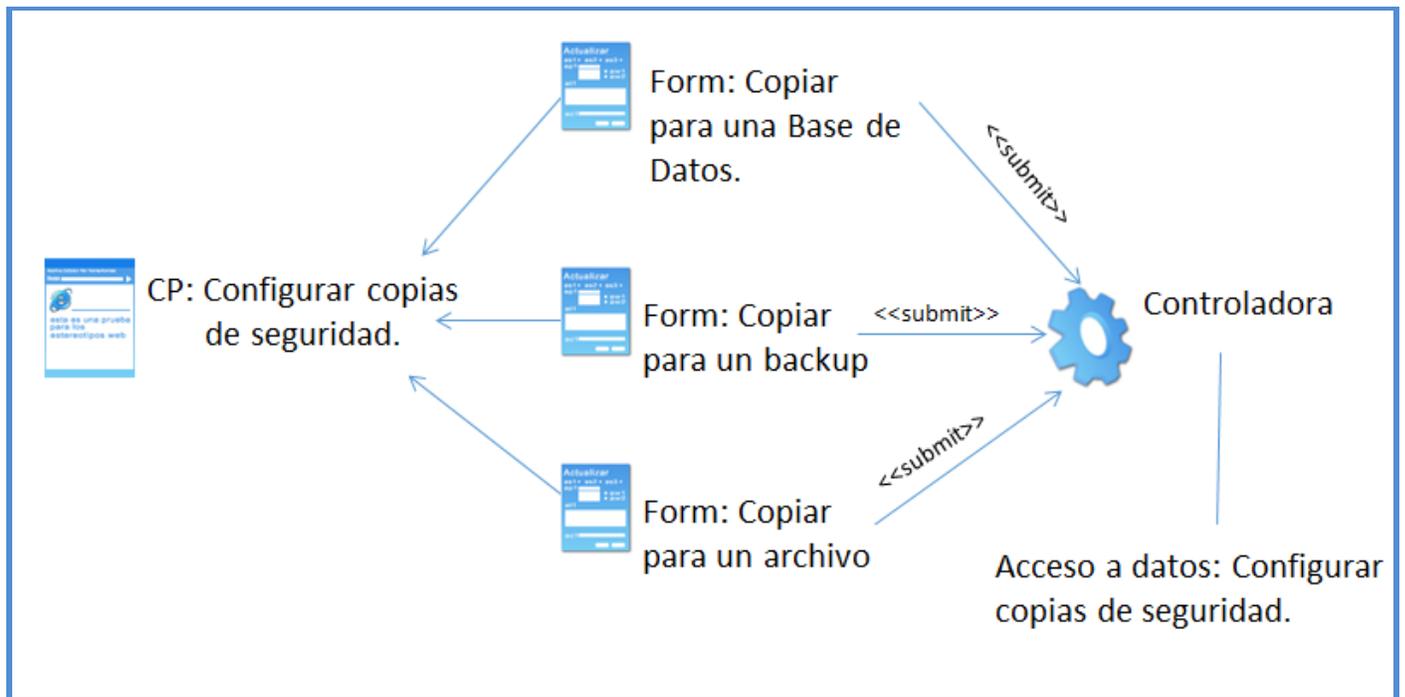
(Remitirse a: Expediente. Proyecto\1. Ingeniería\1.2 arquitectura y diseño, al documento CDR\_TIE\_0121\_Modelo de diseño v2.0)



**Fig. 6 Representación del diagrama de clase: CU Gestionar información de paciente**

El diagrama de clases contiene un componente servicio de mensajería que es el encargado de enviar el mensaje a la clase server page o clase controladora, la cual es la responsable de verificar el tipo de mensaje y que operación quiere realizar. La clase controladora se relaciona con la clase de acceso a

datos, gestionar información de paciente, y la misma se relaciona con la entidad paciente que contiene los datos del mismo.



**Fig. 7** Representación del diagrama de clase: CU Configurar copias de seguridad

El diagrama contiene una clase página cliente o clase de presentación configurar copias de seguridad, donde el administrador selecciona la opción de la copia que quiere realizar, ya sea para un backup, un archivo u otra base de datos y en correspondencia con lo que seleccione, se crea un formulario para llenar los campos correspondientes. Los formularios tienen la responsabilidad de enviar los datos que entra el usuario a la controladora, la cual tiene dentro de sus funciones crear las copias de seguridad en dependencia de la selección del usuario. La clase de acceso a datos es la encargada de gestionar todo lo referente a la información.

## 2.10 Diagramas de Interacción

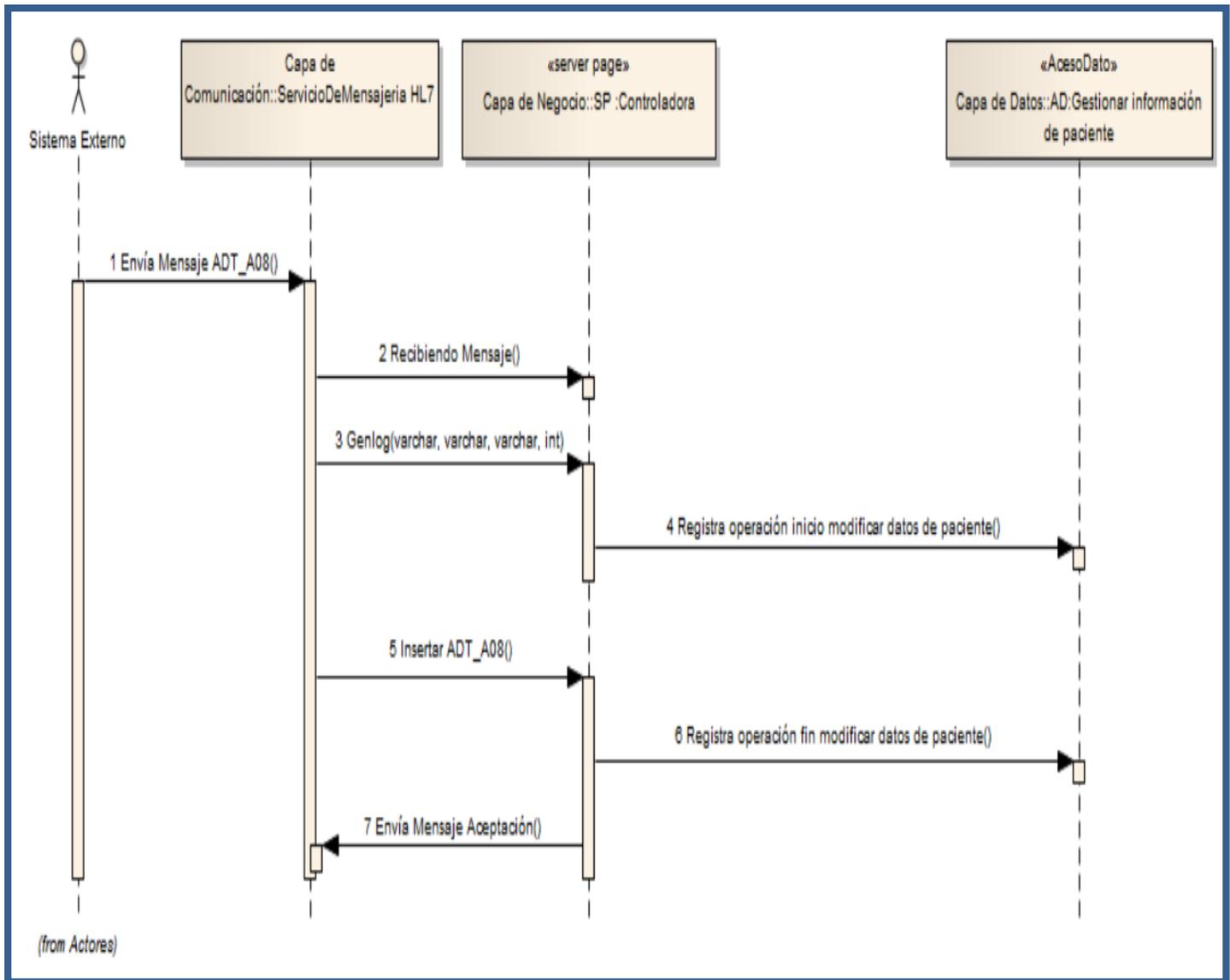
Estos diagramas se pueden expresar en diagramas de secuencia y en diagramas de colaboración. Los primeros muestran las interacciones entre objetos, ordenadas en secuencia temporal durante un

escenario concreto, los segundos destacan la organización de los objetos que participan en una interacción. En los diagramas de secuencia se modelan las interacciones entre las clases de diseño mediante la transferencia de mensajes entre objetos o subsistemas(41).

Se realizaron los diagramas de secuencias, a continuación se muestran algunos de los mismos. Los demás diagramas se encuentran en la siguiente dirección:

**(Remitirse a: Expediente. Proyecto\1. ingeniería\1.2 arquitectura y diseño, al fichero CDR.eap)**

Para la realización del caso de uso gestionar información del médico se determinaron varios diagramas de secuencia construidos por escenarios, a continuación se muestra el diagrama para el escenario Modificar Información del Médico, del caso de uso Gestionar Información del Médico.



**Fig. 8** Diagrama de secuencia: CU Modificar Información del Médico

Se abordaron en el presente capítulo las principales características del sistema, así como un análisis del modelo de dominio que contiene los principales conceptos para el desarrollo del repositorio. Se identificaron los requisitos que se deben cumplir conformando las funcionalidades del sistema mediante la obtención de casos de usos. Se realizó la descripción de la arquitectura del software.

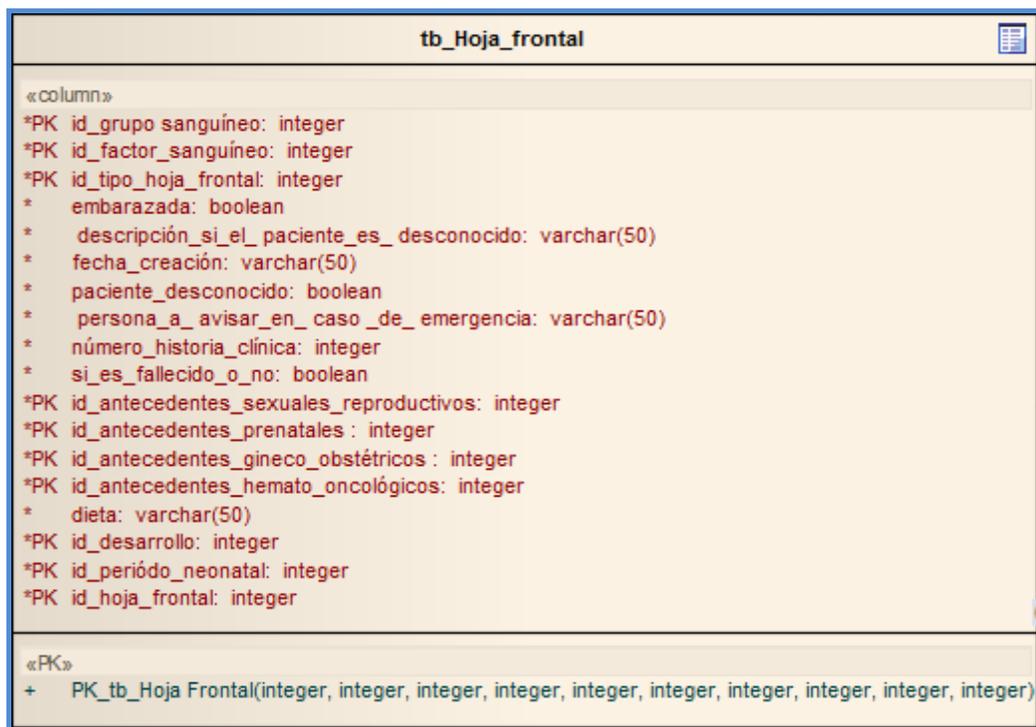
## CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBA

El presente capítulo tiene como objetivo abordar el flujo de trabajo de implementación donde se describe cómo se implementan los elementos del modelo de diseño. Se explican algunas características del servicio de mensajería HL7. Se realiza el modelo de datos, diagrama de despliegue y de componente. Se ejecutan los casos de pruebas para probar el correcto funcionamiento del sistema implementado.

### 3.1 Modelo de datos

Es una descripción de un contenedor de datos, así como de los métodos para almacenar y recuperar información de esos contenedores. Los modelos de datos no son elementos físicos, son abstracciones que permiten la implementación de un sistema de base de datos(42).

Para el funcionamiento del sistema se diseñó un Modelo de Datos de clases persistentes que describe la información que se manejará en el repositorio. Se muestran algunas de las clases más significativas.



**Fig. 9** Clase hoja frontal

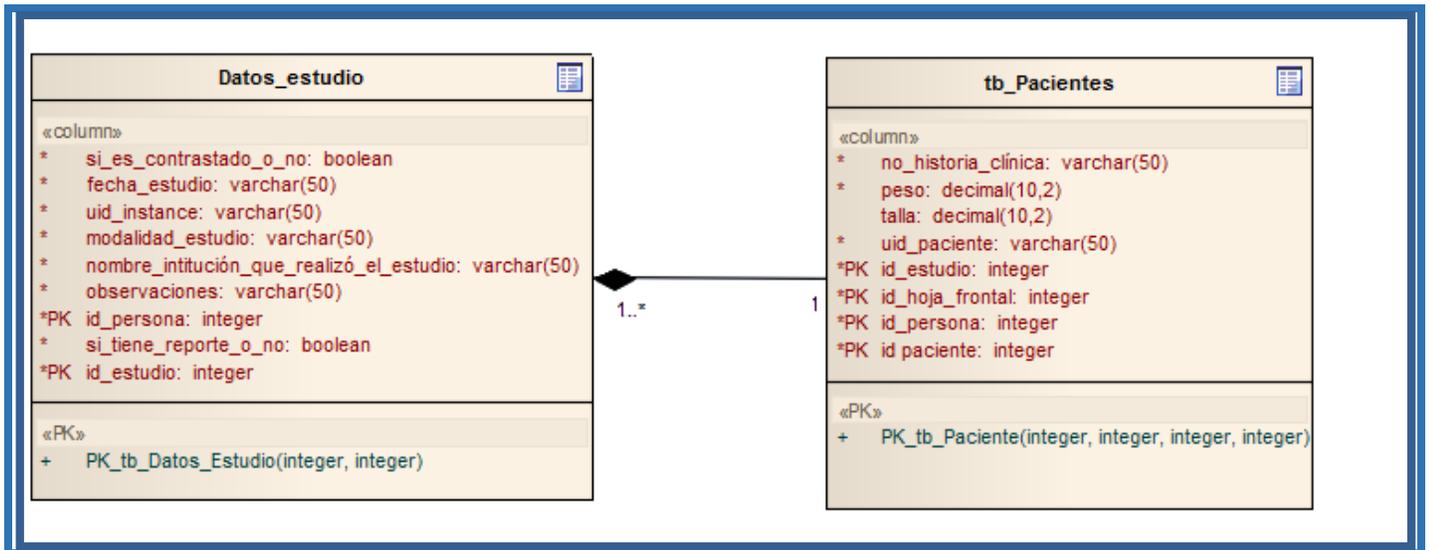


Fig. 10 Clase datos de estudio y clase paciente

### 3.1.1 Descripción textual de algunas entidades importantes del modelo de datos

Se realizaron las descripciones de todas las entidades principales del modelo de datos, a continuación se presentaran algunas de las más importantes, y el resto se encuentra en la siguiente dirección:

(Remitirse a: Expediente. Proyecto\1. ingeniería\1.2 arquitectura y diseño, CDR\_TIE\_0104\_Diccionario de Datos)

#### Entidad Hoja Frontal

<b>Nombre de la entidad</b>	Hoja frontal.		
<b>Descripción de la entidad</b>	Esta entidad es sobre la hoja frontal de los pacientes.		
<b>Nombre del atributo</b>	<b>Descripción</b>	<b>Tipo</b>	<b>Puede ser nulo</b>
Id hoja frontal.	Identificador numérico consecutivo para las tuplas de la tabla.	Integer	No
embarazada	Si la paciente está embarazada.	Boolean	Si

Descripción si el paciente es desconocido.	Una Descripción si el paciente es desconocido o no.	Varchar	Si
Fecha creación	Fecha de creación de la hoja frontal.	Varchar	Si
Paciente desconocido	Si el paciente es desconocido o no.	Boolean	Si
Persona a avisar en caso de emergencia.	La persona a avisar en caso de emergencia.	Varchar	Si
Número de historia clínica.	Número de historia clínica del paciente.	Integer	Si
Si es fallecido o no.	Si el paciente es fallecido o no.	Boolean	Si
Dieta	Dieta de los pacientes.	Varchar	Si
Id grupo sanguíneo	Identificador que representa el grupo sanguíneo en cuestión que se encuentra en la tabla grupo sanguíneo.	Integer	No
Id factor sanguíneo	Identificador que representa el factor sanguíneo en cuestión que se encuentra en la tabla factor sanguíneo.	Integer	No
Id tipo hoja frontal	Identificador que representa el tipo hoja frontal en cuestión que se encuentra en la tabla tipo hoja frontal.	Integer	No
Id antecedentes sexuales reproductivos.	Identificador que representa el antecedentes sexuales reproductivos en cuestión que se encuentra en la tabla antecedentes sexuales reproductivos.	Integer	No
Id antecedentes prenatales.	Identificador que representa el antecedentes prenatales en cuestión que se encuentra en la tabla antecedentes prenatales.	Integer	No

Id antecedentes gineco obstétrico.	Identificador que representa el antecedentes gineco obstétrico en cuestión que se encuentra en la tabla antecedentes gineco obstétrico.	Integer	No
Id antecedentes hemato oncológicos.	Identificador que representa el antecedentes hemato oncológicos en cuestión que se encuentra en la tabla antecedentes hemato oncológicos.	Integer	No
Id desarrollo	Identificador que representa el desarrollo en cuestión que se encuentra en la tabla desarrollo.	Integer	No
Id periodo neonatal	Identificador que representa el periodo neonatal en cuestión que se encuentra en la tabla periodo neonatal.	Integer	No

**Tabla.3** Entidad Hoja Frontal

### ***Entidad Paciente***

<b>Nombre de la entidad</b>	Paciente		
<b>Descripción de la entidad</b>	Esta entidad es sobre los datos del paciente.		
<b>Nombre del atributo</b>	<b>Descripción</b>	<b>Tipo</b>	<b>Puede ser nulo</b>
Id paciente	Identificador numérico consecutivo para las tuplas de la tabla.	Integer	No
Número de historia clínica.	Número de historia clínica del paciente.	Varchar	Si
Peso	Peso del paciente.	Decimal	Si
talla	Talla del paciente.	Decimal	Si
Uid paciente.	Id universal del paciente.	Varchar	No
Id estudio	Identificador que representa el estudio en cuestión que se encuentra en la tabla	Integer	No

	datos de estudio.		
Id hoja frontal	Identificador que representa el hoja frontal en cuestión que se encuentra en la tabla hoja frontal.	Integer	No
Id persona	Identificador que representa el persona en cuestión que se encuentra en la persona.	Integer	No

**Tabla.4** Entidad Paciente

**Entidad Datos de Estudio**

<b>Nombre de la entidad</b>	Datos de estudio.		
<b>Descripción de la entidad</b>	Esta entidad es sobre los datos de estudio de los pacientes.		
<b>Nombre del atributo</b>	<b>Descripción</b>	<b>Tipo</b>	<b>Puede ser nulo</b>
Id datos de estudios.	Identificador numérico consecutivo para las tuplas de la tabla.	Integer	No
Si es contrastado o no	Si es contrastado o no los datos de estudio.	Boolean	Si
Fecha de estudio.	Fecha en la que se realizó el estudio.	Varchar	Si
Uid instance	Identificación de la instancia de los datos de estudio.	Varchar	Si
Uid paciente.	Id universal del paciente.	Varchar	No
Modalidad de estudio.	Modalidad de los datos de estudio del paciente.	Varchar	Si
Nombre de la institución que realizo el estudio.	Id del Nombre de la institución que realizo el estudio de los pacientes.	Varchar	Si
observaciones	Observaciones que se han presentados en los estudios.	Varchar	Si
Si tiene reporte o	Si tiene reporte o no, el estudio.	Boolean	Si

no.			
Id persona	Identificador que representa la persona en cuestión que se encuentra en la tabla persona.	Integer	No

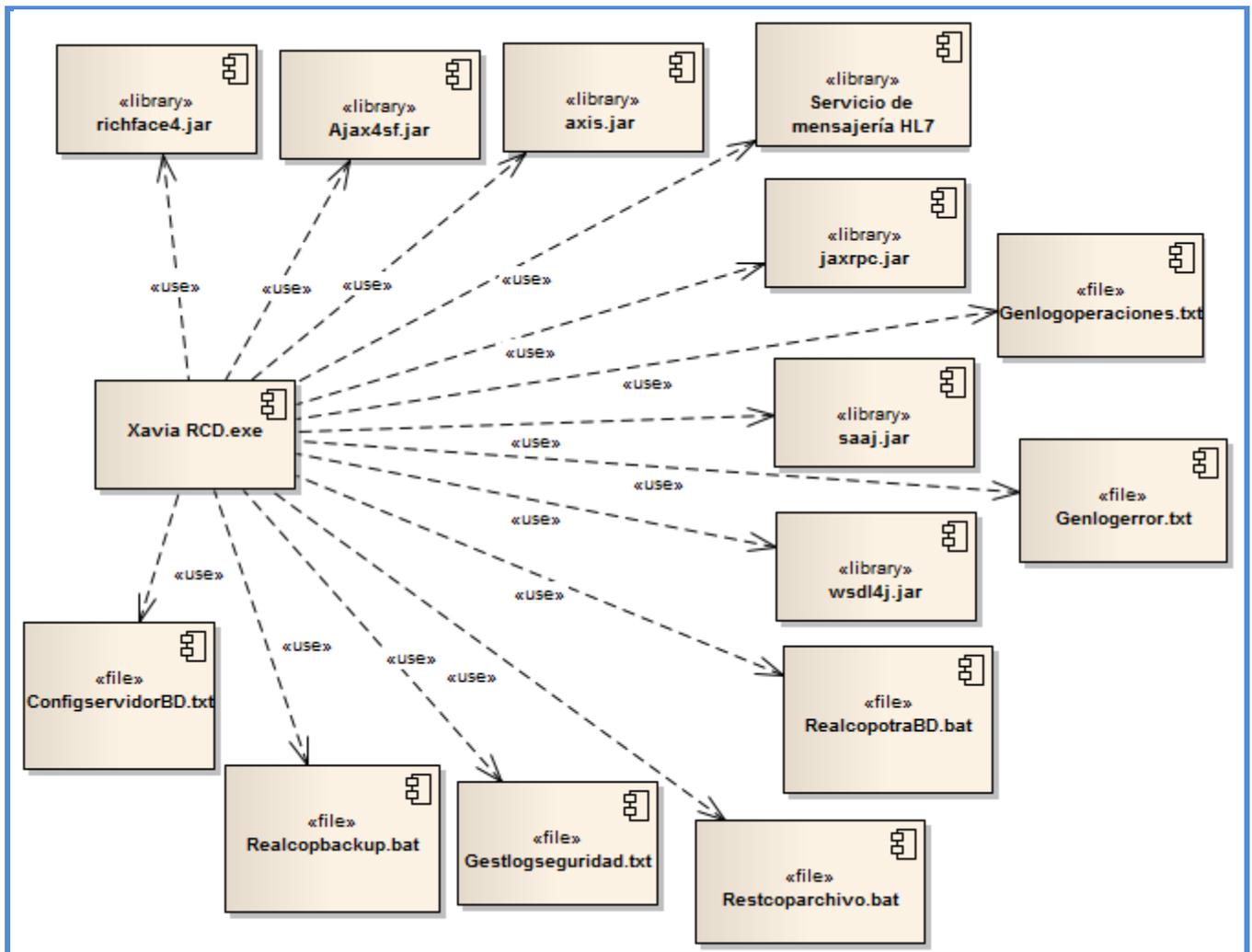
**Tabla.5** Entidad Datos de Estudio

### 3.2 Modelo de implementación

El modelo de implementación especifica cómo los elementos del modelo de diseño, fundamentalmente las clases, se implementan en términos de componentes como ficheros de código fuente, ejecutables, librerías, entre otros. También describe cómo se organizan los componentes de acuerdo con los mecanismos de estructuración y modularización disponibles en el entorno de implementación y en el lenguaje o lenguajes de programación utilizados, y cómo dependen los componentes unos de otros, así como su organización de acuerdo a los nodos específicos en el modelo de despliegue (43).

#### 3.2.1 Diagrama de componentes

Dentro del flujo de implementación se crea como artefacto el diagrama de componentes que describe los elementos físicos del sistema y sus relaciones; así como muestra las opciones de realización incluyendo código fuente, binario y ejecutable. Los componentes representan todos los tipos de elementos de software que entran en la fabricación de aplicaciones informáticas. Estos pueden ser simples archivos, paquetes, bibliotecas cargadas dinámicamente, entre otros.(44)



**Fig.11** Diagrama de componentes

Las librerías wsdl4.jar, axis.jar, jaxrpc.jar y saaj.jar son necesarias para consumir del servicio de mensajería HL7, las Ajax4sf.jar y richface4.jar son utilizadas para el trabajo con la interfaces y el servicio de mensajería HL7 es el componente que se utilizará para el envío y recepción de mensajes HL7 en la versión 2.3.1.

Los archivos utilizados son configservidorBD.txt el cual es el responsable de almacenar la información relacionada a la configuración del servidor en la Base de Datos y contiene datos referente al nombre del

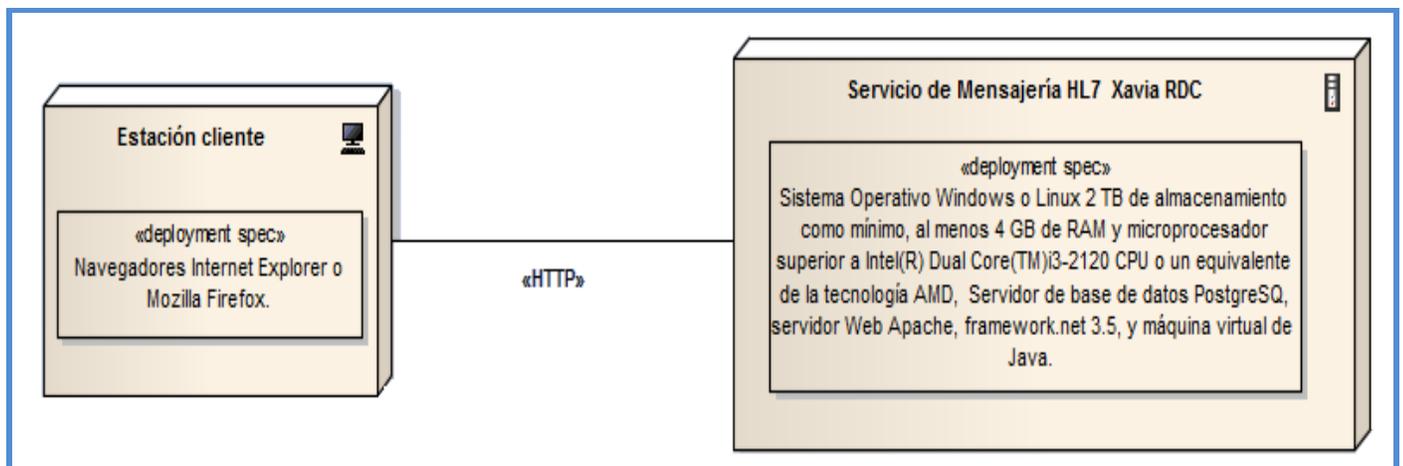
servidor, ip donde se encuentra, usuario y contraseña de dicho servidor de Base de Datos. El realcopbackup.bat es el archivo encargado de almacenar las copias de seguridad para un backup y contiene todos los comandos necesarios para realizar la copia; el realcopotraBD.bat es el que guarda las copias para otra Base de Datos y el restcoparchivo.bat es el que almacena la copia para un archivo

El archivo gestlogseguridad.txt guarda todo lo relacionado con el cifrado de la Base de Datos, genlogerrores.txt almacena toda la información referente a la ocurrencia de algún error y genlogoperacione.txt contiene un registro de todas las operaciones realizadas directamente con la Base de Datos.

### 3.2.2 Diagrama de despliegue

Un diagrama de despliegue muestra las relaciones físicas entre los componentes hardware y software en el sistema final, es decir, la configuración de los elementos de procesamiento en tiempo de ejecución y los componentes de software (procesos y objetos que se ejecutan en ellos). Se representa mediante un grafo de nodos unidos por conexiones de comunicación. Un nodo es un objeto físico en tiempo de ejecución que representa un recurso computacional. (45)

En la realización del diagrama de despliegue no se especificaron características en la estación cliente ya que esto depende de los requisitos de cada sistema externo en particular.



**Fig. 12** Diagrama de Despliegue

### 3.3 Tratamiento de errores

Durante la ejecución de un sistema pueden fracasar diferentes rutinas conocidas como excepciones. Una excepción es la indicación de que ocurrió un problema durante la ejecución de un programa. El tratamiento de excepciones permite restaurar un estado en el que la rutina pueda continuar con la ejecución, lo cual permite obtener un sistema más robusto y fiable. El tratamiento de errores en el componente propuesto se realiza en fragmentos de código donde puedan surgir situaciones inesperadas, donde se ejecutan sentencias que manipulan los datos que acceden hacia y hasta la base de datos, así como los errores que puedan surgir durante la validación de datos provenientes de la interfaz de usuario.

Para el manejo de todos ellos se utiliza el bloque try catch de la siguiente forma:

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

En el bloque try se escribirá el código para detectar cuando ocurra algún fallo y en el catch (capturar) como su nombre lo indica, se capturarán y manejarán dichas excepciones, mostrando mensajes en la interfaz de usuario mediante el componente del framework Seam: FacesMessages o realizando acciones que eliminen el problema internamente sin que el usuario se entere.

### 3.4 Estrategias de codificación. Estándares y estilos a utilizar

En la implementación de un sistema informático, la estandarización de la codificación es muy importante ya que a casi ningún software lo mantiene toda su vida el autor original. Las convenciones de código o estándares de codificación mejoran la lectura del software, permitiendo entender código nuevo mucho más rápido y más a fondo. El 80% del coste del código de un programa va a su mantenimiento.

Si se aplica de forma continuada un estándar de codificación bien definido, se utilizan técnicas de programación apropiadas, y, posteriormente, se efectúan revisiones del código de rutinas, caben muchas posibilidades de que un proyecto de software se convierta en un sistema de software fácil de comprender y de mantener.(46)

Teniendo en cuenta esta idea a continuación se plantean una serie de estrategias de codificación a utilizar para la implementación del Repositorio de Datos Clínicos:

### **Idioma**

Se utiliza el idioma español para la escritura del código y las palabras no se acentuarán.

### **Indentación y longitud de la línea**

Se deben emplear cuatro espacios como unidad de indentación. La construcción exacta de la indentación (espacios en blanco contra tabuladores) no se especifica. Los tabuladores deben ser exactamente cada ocho espacios.

### **Comentarios**

Los programas Java pueden tener dos tipos de comentarios: comentarios de implementación y comentarios de documentación. Los primeros son para comentarios acerca de una implementación particular. Los comentarios de documentación son para describir la especificación del código, libre de una perspectiva de implementación, y para ser leídos por desarrolladores que pueden no tener el código fuente a mano.

Se deben usar los comentarios para dar descripciones de código y facilitar información adicional que no es legible en el código mismo. Los comentarios deben contener sólo información que es relevante para la lectura y entendimiento del programa. No deben encerrarse en grandes cuadrados dibujados con asteriscos u otros caracteres y nunca deben incluir caracteres especiales como backspace(47).

### **Variables**

#### ***Declaración***

Se recomienda una declaración por línea, de esta forma se facilitan los comentarios a las mismas. No se deben poner diferentes tipos en la misma línea.

### ***Inicialización***

Las variables deben ser inicializadas en el lugar donde se declaran. Solo no se cumplirá esta consideración si el valor inicial de la misma depende de algún cálculo que deba ocurrir.

### **Clases e Interfaces**

#### ***Declaración***

Para la codificación de clases e interfaces de Java se deben seguir las siguientes reglas de formato:

- No debe existir ningún espacio en blanco entre el nombre de un método y el paréntesis que abre su lista de parámetros.
- La llave de apertura debe aparecer al final de la misma línea de la sentencia declaración.
- La llave de cierre debe empezar una nueva línea indentada para ajustarse a su sentencia de apertura correspondiente, excepto cuando no existen sentencias entre ambas, que debe aparecer inmediatamente después de la de apertura.
- Los métodos se separan con una línea en blanco.

#### ***Líneas en blanco***

Las líneas en blanco mejoran la facilidad de lectura separando secciones de código que están lógicamente relacionadas. Se deben usar siempre dos líneas en blanco entre las secciones de un fichero fuente o entre las definiciones de clases e interfaces. Se usará solo una línea en blanco entre métodos, entre las variables locales de un método y su primera sentencia, antes de un comentario de bloque o de un comentario de una línea o entre las distintas secciones lógicas de un método para facilitar la lectura.(47)

#### ***Espacios en blanco***

Se deben usar espacios en blanco en las siguientes circunstancias:

- Una palabra clave del lenguaje seguida por un paréntesis debe separarse por un espacio.
- Debe aparecer un espacio en blanco después de cada coma en las listas de argumentos.

- Todos los operadores binarios excepto "." se deben separar de sus operandos con espacios en blanco. Los espacios en blanco no deben separar los operadores unarios, incremento ("++") y decremento ("--") de sus operandos.(47)

## Convenciones de nombres

### **Clases**

Los nombres de las clases deben ser sustantivos, cuando son compuestos tendrán la primera letra de cada palabra que lo forma en mayúsculas. Se deben evitar acrónimos y abreviaturas.

### **Métodos**

Los métodos deben ser verbos, cuando son compuestos tendrán la primera letra en minúscula, y la primera letra de las siguientes palabras que lo forman en mayúscula.

### **Variables**

Todas las instancias y variables de clase o método empezarán con minúscula. Las palabras internas que lo forman (si son compuestas) empiezan con su primera letra en mayúsculas. Los nombres de variables de un solo carácter se deben evitar, excepto para variables índices temporales.(47)

## 3.5 Seguridad informática

Garantizar que los recursos informáticos de una compañía estén disponibles para cumplir sus propósitos, es decir, que no estén dañados o alterados por circunstancias o factores externos, es una definición útil para conocer lo que implica el concepto de seguridad informática. En términos generales, la seguridad informática puede entenderse como aquellas reglas técnicas y/o actividades destinadas a prevenir, proteger y resguardar lo que es considerado como susceptible de robo, pérdida o daño, ya sea de manera personal, grupal o empresarial (48). En este sentido, es la información el elemento principal a proteger, resguardar y recuperar dentro de las redes empresariales.

Todos los componentes de un sistema informático están expuestos a un ataque (hardware, software y datos), de datos o información ya que son los sujetos principales de protección de las técnicas de seguridad. La seguridad informática se dedica principalmente a proteger la confidencialidad, la integridad y

disponibilidad de la información. En el Repositorio de Datos Clínicos estos objetivos se garantizan de la siguiente forma:

- **Integridad:** La información solo puede ser modificada por los sistemas externos con accesos al repositorio. En el RDC la información no es eliminada, solo se inserta y modifica por las personas autorizadas.
- **Confidencialidad:** La información solo debe ser accedida por los sistemas externos con acceso al repositorio. El RDC no garantiza la confidencialidad de la información, guardará todo lo que se le envíe desde los sistemas asociados al mismo. Depende de los sistemas externos garantizar que accedan al repositorio las personas autorizadas mediante roles y permisos.
- **Disponibilidad:** El RDC garantiza que los sistemas autorizados accedan a la información, de lo contrario escuchará la petición pero no dará respuesta.
- **No Repudio:** El RDC guardará en un fichero los datos asociados al sistema que introdujo la información.
- **Control de Acceso:** El RDC no define el nivel de acceso de las personas autorizadas, es responsabilidad del sistema externo especificar los permisos necesarios mediante roles.

### 3.6 Análisis de resultados

La validación de sistemas informáticos se realiza para medir hasta qué punto es correcta la implementación del sistema y que este cumple con las necesidades y los requisitos del usuario. Se ejecutan los casos de prueba para verificar el correcto funcionamiento de los componentes implementados, se hace un análisis de los resultados obtenidos y se resuelven las no conformidades detectadas.

#### 3.6.1 Pruebas del Software

##### Pruebas de caja negra

Para la realización de estas pruebas se diseñaron casos de prueba por cada caso de uso basados en la técnica de partición de equivalencia. Con estos casos de prueba se verificó que las funciones del software son operativas, que las entradas se acepten de forma adecuada y se produzca un resultado correcto, teniendo en cuenta que la integridad de la información externa se mantenga. Para la confección de cada caso de prueba se utilizó un Excel que consta de tres hojas de cálculo: Presentación, Variables y Caso de Prueba.

**Presentación:** Es la hoja de cálculo donde se encuentra la presentación del caso de prueba, en ella aparece el nombre del centro de desarrollo y el proyecto al que pertenece el software a probar. También cuenta con dos tablas una para el control del documento y otra para el control de cambio. A continuación se describen los campos que presentan dichas tablas.

Tabla Control del documento:

- Elaborado por: Campo donde se describe el nombre de la persona que confeccionó el caso de prueba.
- Cargo: El cargo que desempeña en el proyecto la persona que confeccionó el caso de prueba.
- Aprobado por: Nombre completo de la persona que revisó y aprobó el caso de prueba.
- Cargo: Cargo de la persona que revisó y aprobó el caso de prueba.
- Firma: De la persona que revisó y aprobó el caso de prueba.
- Fecha: Fecha de revisión y aprobación del caso de prueba.

Tabla Control de cambio:

- Versión: Versión del caso de prueba.
- Sección, figura, tabla:
- Tipo: Indicar el tipo de cambio: A Alta; B Baja; M Modificación.
- Fecha: Fecha del cambio.
- Autor del cambio: Nombre de la persona que realizó el cambio.
- Descripción del cambio: Donde se describe el cambio que se realizó.

**Variables:** Es la hoja de cálculo donde se definen las variables del caso de prueba, se entiende como una variable a cada uno de los campos de entrada que contiene la interfaz del sistema correspondiente al caso

de uso, para realizar la descripción de las variables se confecciona una tabla que contiene los siguientes campos:

- No: Número de la variable, el cual se denota como Vn.
- Nombre del campo: Nombre del campo en la interfaz del sistema.
- Clasificación: Clasificación del campo (campo de texto, lista desplegable, campo booleano, etc.)
- Valor nulo: En este campo solo se pone Si o No en dependencia de si el campo de entrada del caso de uso puede estar vacío en algún momento.
- Descripción: Aquí se hace una breve descripción del campo donde se ponen las restricciones que debe tener el campo de entrada (por ejemplo si el campo puede admitir solo texto o solo número en caso de ser un campo de texto).

**Caso de Prueba:** Es la hoja de cálculo donde se confecciona el caso de prueba, en ella aparece una descripción general del caso de uso al que se le va a realizar el caso de prueba, la condiciones de ejecución que son necesaria para que se ejecute el caso de uso y la tabla SC (SC de sección y se realiza una tabla por cada sección del caso de uso) la cual consta con los siguientes campos:

**Escenario:** Los escenarios son las condiciones bajo las que se prueba el software con el fin de determinar si la respuesta que da el sistema es la correcta o si existe algún error, es decir en caso de que los datos entrados sean los correctos, en el caso de que existan datos incompletos y en el caso de que existan datos incorrectos.

**Descripción:** Es donde se hace una breve descripción del escenario.

**Variable 1...n:** Son las variables que fueron definidas anteriormente en la hoja de cálculo Variable, estas son probadas en cada uno de los escenarios

**Respuesta del sistema:** Es donde se describe que respuesta debe dar el sistema cuando se ejecuta cada clase de equivalencia.

**Flujo central:** También conocido como las migas de pan, marcan el camino que debe recorrerse para poder ejecutar la sección del caso de uso que se desea probar.

### **Pruebas de caja negra**

Luego de la implementación del Repositorio de Datos Clínicos, al sistema se le efectuaron un conjunto de pruebas de caja negra por cada caso de uso, las mismas se encuentra en la siguiente dirección.

**(Remitirse a: Expediente. Proyecto\1. ingeniería\1.2 implementación y prueba, al documento CDR\_TIE\_0111\_Diseño de Casos de Prueba basado en Casos de Uso.)**

Como resultado de este capítulo se representaron los elementos necesarios para la comprensión del software implementado, como el diagrama de componente y el de despliegue. Además se hace una descripción de los estándares de codificación utilizados para la implementación del sistema y la utilización del tratamiento de errores. Se ejecutaron los casos de prueba para verificar el correcto funcionamiento de los componentes implementados.

## CONCLUSIONES

La realización del presente Trabajo de Diploma ha posibilitado cumplir con el objetivo definido en el diseño de la investigación, por lo que se han arribado a las siguientes conclusiones:

1. Se realizó un estudio de los sistemas informáticos que cuentan con Repositorios de Datos Clínicos a nivel nacional e internacional, sin embargo no proporcionaban una solución completa al problema de la investigación, pero sí conocimientos que contribuyen a solventar el mismo.
2. El estudio realizado sobre metodologías, lenguajes y herramientas determinó la selección de las utilizadas en este trabajo.
3. El modelo del diseño permitió generar los elementos necesarios para desarrollar la implementación del sistema, al concebir una representación técnica del mismo.
4. Se desarrolló mediante la metodología definida en el CESIM, los artefactos generados en los flujos de trabajo de “Modelado de Negocio”, “Administración de Requisitos”, “Análisis y Diseño” e “Implementación”.
5. Se desarrolló un Repositorio de Datos Clínicos que cumple con los requisitos definidos por el cliente, lográndose así el objetivo propuesto.

## RECOMENDACIONES

Luego de culminada la investigación y cumplido el objetivo general de la misma, con vistas a mejorar el funcionamiento del sistema y con el fin que se incrementen las funcionalidades brindadas para un mejor uso de éste, se recomienda:

- Adicionar a la solución las versiones v 2.6 y 3.x de mensajes HL7.
- Analizar periódicamente las estructuras y estándares internacionales relacionados con el almacenamiento de información en HCE para determinar nuevos datos que no esté incluidos en el repositorio y así agregarlos al mismo.
- Realizar encuestas periódicas sobre la efectividad del repositorio, el grado de satisfacción de los sistemas externos, así como funcionalidades que se le pueden incorporar. Todo esto previendo un futuro perfeccionamiento del sistema.

## REFERENCIAS BIBLIOGRÁFICAS

1. Propuesta de análisis para el sistema de información radiológica. [En línea] 2007. [http://www.ucmh.sld.cu/rhab/editorial\\_rev10.htm..](http://www.ucmh.sld.cu/rhab/editorial_rev10.htm..)
2. **Pilar, Dr. Javier Gómez.** Impacto, ventajas e inconvenientes de la Historia Clínica Electrónica. [En línea] <http://www.tel.uva.es/personales/jgompil/ctmiii/HCE.pdf>.
3. Repositorio de Datos. [En línea] [http://www.conganat.org/SEIS/informed02/INFORMED2002\\_romerorc.pdf..](http://www.conganat.org/SEIS/informed02/INFORMED2002_romerorc.pdf..)
4. Creada en la Sección UCI de la Sociedad Cubana de Informática Médica . [En línea] <http://www.uci.cu/creada-la-seccion-uci-de-la-sociedad-cubana-de-informatica-medica>.
5. Kanteron system. [En línea] <http://www.kaneron.com/blog/es/products/kaneron-his/>.
6. Kanteron system. [En línea] <http://www.kaneron.com/blog/es/products/kaneron-ris/>.
7. Metamodel Web análisis de desarrollo de software. [En línea] <http://metamodel.wordpress.com/2011/03/04/concepto-de-interoperabilidad/>.
8. Definición de Clínica. [En línea] <http://www.definicionabc.com/salud/clinica.php>.
9. Informacion clinica. [En línea] <http://www.cancer.gov/diccionario?cdrid=44007> .
10. MasterMagazine. [En línea] <http://www.mastermagazine.info/termino/6534.php>.
11. **Serrano, Ramón Carlos Romero.** REPOSITORIO DE DATOS. [En línea] [http://www.conganat.org/SEIS/informed02/INFORMED2002\\_romerorc.pdf](http://www.conganat.org/SEIS/informed02/INFORMED2002_romerorc.pdf).
12. Repositorio Clínica Hitachy. [En línea] <http://www.agenciasrelacionespublicas.com/articles/article-4893.htm..>
13. Estación de Trabajo Clínico Regional (CWS) y Repositorio de Datos Clínicos (CDR). [En línea] <http://www.ithealthboard.health.nz/content/regional-clinical-workstation-cws-and-clinical-data-repository-cdr..>
14. Repositorio de Datos Clínicos del Departamento de Defensa para asuntos de Veteranos. [En línea] <http://dhims.health.mil/products/data-sharing/chdr.aspx..>
15. Repositorio de Datos Clínicos de la Universidad de Medicina en Virginia. [En línea] [http://www.medicine.virginia.edu/..](http://www.medicine.virginia.edu/)
16. Sistemas para archivo y comunicación de imágenes (PACS). [En línea] [http://www.cenetec.salud.gob.mx/descargas/equipo\\_guias/guias\\_tec/41gt\\_PACS.pdf..](http://www.cenetec.salud.gob.mx/descargas/equipo_guias/guias_tec/41gt_PACS.pdf..)

17. Propósito de informes hospital hermanos almejeiras. [En línea] C:\Users\lsmayr.lsmayr-PC\Desktop\Revista Cubana de Salud Pública - Dual purpose of the hospital discharge report in 'Hermanos Ameijeiras' hospital.htm. ISSN 0864-3466.
18. Información médica y estándares. [En línea] <http://informatica-medica.blogspot.com/>.
19. Health Level Seven International . [En línea] <http://www.hl7.org/>.
20. Documentos clínicos electrónicos HL7 CDA R2 - Estructura general . [En línea] <http://hl7es.blogspot.com/2011/04/documentos-clinicos-electronicos-hl7.html>.
21. El Proceso Unificado de Desarrollo de Software. [En línea] <http://www.chaco.gov.ar/utn/disenodesistemas/apuntes/oo/ApunteRUP.pdf>.
22. Lenguaje Unificado de Modelado (UML) . [En línea] <http://www.disca.upv.es/enheror/pdf/ActaUML.PDF>.
23. Enterprise Architect. [En línea] [http://www.sparxsystems.com/products/ea/7.5/..](http://www.sparxsystems.com/products/ea/7.5/)
24. Java. [En línea] [http://www.java.com/es/download/whatis\\_java.jsp..](http://www.java.com/es/download/whatis_java.jsp..)
25. Entorno de desarrollo integrado. [En línea] <http://carlosblanco.pro/2012/04/entornos-desarrollo-integrado-introduccion/>.
26. PostgreSQL. [En línea] [http://www.postgresql.org/about/news/1108/..](http://www.postgresql.org/about/news/1108/)
27. Jboss server. [En línea] <http://www.alferez.es/documentos/Jboss.pdf>.
28. **Montano Guevara, C. A. y R. R., Robreto.** Implementación de las hojas de anestesia estándar y cardiovascular del Quirófano del Sistema de Información Hospitalaria alas HIS, Universidad de las Ciencias Informáticas. [En línea] JULIO de 2010.
29. Enterprise JavaBeans 3.0. [En línea] <http://www.oracle.com/technetwork/java/javaee/ejb/index.html>.
30. Tecnología Java. [En línea] <http://www.slideshare.net/Dryope/tecnologia-java-420996..>
31. Actualización y avances de la investigación. [En línea] [http://www.femisaluddigital.org.uy/admin/files/femi/from\\_old\\_website..](http://www.femisaluddigital.org.uy/admin/files/femi/from_old_website..)
32. Novedades en Hibernate 3.3 . [En línea] <http://www.dosideas.com/noticias/java/176-novedades-en-hibernate-33.html>.
33. Oracle. [En línea] <http://www.oracle.com/technetwork/java/index-jsp-140203.html..>
34. **Juan Alonso Ramos.** Autentia Soporte de desarrollo informático. [En línea] <http://adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=Ajax4Jsf>.

35. Manipulación MLLP (Minimal Lower Layer Protocol). [En línea] <http://publib.boulder.ibm.com/infocenter/wmbhelp/v7r0m0/index.jsp?topic=%2Fcom.ibm.etools.mft.samples.healthcare.doc%2Fdoc%2Fmllpmanipulation.htm>.
36. Modelo de dominio. [En línea] <http://lsi.ugr.es/~ig1/isoo/larman/Modelo%20del%20dominio.pdf>.
37. Especificación de Requerimientos del Software. [En línea] [http://merinde.net/index.php?option=com\\_content&task=view&id=475&Itemid=291](http://merinde.net/index.php?option=com_content&task=view&id=475&Itemid=291).
38. Caso de usos. [En línea] [http://biblioteca.clacso.edu.ar/archivos\\_web\\_adj/148.pdf](http://biblioteca.clacso.edu.ar/archivos_web_adj/148.pdf).
39. **Juan Pelaez**. Arquitectura basada en capas. [En línea] <http://www.juanpelaez.com/geek-stuff/arquitectura/arquitectura-basada-en-capas/>.
40. Diagrama de clases. [En línea] [http://www-2.dc.uba.ar/materias/isoft1/is1-2005\\_2/apuntes/SlidesDC.pdf](http://www-2.dc.uba.ar/materias/isoft1/is1-2005_2/apuntes/SlidesDC.pdf).
41. Diagrama de Interacción\_ UML. [En línea] <http://www.slideshare.net/jlrvpuma/diagramas-de-interaccion..>
42. Modelo de Datos 1ª parte. [En línea] [http://blogs.ua.es/mu171m3d14/2011/04/17/modelo-de-datos-1%C2%AA-parte/..](http://blogs.ua.es/mu171m3d14/2011/04/17/modelo-de-datos-1%C2%AA-parte/)
43. **Brites, Walter Fernando**. Los modelos de implementación. [En línea] <http://www.eumed.net/libros-gratis/2010e/811/modelos%20de%20implementacion.htm>.
44. Conceptos en un Diagrama de Implementación. [En línea] <http://webdocs.cs.ualberta.ca/~pfiguero/soo/uml/implementacion01.html>.
45. Módulo 2. Modelo de implementación. . [En línea] <http://www.slideshare.net/joshell/diagramas-uml-componentes-y-despliegue>.
46. **Cordova Eireos, Lixandra y Benítez Ortiz, Leandro**. *Desarrollo de los reportes del Módulo Admisión del Sistema de Información Hospitalaria alas HIS*. La Habana : s.n., JULIO 2011.
47. **Yaneisi Betancourt Vinent, Yasniel Pérez Acosta**. *Componente web para Programas de enfermedades transmisibles del Sistema Integral para la Atención Primaria de Salud*. La Habana : s.n., 2012.
48. Seguridad Informática. [En línea] <http://definicion.de/seguridad-informatica/>.

**BIBLIOGRAFÍA**

1. **Brites, Walter Fernando.** Los modelos de implementación. [En línea] <http://www.eumed.net/libros-gratis/2010e/811/modelos%20de%20implementacion.htm>
2. Caso de usos. [En línea] [http://biblioteca.clacso.edu.ar/archivos\\_web\\_adj/148.pdf](http://biblioteca.clacso.edu.ar/archivos_web_adj/148.pdf)
3. Conceptos en un Diagrama de Implementación. [En línea] <http://webdocs.cs.ualberta.ca/~pfiguero/soo/uml/implementacion01.html>.
4. **Cordova Eireos, Lixandra y Benítez Ortiz, Leandro.** *Desarrollo de los reportes del Módulo Admisión del Sistema de Información Hospitalaria alas HIS.* La Habana : s.n., JULIO 2011.
5. Diagrama de clases. [En línea] [http://www-2.dc.uba.ar/materias/isoft1/is1-2005\\_2/apuntes/SlidesDC.pdf](http://www-2.dc.uba.ar/materias/isoft1/is1-2005_2/apuntes/SlidesDC.pdf).
6. Diagrama de Interacción\_ UML. [En línea] <http://www.slideshare.net/jlrvpuma/diagramas-de-interaccion>
7. El Proceso Unificado de Desarrollo de Software. [En línea] <http://www.chaco.gov.ar/utn/disenodesistemas/apuntes/oo/ApunteRUP.pdf>.
8. Enterprise Architect. [En línea] <http://www.sparxsystems.com/products/ea/7.5/>.
9. Enterprise JavaBeans 3.0. [En línea] <http://www.oracle.com/technetwork/java/javaee/ejb/index.html>.
10. Health Level Seven International . [En línea] <http://www.hl7.org/>.
11. Infraestructura de Perfiles IHE. Infraestructura de Perfiles IHE. [En línea] HYPERLINK [http://wiki.ihe.net/index.php?title=Profiles#IHE\\_IT\\_Infrastructure\\_Profiles](http://wiki.ihe.net/index.php?title=Profiles#IHE_IT_Infrastructure_Profiles)
12. Jboss server. [En línea] <http://www.alferez.es/documentos/Jboss.pdf>.
13. Lenguaje Unificado de Modelado (UML) . [En línea] <http://www.disca.upv.es/enheror/pdf/ActaUML.PDF>.
14. Los estilos y estrategias de aprendizaje en estudiantes de arquitectura de una universidad privada. [Revisado el 10 de abril de 2013]: Disponible en <http://www.humus.pro.br/201123852.pdf>
15. Modelo de Datos 1ª parte. [En línea] <http://blogs.ua.es/mu171m3d14/2011/04/17/modelo-de-datos-1%C2%AA-parte/>.
16. Modelo de dominio. [En línea] <http://lsi.ugr.es/~ig1/isoo/larman/Modelo%20del%20dominio.pdf>.
17. Módulo 2. Modelo de implementación. . [En línea] <http://www.slideshare.net/joshell/diagramas-uml-componentes-y-despliegue>.
18. **Montano Guevara, C. A. y. R. R., Robreto.** Implementación de las hojas de anestesia estándar y cardiovascular del Quirófano del Sistema de Información Hospitalaria alas HIS, Universidad de las Ciencias Informáticas. [En línea] JULIO de 2010.

19. **Pilar, Dr. Javier Gómez.** Impacto, ventajas e inconvenientes de la Historia Clínica Electrónica. [En línea] <http://www.tel.uva.es/personales/jgompil/ctmiii/HCE.pdf>.
20. Propósito de informes hospital hermanos almejeiras. [En línea] C:\Users\lsmayr.lsmayr-PC\Desktop\Revista Cubana de Salud Pública - Dual purpose of the hospital discharge report in 'Hermanos Ameijeiras' hospital.htm. ISSN 0864-3466.
21. **Serrano, Ramón Carlos Romero.** REPOSITORIO DE DATOS. [En línea] [http://www.conganat.org/SEIS/informed02/INFORMED2002\\_romerorc.pdf](http://www.conganat.org/SEIS/informed02/INFORMED2002_romerorc.pdf).
22. Sistemas para archivo y comunicación de imágenes (PACS). [En línea] [http://www.cenetec.salud.gob.mx/descargas/equipo\\_guias/guias\\_tec/41gt\\_PACS.pdf](http://www.cenetec.salud.gob.mx/descargas/equipo_guias/guias_tec/41gt_PACS.pdf).
23. Tecnología Java. [En línea] <http://www.slideshare.net/Dryope/tecnologia-java-420996>.
24. **Yaneisi Betancourt Vinent, Yasniel Pérez Acosta.** *Componente web para Programas de enfermedades transmisibles del Sistema Integral para la Atención Primaria de Salud*. La Habana : s.n., 2012.

**WEBGRAFÍA**

1. Actualización y avances de la investigación. [En línea]  
[http://www.femisaluddigital.org.uy/admin/files/femi/from\\_old\\_website](http://www.femisaluddigital.org.uy/admin/files/femi/from_old_website)
2. Creada en la Sección UCI de la Sociedad Cubana de Informática Médica . [En línea]  
<http://www.uci.cu/creada-la-seccion-uci-de-la-sociedad-cubana-de-informatica-medica>.
3. Definición de Clínica. [En línea] <http://www.definicionabc.com/salud/clinica.php>.
4. Documentos clínicos electrónicos HL7 CDA R2 - Estructura general . [En línea]  
<http://hl7es.blogspot.com/2011/04/documentos-clinicos-electronicos-hl7.html>.
5. Entorno de desarrollo integrado. [En línea] <http://carlosblanco.pro/2012/04/entornos-desarrollo-integrado-introduccion/>.
6. Especificación de Requerimientos del Software. [En línea]  
[http://merinde.net/index.php?option=com\\_content&task=view&id=475&Itemid=291](http://merinde.net/index.php?option=com_content&task=view&id=475&Itemid=291).
7. Estación de Trabajo Clínico Regional (CWS) y Repositorio de Datos Clínicos (CDR). [En línea]  
<http://www.ithealthboard.health.nz/content/regional-clinical-workstation-cws-and-clinical-data-repository-cdr>.
8. Información clínica. [En línea] <http://www.cancer.gov/diccionario?cdrid=44007> .
9. Información médica y estándares. [En línea] <http://informatica-medica.blogspot.com/>.
10. Java. [En línea] [http://www.java.com/es/download/whatis\\_java.jsp](http://www.java.com/es/download/whatis_java.jsp)
11. **Juan Alonso Ramos.** Autentia Soporte de desarrollo informático. [En línea]  
<http://adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=Ajax4Jsf>.
12. **Juan Pelaez.** Arquitectura basada en capas. [En línea] <http://www.juanpelaez.com/geek-stuff/arquitectura/arquitectura-basada-en-capas/>
13. Kanteron system. [En línea] <http://www.kaneron.com/blog/es/products/kaneron-his/>.
14. Kanteron system. [En línea] <http://www.kaneron.com/blog/es/products/kaneron-ris/>.
15. Manipulación MLLP (Minimal Lower Layer Protocol). [En línea]  
<http://publib.boulder.ibm.com/infocenter/wmbhelp/v7r0m0/index.jsp?topic=%2Fcom.ibm.etools.mft.samples.healthcare.doc%2Fdoc%2Fmllpmanipulation.htm>.
16. MasterMagazine. [En línea] <http://www.mastermagazine.info/termino/6534.php>.
17. Metamodel Web análisis de desarrollo de software. [En línea]  
<http://metamodel.wordpress.com/2011/03/04/concepto-de-interoperabilidad/>.

18. Novedades en Hibernate 3.3 . [En línea] <http://www.dosideas.com/noticias/java/176-novedades-en-hibernate-33.html>.
19. Oracle. [En línea] <http://www.oracle.com/technetwork/java/index-jsp-140203.html>.
20. PostgreSQL. [En línea] <http://www.postgresql.org/about/news/1108/>.
21. Propuesta de análisis para el sistema de información radiológica. [En línea] 2007. [http://www.ucmh.sld.cu/rhab/editorial\\_rev10.htm](http://www.ucmh.sld.cu/rhab/editorial_rev10.htm).
22. Repositorio Clínica Hitachy. [En línea] <http://www.agenciasrelacionespublicas.com/articles/article-4893.htm>
23. Repositorio de Datos. [En línea] [http://www.conganat.org/SEIS/informed02/INFORMED2002\\_romerorc.pdf](http://www.conganat.org/SEIS/informed02/INFORMED2002_romerorc.pdf).
24. Repositorio de Datos Clínicos de la Universidad de Medicina en Virginia. [En línea] <http://www.medicine.virginia.edu/>.
25. Repositorio de Datos Clínicos del Departamento de Defensa para asuntos de Veteranos. [En línea] <http://dhims.health.mil/products/data-sharing/chdr.aspx>.
26. Seguridad Informática. [En línea] <http://definicion.de/seguridad-informatica/>

## GLOSARIO DE TÉRMINOS

**AJAX:** Técnica para el desarrollo Web que posibilita la creación de aplicaciones interactivas.

**API:** (Application Programming Interface): Conjunto de funciones y procedimientos que poseen algunas librerías con el objetivo de ser utilizadas por otro software como una capa de abstracción.

**Estándar:** Que sirve como tipo, modelo, norma, patrón o referencia por ser corriente.

**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.

**Java EE, Java Platform, Enterprise Edition o Java EE** (anteriormente conocido como Java 2 Platform, Enterprise Edition o J2EE): Es una plataforma de programación, parte de la Plataforma Java, para desarrollar y ejecutar software de aplicaciones en lenguaje de programación Java con arquitectura de N capas distribuidas y que se apoya ampliamente en componentes de software modulares ejecutándose sobre un servidor de aplicaciones.

**JavaBeans:** Modelo de componentes para la construcción de aplicaciones en Java. Se usan para encapsular varios objetos en un único objeto. Constituyen componentes de software reutilizables que se puedan manipular visualmente en una herramienta de construcción.

**JDBC** (Java Database Connectivity): Es un API para trabajar con bases de datos desde Java, independientemente de la base de datos a la que se acceda.

**JSP** (JavaServer Pages): Tecnología Java que permite generar contenido dinámico para web, en forma de documentos HTML, XML o de otro tipo.

**Open Source** (Código abierto): Término con el que se conoce al software distribuido y desarrollado libremente.

**POJO** (Plain Old Java Object): Enfatiza el uso de clases simples y que no dependen de un framework en especial.

**XML** (eXtensible Markup Language o Lenguaje de Marcas Extensibles): Es un metalenguaje extensible de etiquetas que permite definir la gramática de lenguajes específicos. No constituye realmente un lenguaje en particular, sino una manera de definir lenguajes para diferentes necesidades.

**HIS**: Sistema de Información Hospitalaria. Son sistemas de software que trabajan acoplados y permiten la informatización de los distintos servicios de las instituciones de salud.

**PACS**: Picture Archiving and Communication System (PACS). Sistema para el almacenamiento y comunicación de imágenes médicas.

**RIS**: Sistema de Información Radiológica (Radiological Information System). Es un sistema encargado de la gestión de la información generada y manipulada como resultado de los procesos de negocio de carácter radiológico (imagenológico).