



UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS

FACULTAD 7

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

Componente de Consulta Clínica del subsistema web del Sistema
Integral para la Atención Primaria de Salud

Autores:

Marisleivy Peláez Hernández.

Humberto Ruiz Salcedo.

Tutores:

Ing. Johander León Garcés.

Ing. Yosleiby Izquierdo Sánchez.

Ciudad de La Habana, Julio de 2010

“Año 52 del Triunfo de la Revolución”

DATOS DE CONTACTO

Johander León Garcés (jleong@uci.cu): Graduado de Ingeniero en Ciencias Informáticas en el año 2007 con Título de Oro. Actualmente labora en la Universidad de Ciencias Informáticas (UCI), desempeñándose como profesor de la Facultad No. 7 vinculado a la producción y como miembro de la reserva del Comandante en Jefe. Ha participado en diferentes eventos científicos del centro obteniendo resultados relevantes. Actualmente es parte del proyecto Sistema Integral para la Atención Primaria de Salud (SIAPS) donde se desarrolla como jefe de departamento del Centro de Informática Médica (CESIM). Ha cursado varios cursos de postgrado como parte de su superación profesional, así como un diplomado de Líder de Proyecto.

Yosleiby Izquierdo Sánchez (yizquierdo@uci.cu): Graduado de Ingeniero en Ciencias Informáticas en el año 2008. Actualmente labora en la Universidad de Ciencias Informáticas (UCI), desempeñándose como profesor de la Facultad No. 7 vinculado a la producción y como miembro de la reserva del Comandante en Jefe. Actualmente es parte del proyecto Sistema Integral para la Atención Primaria de Salud (SIAPS) donde se desarrolla como jefe de Equipo de Desarrollo. Ha cursado varios cursos de postgrado como parte de su superación profesional.

Resumen

La gestión de la información generada durante las consultas clínicas de Atención Primaria de Salud constituye el eslabón fundamental para lograr una atención de mayor calidad a cada uno de los individuos de la sociedad y sirve de base para la toma de decisiones y para conformar la Historia Clínica Individual de los pacientes. Esta información, bien gestionada permite a los especialistas de los diferentes niveles de atención de salud contar con más elementos a la hora de emitir diagnósticos y tratamientos efectivos, además de contribuir a la obtención de estadísticas médicas y al desarrollo de investigaciones.

Actualmente no existe ningún sistema informático que lleve todo el control de la información que se genera en dichas consultas. Por estas razones se propone el desarrollo del Componente de Consulta Clínica basado en tecnologías libres y de código abierto.

Para el desarrollo del sistema se usa como plataforma de desarrollo Java Enterprise Edition 5.0, Java como lenguaje de programación, como gestor de base de datos PostgreSQL y los frameworks SEAM, EJB, Hibernate, JSF, RichFaces. Se usó como notación para el modelado de procesos de negocio BPMN (Business Process Management Notation) y como lenguaje de modelado UML (Lenguaje Unificado de Modelado).

Con la culminación de la investigación se espera contar con un sistema que provea información legible y bien estructurada, un historial único de consultas por paciente, además de facilitar el control y seguimiento de los mismos; que sirva de base para la dispensarización de la población, facilitando la rápida obtención de estadísticas médicas y las posibilidades de desarrollar programas de prevención y promoción de salud.

Palabras Claves: Atención Primaria de la Salud, Historia Clínica Individual, Consultas Clínicas.

Tabla de Contenidos

Introducción	1
Capítulo 1. Fundamentación Teórica	6
1.1. Marco Conceptual	6
1.3. Estado del Arte.....	10
1.3.1. Análisis de Soluciones Existentes.....	14
1.4. Situación Problémica.....	16
1.5. Tecnologías	20
1.5.1. Java EE 5	20
1.5.2. Estilos Arquitectónicos.....	20
1.5.3. Lenguajes	21
1.5.4. Metodología de desarrollo	23
1.5.7. Frameworks.....	24
1.5.8. Herramientas	26
Capítulo 2. Características del Sistema	28
2.1. Modelo de Negocio	28
2.1.2. Procesos de negocio	29
2.1.3. Descripción de los Procesos.....	29
2.1.4. Diagramas de Procesos.....	30
2.2. Propuesta de Sistema	34
2.2.1. Especificación de Requerimientos de Software	35
2.2.1.1. Requisitos Funcionales.....	35
2.2.1.2. Requerimientos no funcionales.....	40
Capítulo 3. Diseño del Sistema	44
3.1. Diseño.....	44
3.2. Modelo de diseño.....	45
3.3. Definición de elementos del Diseño	45
3.4. Uso de Patrones	47
3.4.1. Patrones GRASP.....	47
3.5. Diagramas de clases del diseño.....	49
3.5.1. Descripción de las clases del diseño	52
3.6. Modelo de Datos	55
3.6.1. Descripción de tablas y atributos	55
Capítulo 4. Implementación	56
4.1 Propuesta de integración	56
4.2. Diagrama de despliegue	56
4.3 Tratamiento de Excepciones.....	59
4.4 Seguridad.....	60
4.5 Estándares de Codificación.....	62
Conclusiones	67
Recomendaciones	68
Referencias Bibliográficas.....	69
Bibliografía.....	73
Glosario de Términos.....	77

Introducción

Garantizar la calidad en la prevención, cuidado, rehabilitación de los pacientes y especializar los servicios de salud es el objetivo esencial por el cual se crearon las estructuras sanitarias denominadas: Sistema Nacional de Salud (SNS).

La forma y los métodos que sirven de base para la organización de la atención a la salud en un país determinado, es lo que se conoce como SNS. La Organización Mundial de la Salud (OMS) lo define como: “Un complejo de elementos interrelacionados que contribuyen a la salud en los hogares, lugares de trabajo, lugares públicos y comunidades, así como el medio ambiente físico y psicosocial en el sector de salud y otros sectores afines”. [1] El mismo comprende tres niveles de atención médica organizados en: Atención Médica Primaria (APS), Atención Médica Secundaria y Atención Médica Terciaria, siendo la APS el eslabón fundamental de esta cadena.

La APS, según la definición dada en la Declaración de Alma-Ata en 1978, convocada por la OMS es la siguiente: “La asistencia sanitaria esencial basada en métodos y tecnologías prácticos, científicamente fundados y socialmente aceptables, puesta al alcance de todos los individuos y familias de la comunidad mediante su plena participación y a un costo que la comunidad y el país puedan soportar, en todas y cada una de las etapas de su desarrollo con un espíritu de autorresponsabilidad y autodeterminación. La atención primaria forma parte integrante tanto del sistema nacional de salud, del que constituye la función central y el núcleo principal, como del desarrollo social y económico global de la comunidad”. [2]

Por otra parte, la evolución y el desarrollo de las tecnologías de la información han derivado la incorporación de aplicaciones informáticas en las instituciones para gestionar la información, con el objetivo de lograr una mayor dinámica en sus procesos de negocio. Son aplicables a múltiples esferas de la vida social. Las comunicaciones, los servicios y la salud son las áreas que más los demandan. Dichas tecnologías se han convertido en parte integrante del concepto moderno de salud pública y de las políticas sanitarias nacionales. Estos recursos pueden ayudar a las instituciones sanitarias a planear sus estrategias de promoción de salud y a comunicar mensajes claves relacionados con la salud pública.

Muchos teóricos denominan la época actual como sociedad de la información, sociedad del conocimiento, sociedad digital, etc. La base para denominarla así es la influencia del paradigma tecnológico imperante que ha ocasionado cambios en las relaciones humanas y en la forma de vida.

En la actualidad a nivel internacional existe una gran demanda de sistemas informáticos para la salud pues la necesidad de una mejor gestión en la información médica es imperativa para garantizar un diagnóstico más certero y una atención de calidad. Pero también muchas de las soluciones informáticas han fracasado, debido a malos usos de las mismas o a causas propias de mal funcionamiento y planificación, acarreando una notable desconfianza por parte del personal de salud encargado de su uso. Con el advenimiento de la informatización de los SNS surge como primera prioridad la informatización de la APS como eje de dicho proceso.

Los sistemas de información computarizados en la APS se desarrollaron electromecánicamente entre los años 1930-40, y electrónicamente desde 1974. La informática e informatización de la APS y MGI (Medicina general Integral) aparecen a partir de 1985, como apoyo al médico en el diagnóstico y la terapéutica clínica, y al programa de salud en la comunidad. [3]

La informatización de la APS, y en particular la ofrecida por el médico especialista, en muchas ocasiones se percibe como un problema simple que se resuelve solamente creando redes de computadoras personales y otras tecnologías de información y comunicación entre los consultorios y policlínicos. Sin embargo, en la realidad es un problema complejo que requiere aún de múltiples proyectos de investigación científica multidisciplinaria, de desarrollo tecnológico y de mucho intercambio académico científico-tecnológico. Esto se debe no a los grandes retos de carácter informático, sino también a los enormes desafíos médicos científico-tecnológicos de la propia naturaleza de los procesos de toma de decisiones a ese nivel de atención y de especialización. También, se debe la complejidad a la integración que requiere con otros niveles de atención y especialidades, para no duplicar esfuerzos, y además con la docencia, la investigación y la gerencia.

Los especialistas clínicos en las entidades de APS realizan actividades de gran envergadura por los resultados aportados a la población, puesto que ayudan en el diagnóstico temprano de enfermedades sin ser necesario que el paciente acuda a un centro especializado, brindándoles la posibilidad de recibir los servicios en su propio escenario social.

A nivel nacional e internacional existen carencias de sistemas informáticos relacionados con la gestión de la información de las especialidades clínicas en la APS, por lo que se ve afectada la comunicación entre las distintas especialidades imposibilitando al médico tener una visión más amplia de los estudios realizados al paciente en un entorno integral, donde cada especialista pueda tener acceso al historial clínico del paciente, contar con información médica actualizada y tener acceso a los resultados de los exámenes realizados de forma instantánea.

Todo lo anterior impide realizar diagnósticos certeros así como facilitar el tratamiento y seguimiento del paciente, disminuyendo la calidad de vida de los pacientes; ya que al no garantizar la oportunidad y consistencia de la información se dificulta la efectividad y eficiencia de los procesos relacionados con la salud, trayendo como consecuencia un decremento de la calidad en la atención médica.

Pero el principal problema al que se enfrenta la APS en los últimos años a nivel internacional es la demanda excesiva de consultas. Ésta puede ser definida como la que conlleva a masificación y, como consecuencia, una disminución del tiempo de atención deseable para cada paciente o un alargamiento excesivo de la duración de la consulta a demanda, en detrimento de otras actividades del centro. La demanda excesiva se manifiesta sobre todo a través de dos indicadores, el primero sería la presión asistencial que no es más que el número total de pacientes atendidos por el profesional sanitario al día, y el segundo sería la frecuentación que es el número de visitas por paciente al año.[4]

La sobrecarga asistencial tiene como consecuencias principales una disminución del tiempo de consulta dedicado al paciente, muy por debajo actualmente de los 10 minutos que reclaman la mayoría de los profesionales, incapacidad de desarrollar programas de prevención y promoción de la salud y una mayor prescripción, lo que a su vez genera un aumento de la frecuentación por parte del usuario, con lo que se entra en una espiral de deterioro progresivo de la calidad de la asistencia que se presta en los centros de salud.[5]

La atención médica en las consultas clínicas en la APS se ve limitada por los disímiles factores mencionados anteriormente que atentan contra el aumento de la calidad de la misma, afectándose la calidad de vida del paciente.

Atendiendo a estos factores se identificó el siguiente **problema**: ¿Cómo facilitar el proceso de gestión de la información en las consultas clínicas de APS?

Se definió como **objeto de estudio** el Proceso de gestión de la información en las consultas de APS y como **campo de acción**: el Proceso de gestión de la información en las consultas clínicas de APS.

Para dar solución al problema planteado se propuso como **objetivo general**: Desarrollar el Componente de Consulta Clínica del Subsistema Web del Sistema Integral para la Atención Primaria de Salud.

Para dar cumplimiento al objetivo general de la investigación fueron realizadas las siguientes tareas:

1. Identificación de las áreas y procesos que se realizan en las instituciones de APS relacionadas con las consultas clínicas.
2. Análisis del estado del arte de las tendencias actuales a nivel nacional e internacional relacionadas con los sistemas de apoyo a las consultas clínicas en la APS.
3. Realización del levantamiento de los procesos de cada una de las áreas relacionadas con las consultas clínicas en la APS.
4. Realización de la preparación correspondiente para el uso de la arquitectura definida y aprobada por el Centro de Informática Médica (CESIM) para el desarrollo de aplicaciones web.
5. Obtención de los artefactos correspondientes a las fases: “Modelado de Negocio”, “Requerimientos”, “Diseño” e “Implementación” definidas en la guía de desarrollo del programa de mejoras.

En la actualidad, existen otras aplicaciones desarrolladas tanto a nivel nacional como internacional que contribuyen al proceso de gestión de la información de las consultas clínicas en la APS, pero no cumplen con las funcionalidades requeridas en su totalidad y las que cumplen con algunas de las funcionalidades no pertenecen a la APS, por lo que presentan un grupo de limitaciones que hacen necesario desarrollar una solución diferente.

El presente documento está compuesto por cuatro capítulos que incluyen todo lo relacionado con el trabajo investigativo, así como el análisis, diseño e implementación del sistema.

Capítulo I Fundamentación Teórica: Fundamenta la necesidad, el valor de la investigación y se hace referencia a la informatización de la APS, los antecedentes y las estrategias. Se realiza un estudio de sistemas que resuelven problemas similares al planteado, así como cada una de las tecnologías, metodologías y herramientas utilizadas para el desarrollo de la aplicación.

Capítulo II Características del sistema: Describe la propuesta del sistema, identificando los procesos del negocio y las reglas que lo rigen, así como las mejoras que propone el mismo; además se describen cada uno de los procesos identificados por área, quienes son los encargados de llevar a cabo cada proceso y las actividades específicas realizadas por cada cual. Se detallan las funcionalidades que el sistema debe ser capaz de cumplir y se detallan los prototipos de interfaz de usuario.

Capítulo III Diseño del sistema: Describe los aspectos relacionados al diseño de la solución propuesta. Se estructura el modelo de diseño, se modelan los diagramas y se describen los patrones usados. También se realiza el diagrama de clases persistentes modelando toda la información que se procesa en el sistema y se describen los atributos, entidades y relaciones del mismo.

Capítulo IV Implementación: Trata los aspectos relacionados con la construcción de la solución propuesta. Aborda aspectos donde se define la implementación del sistema, la integración con otros módulos o sistemas y se describen cada uno de los componentes, módulos o sistemas implementados y el tratamiento de errores en la solución del sistema.

Capítulo 1. Fundamentación Teórica

Desde hace algunos años, a nivel mundial, los gobiernos y las direcciones de salud en particular se han percatado de la necesidad imperiosa de mejorar las actividades realizadas por los médicos ya que influyen enormemente en la calidad de vida de la población. En esta esfera se maneja gran cantidad de información la cual sirve como base imprescindible para tomar decisiones en cuanto a la salud del paciente. Por tanto no es de extrañar que esta área haya incorporado casi tanta tecnología a su infraestructura como el sector tecnológico de la economía.

Muchos han visto la solución en la informatización, dejando a un lado el estudio a profundidad del engranaje mediante el cual funciona el sector de la salud y pensando que con solo tener informatizado el sector ya resuelven todo el problema, pero nada más alejado de la realidad, un problema tan complejo necesita un estudio a profundidad y en muchos casos la introducción de tecnologías; si no se emplean correctamente, solo entorpecen el trabajo del médico.

En este capítulo se hace referencia a la informatización de la APS en el SNS y se analizan los conceptos fundamentales de la investigación. No solo se abordan los temas relacionados con la información generada en las consultas, sino que se profundiza en otras actividades que ocurren de forma paralela y su integración. También se realiza un estudio sobre aplicaciones existentes a nivel nacional e internacional, los niveles de empleo de las mismas por parte del personal de salud, las tendencias en las tecnologías usadas, las estrategias y tácticas para la informatización del SNS, así como el estado del arte de las tecnologías, técnica, metodologías y herramientas a utilizar para dar cumplimiento al objetivo general de la investigación.

1.1. Marco Conceptual

Algunos conceptos a tener en cuenta son los siguientes:[6]

Presión Asistencial: Número total de pacientes atendidos por el profesional sanitario al día.

Frecuentación: Acción de frecuentar. Número de visitas por paciente al año.

Especialidades Médica: Estudios cursados por un Graduado o Licenciado en Medicina en su período de postgrado, y que lo dotan de un conjunto de conocimientos médicos especializados relativos a un área

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

específica del cuerpo humano, a unas técnicas quirúrgicas o a un método diagnóstico determinado. Las especialidades médicas se corresponden con la figura tradicional del "médico" que asiste personalmente al paciente, realiza la anamnesis (se refiere a la información recolectada por un médico u otro profesional de la salud mediante preguntas específicas formuladas bien al propio paciente o bien a otras personas que conozcan a este último y puedan aportar datos útiles) y la historia clínica, diagnóstica (apoyándose en otros especialistas de diagnóstico) y tratan, pero sin utilizar técnicas quirúrgicas. La tendencia actual es a especializarse por sistemas, existiendo también algunas especialidades centradas en un determinado tipo de patología, o grupo de edad específico.

Sistema Nacional de Salud

La Salud es un derecho consagrado en numerosos tratados internacionales. La Declaración Universal de Derechos Humanos en el artículo 25, párrafo 1, establece: "toda persona tiene derecho a un nivel de vida adecuado que le asegure, así como a su familia, la Salud y en especial la alimentación, el vestido, la vivienda, la asistencia médica y los servicios sociales necesarios". La Salud es conceptualmente un proceso complejo que se construye social y biológicamente. El carácter histórico de este concepto exige contextualizarlo en tiempo y lugar y con perspectiva cultural. [7]

Conceptualmente, un Sistema Nacional de Salud es "el conjunto de unidades administrativas, de servicios, producción, docencia e investigación, responsabilizadas con la atención integral de la Salud a la población". [8]

Dentro de los principales objetivos del SNS están: perfeccionar la capacidad del personal médico de la salud, garantizar la introducción de tecnologías de punta en los programas de salud y elevar la calidad de los servicios que brindan a la población, mediante estrategias dirigidas al fortalecimiento de la Atención Primaria principalmente.

De acuerdo a la complejidad de las acciones curativas, preventivas y de rehabilitación, la atención de salud está organizada en tres niveles en estrecha relación. El eslabón principal lo constituye el nivel de *Atención Primaria de Salud* al que le siguen la *Atención Secundaria de Salud* y la *Atención Terciaria de Salud*.

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

Atención Médica Terciaria: El nivel terciario debe abarcar alrededor del 5 % de los problemas de salud, relacionados con secuelas o aumento de las complicaciones de determinadas dolencias. Se brindan servicios de muy alta complejidad, con la óptima utilización de los recursos y medios existentes y el desarrollo de la investigación.

Atención Médica Secundaria: Este nivel da cobertura a cerca del 15 % de los problemas de salud, su función fundamental es tratar al hombre ya enfermo, tanto desde el punto de vista individual como colectivo. Se llevan a cabo acciones de salud más complejas y especialidades médicas a mayor profundidad que en el nivel primario. Comprende la atención médica brindada en los diferentes tipos de instituciones hospitalarias.

Atención Primaria de Salud (APS): Da solución aproximadamente al 80 % de los problemas de salud de la población y a lo que corresponda con las acciones de promoción, investigación, prevención, prescripción y protección de la salud en la comunidad, así como acciones curativas básicas. Sus actividades están relacionadas fundamentalmente con los Centros de Salud Familiar o de la Comunidad, Unidades Móviles, Unidades Rurales, así como en Centros Especializados en Atención Primaria de Salud (CEAPS) que cumplen objetivos concretos en la sociedad.

1.2. Proceso de Consulta en APS

La actividad fundamental que se realiza en la APS es la **consulta**, que no es más que el encuentro y comunicación del paciente con el médico para el conocimiento y solución de los problemas de salud. La consulta puede ser **programada** o **a demanda**.

La **programada** es aquella mediante la cual el médico organiza y otorga los turnos, al finalizar la consulta primera le planifica una nueva consulta.

La **espontánea** o **a demanda** es aquella que se realiza sin previa citación al paciente y este acude a atenderse ese día.

Además se realizan acciones de **Terreno** que no es más que un tipo de consulta realizada fuera del local de consulta, el cual es planificado mediante las prioridades establecidas por las condiciones de los pacientes.

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

También se realizan actividades de **Interconsulta** que es el intercambio científico-técnico que se establece entre especialistas para llegar a un diagnóstico, definir un pronóstico y establecer una conducta con un objetivo asistencial y docente.

Durante la investigación realizada en los centros de APS se identificaron un conjunto de actividades realizadas en las consultas de las distintas especialidades. Todas ellas tienen un conjunto de acciones comunes que les permiten lograr su objetivo denominado método clínico.

El método clínico o "proceso del diagnóstico", son los pasos ordenados que todo médico aplica en la búsqueda del diagnóstico en sus enfermos individuales, y consisten en: formulación por el enfermo de sus quejas de salud; obtención de la información necesaria (síntomas, signos y otros datos) para después establecer las hipótesis diagnósticas presuntivas e ir a su comprobación final, por intermedio de una contratación que, en la mayoría de las circunstancias, aunque no en todas, se realiza a través de análisis de laboratorio, de cualquier tipo que sean. Así pues, los 5 pasos o etapas del método son: formulación, información, hipótesis, contratación y comprobación. [9]

El método clínico no es otra cosa que el método científico o experimental de todas las ciencias que fuera elaborado de forma coherente y definitiva por Claude Bernard en su libro "Introducción al estudio de la medicina experimental", escrito en la segunda mitad del siglo XIX, pero aplicado esta vez no a una investigación de laboratorio, sino a la atención individual de enfermos. [10]

Toda consulta comienza con un interrogatorio (la recopilación de datos y la información se basa únicamente en la sintomatología, en la indagación sobre la queja principal que presenta y con la estimación de los problemas que plantea de cara a determinar las áreas en las que se habrá de profundizar para establecer el tratamiento adecuado a seguir). En el mismo se trata de obtener información sobre los síntomas que aquejan al paciente. Seguidamente se le realiza un examen físico (opcional) y posteriormente se le indica la realización de exámenes complementarios en caso necesario para corroborar el diagnóstico y posterior tratamiento. Si el médico lo estima conveniente lo citará para una nueva consulta, ordenará ingreso, emitirá certificados, prescripciones médicas o dietas.

Una consulta genera gran cantidad de información la cual se guarda en documentos clínicos como pueden ser la hoja de cargo y la historia clínica individual del paciente, esta información es de gran importancia ya que permite una mayor eficiencia a la hora de tratar un paciente.

1.3. Estado del Arte

No cabe duda de que la tecnología está revolucionando el mundo y la sociedad actual. La forma en que nos comunicamos dentro de la comunidad y con las organizaciones se ha visto claramente influenciada por el efecto transformador de las Tecnologías de la Información y las Comunicaciones (TIC). Las TICs tienen la capacidad de catalizar los procesos de negocio, haciendo que estos sean cada vez más rápidos y más precisos de lo que alguna vez se pensó que llegarían a ser. Las estrategias tecnológicas digitales son aplicables a la mayoría de las instituciones públicas, redundando en una atención más rápida y efectiva al ciudadano y brindando nuevas soluciones a viejos problemas.

La implantación de estas tecnologías en el área de la salud permitirán facilitar los servicios brindados a los pacientes elevando así la calidad de la atención médica brindada. Dentro los procesos aplicables al sector de salud se encuentra la informatización, la cual permite facilitar los procesos del sector y catalizarlos de tal manera que solo se podía soñar con algo así. Pero la informatización trae consigo la necesidad de un análisis profundo ya que los altos costos de tiempo, esfuerzo y dinero que implica esta nueva campaña convierten el fracaso de la misma en inaceptable.

Con el objetivo de informatizar el SNS se establecen a nivel nacional y mundial diferentes estrategias que son muy diversas pues dependen del ambiente y la situación económica, ideológica y organizativa del país, pero todas tienen algo en común, el primer paso en su estrategia es el informatizar su base de forma tal que permita ir escalando los demás niveles de atención y el intercambio de información asistencial, o sea informatizar la APS.

Para lograr informatizar el sector se requieren de grandes inversiones y una gran apuesta estratégica en las Historias Clínicas Electrónicas (HCE) que permitan tener un registro médico único por paciente con bases de datos centralizadas, además de permitir la interoperabilidad con otros sistemas y el intercambio de información.

Una HCE es un repositorio de información, mantenido electrónicamente, con los datos de salud de toda la vida de un paciente, guardados de tal manera que puedan servir para múltiples usuarios del registro médico. Tradicionalmente, el registro médico fue un registro de datos médicos provisto por un paciente que estaba enfermo. Actualmente, los profesionales de la salud se focalizan en el seguimiento longitudinal del paciente, registrando información de pacientes sanos como de pacientes con alguna alteración

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

mórbida. Una HCE agrega herramientas de manejo de información que provee recordatorios clínicos y alertas enlazados a bases de conocimientos para el soporte a la toma de decisiones. A su vez, brindan opciones de análisis de datos agregados tanto para el gerenciamiento del cuidado como para la investigación.

En una historia clínica en papel, el lector debe manipular los datos mentalmente o destacar la información clínica relevante en forma manual (mediante subrayados con color o signos). En cambio, en un sistema de HCE, existen herramientas informáticas que pueden ayudar al lector a organizar, interpretar y reaccionar ante los datos.

En contraste al registro tradicional en papel, cuya funcionalidad está atada a la estática natural del mismo y donde los datos se guardan en un solo formato tanto en su ingreso como en su recuperación, la HCE es flexible y adaptable. Los datos pueden ser ingresados de manera tal que simplifiquen el proceso de ingreso y pueden ser mostrados en formatos diferentes para su mejor interpretación. Además la HCE puede integrar información multimedia como imágenes radiológicas, videos de ecocardiografías, informes de logaudiometrías, etc., que nunca hasta ahora fueron parte del registro tradicional. Los datos pueden ser usados para guiar el cuidado de un único paciente o en forma agregada para ayudar a los administradores a desarrollar políticas de salud poblacionales. [11]

Además de permitir la múltiple visualización de los datos permite una mejor comunicación con otros profesionales y con el paciente ya que muchos de estos sistemas tienen prestaciones similares al correo electrónico incluidas o la mensajería instantánea. De manera tal que diferentes profesionales pueden enviarse mensajes vinculados con la atención de ese paciente y temas relacionados.

Estas HCE usan una variante de la Historia Clínica Orientada Problemas la cual fue creada por Laurence Weed a finales de los 60's. La orientación a problemas es uno de los pocos e importantes cambios introducidos al registro médico en sus más de 2500 años de vida. Fue ideado para el registro en papel pero mejorando la organización de la información y así, poder realizar análisis estadísticos y epidemiológicos. Este cambio tuvo un gran impacto en el mantenimiento de las historias clínicas, pese a que en la actualidad no se utiliza el registro orientado a problemas de la forma que se ideó. [12]

En el verdadero modelo orientado a problemas el registro está segmentado por problemas médicos y, dentro de cada uno de ellos, toda la información es ingresada bajo cuatro encabezados: Subjetivo,

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

Objetivo, Evaluación y Plan; o también conocido como modelo SOEP. O sea, por cada problema hay un SOEP. [13]

Lo Subjetivo es aquello que el paciente refiere: síntomas, su evolución, etc. Lo Objetivo es lo que el personal de la salud observa cuando examina al paciente o cuando obtiene los resultados de los laboratorios. La Evaluación es lo que el profesional cree que está sucediendo: diagnóstico, pronóstico, etc. El Plan es lo que el profesional de la salud planea hacer como prescribir medicamentos o solicitar más estudios. [14]

En la HCE se aplica el modelo orientado a problemas con una ligera variante, el modelo SOEP se aplica a todo el encuentro médico y no a cada problema de salud en particular. O sea que en Subjetivo, se describen todos los síntomas del paciente e información relacionada y todos sus problemas. En Objetivo se agrega todo lo encontrado en el examen físico y estudios complementarios. Luego, en Evaluación, se listan todos los problemas encontrados y en Plan se describen las acciones que se va a tomar sobre cada uno de los mismos; fusionándose, a veces, una evaluación y un plan combinados bajo un mismo encabezado. Hay que tener en cuenta que es muy común, tanto en registro en papel como electrónico, tener listas de problemas que resuman la patología del paciente en un listado actualizado. [15]

Para la elaboración de estos registros médicos electrónicos las direcciones de SNS contratan compañías de desarrollo de software introduciéndose uno de los grandes problemas de la esfera puesto que los productos que entregan son software propietario y encadenan a la APS en particular a tener además de pagar altos precios por la compra del software, el tener que pagar el mantenimiento del mismo y comprar las próximas versiones a los precios que las compañías estimen conveniente. Una de las alternativas a este problema es el uso de sistemas desarrollados con software libre pues los precios del producto son más bajos y no hay necesidad de atarse a un proveedor.

Pero aún con estos problemas su uso va en aumento, existe una gran cantidad de información disponible sobre los intentos de cuantificar el uso de las HCE. Esta información está cambiando todo el tiempo, aunque es claro que la adopción está aumentando gradualmente.

Las tasas mayores de adopción se encuentran en países desarrollados como Reino Unido, Australia y otros países de Europa Occidental seguidos de cerca por los EEUU y aunque se encuentran un poco

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

lejos, los países Latinoamericanos también están dando pasos en la implementación y puesta en marcha de una HCE.

Alrededor del mundo existen diferentes tipos de sistemas que brindan mucha ayuda al diagnóstico del médico, están desde los que realizan sugerencias hasta los que solo se limitan a registrar la información.

Los registros médicos electrónicos pueden ser de 4 tipos: [16]

- Visualización de la información, son aquellos que simplemente muestran información, no necesariamente se hacen recomendaciones.
- Sistemas de recordatorios, son aquellos que les recuerdan a los profesionales realizar ciertas acciones.
- Los de Alarmas, son aún más interactivos. Alertan a los médicos en situaciones críticas.
- Las guías de práctica clínica que son la última forma de soporte para la toma de decisiones proveen protocolos que permiten estandarizar la atención. En otras palabras, la atención se realizaría con un estilo estandarizado por los distintos profesionales, de acuerdo con el actual conocimiento científico sobre una condición.

Pero además de los sistemas informáticos, el mundo está mirando hacia el uso de otras tecnologías que permitan la transportación y más fácil trasiego de información así como la realización de los procesos asistenciales en ámbitos muy difíciles.

Entre las nuevas tecnologías usadas se encuentran las tarjetas inteligentes que permiten el almacenamiento de los datos contenidos en la historia clínica electrónica de forma que esta se convierte en la llave del sistema informatizado y evita los problemas de seguridad y confidencialidad de la información.

Otras de las tecnologías con tendencias positivas en su uso, es la teleconsulta o consulta a distancia. Esta tecnología permite que un paciente pueda recibir atención médica sin necesidad de que el médico se encuentre presente o realizar una interconsulta con otro médico para llegar a un mejor diagnóstico. El principal problema a que se enfrenta esta tecnología es el ancho de banda, el cual no pueden costear las

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

naciones menos favorecidas pues ya les resulta una gran preocupación mantener el presupuesto actual de su APS como para introducir otro gasto aún mayor y la creencia cada vez más arraigada de que esta tecnología afecta la relación médico paciente.

Todas estas tecnologías además de sus beneficios han traído para los médicos muchos retos como son el consumo de tiempo en la utilización de las tecnologías en detrimento del empleado en el paciente, y para los desarrolladores de los mismos las barreras de interoperabilidad. Con tal de contrarrestar estos males las soluciones informáticas actuales están orientadas hacia la confección de interfaces de usuarios más usables e intuitivas que le permitan al médico dedicar menos tiempo a la introducción de datos a la HCE así como la utilización de estándares que permitan la comunicación entre los diferentes sistemas que se utilizan en la APS.

1.3.1. Análisis de Soluciones Existentes

En el ámbito internacional en distintas regiones del mundo se utilizan sistemas para registrar la información generada en las consultas independientemente del nivel de atención. Se pueden citar sistemas como el Itálica, sistema desarrollado para la informatización de la capa clínica del Hospital Italiano de Buenos Aires. Este sistema está centrado en el paciente (una única HC por paciente) y está orientada a problemas. En su estructura de datos, la lista de problemas, es la columna vertebral y todo lo que se registre deberá estar relacionado a un problema. Contiene un módulo de solicitud de exámenes complementarios que permite la búsqueda textual de la práctica. Dicho módulo también está integrado con el componente de los servicios efectores que realizan las prácticas por medio de mensajería electrónica basada en el estándar HL7. Por medio del mismo sistema de mensajería electrónica basado en HL7 los resultados e informes de exámenes complementarios son enviados de vuelta.

Permite además la visualización de los fármacos consumidos actualmente por el paciente y el detalle de cada uno de ellos. También provee de acceso a bases de conocimiento sobre los principios activos (monografías de los mismos) e información relevante para pacientes.

Otro sistema de HCE es el Logician desarrollado en Estados Unidos por General Electric y al igual que Itálica fue diseñado para la atención hospitalaria permitiendo al equipo de salud documentar los encuentros ambulatorios de los pacientes así como la visualización integrada de los datos del paciente,

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

soporte para la toma de decisiones clínicas, módulo de Solicitud de Estudios y Ordenes Clínicas, acceso a bases de conocimiento y soporte integrado para la comunicación.

Una funcionalidad siempre presente en este sistema es la de agrupar en solapas algunos detalles específicos para una organización modular de la información. Logician, como la mayoría de los sistemas renombrados, permiten la aplicación de reglas de soporte para la toma de decisiones y contiene un módulo para el ingreso estructurado de órdenes médicas. Se establece la posología y es posible ver los fármacos consumidos por el paciente actualmente y la lista de alergias. Si el profesional lo desea, puede también renovar las indicaciones directamente según la farmacia preferida por el paciente, electrónicamente o por fax. También cuenta con un sistema de comunicación integrada que permite envío de correo electrónico entre los distintos profesionales con respecto a un paciente.

De acuerdo con los estudios realizados, en Cuba existen intentos de trabajo como el SYSAPS (Aplicación web para la gestión de datos bioestadísticos en APS), que está orientado a la atención primaria de salud (APS), siendo una aplicación Web, que emplea las enormes que brinda la red de redes (Internet) y que permite gestionar datos sobre los pacientes sean: edad, sexo, antecedentes patológicos personales, y otros datos de la historia clínica individual así como datos de la historia de salud familiar respecto a las viviendas, como las condiciones estructurales, cultura sanitaria, etc.; así mismo se permite consultar y reportar información referente a las reacciones adversas a los medicamentos. [17]

APUS es otra aplicación, desarrollada por el CEDISAP (Centro de Desarrollo Informático para la Salud Pública) hace algunos años, contiene un módulo: el Sistema Informático para el Registro y Control de Consultas Médicas, con el fin de registrar la información recogida a través del Modelo del médico generado en las consultas. Al comienzo del uso del software APUS mejoró un poco la eficiencia de trabajo del policlínico, no así de los consultorios, y confrontó las dificultades siguientes: [18]

- Dificil codificación clínica de muchos problemas de salud por desajuste de las clasificaciones estándares de enfermedades y de problemas de salud en la APS.
- Poca aceptación médica del llenado más cuidadoso y lento de datos en sus modelos, para que las estadísticas los entraran en las computadoras, sin tener retroalimentación.

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

No contó con toda la comprensión gerencial, médica y técnica estadística, ni con las condiciones informáticas y prioridades requeridas. [19]

Así, estos sistemas fueron desarrollados para darle solución a un problema en específico y no como partes de un todo que al unirse permita una informatización de la APS, no cuentan con un diseño que les permita el intercambio de información con otros sistemas, ni la comunicación, por lo que solo entorpecen más la labor del médico; y aquellos que si tienen en cuenta la interoperabilidad y la integración han sido desarrollados con herramientas propietarias y tienen grandes requerimientos de hardware y software.

Realizar modificaciones en estas soluciones informáticas para convertirla en la solución óptima requeriría más esfuerzo, presupuesto y riesgo que realizar una nueva, la documentación sobre las mismas es poca o inexistente, se deberían incorporar desde cero las concepciones de interoperabilidad, usabilidad y configuración y no sería factible ya que las soluciones son mayormente propietarias y su código fuente no está libre para ser modificado.

1.4. Situación Problemática

Los centros de APS en todo el mundo presentan una serie de problemas, entre ellos se encuentran los relacionados con esta gestión de la información generada durante las consultas y otras actividades de apoyo a las mismas. Estos problemas afectan en gran medida la capacidad de respuesta de estos centros a la población, trayendo como consecuencia que se vea afectada la calidad asistencial ofrecida y con ello la calidad de vida de los pacientes.

Actualmente, el procesamiento de dicha información se realiza de forma manual resultando engorroso para el personal médico que lo realiza. La información es recogida en modelos confeccionados en papel, material que con el paso del tiempo se deteriora y puede implicar pérdida de información importante para futuros estudios que se quieran obtener.

La gestión manual de toda la información que se genera en las consultas entorpece en gran medida el trabajo del personal que labora en los centros de APS, afectando la calidad de los servicios brindados al paciente e influyendo de manera negativa en la toma de decisiones. Además, la introducción y actualización de datos de forma manual trae como consecuencia que muchas veces se omitan detalles y que exista falta de homogeneidad en la información debido a la no utilización de estándares, lo que

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

provoca problemas de comunicación entre las distintas especialidades médicas que dificultan la atención al paciente; y la obtención de estadísticas. La dificultad del proceso se evidencia más a la hora de realizar búsquedas, demorando la obtención del historial clínico de los pacientes.

Otro de los problemas que se presenta es la imposibilidad de acceder a la información fuera del área a la cual pertenece el paciente o también que muchas veces estos no cuentan con un historial clínico, y los que si lo tienen, en la mayoría de los casos, no están completos. Hay que tener en cuenta que no todos los médicos anotan en la historia clínica y los que si lo hacen, usan su propio lenguaje. Todo esto hace que la información contenida en la historia clínica del paciente varíe de médico a médico, eso sin contar que actualmente existe pero no se usa (o ya no se produce) un formato estándar de historia clínica, en la mayoría de los casos es simplemente una libreta en blanco en la que el médico anota lo que cree importante. En ocasiones es necesario crear una nueva historia clínica al paciente cuando llega a la consulta, proceso que retrasa la atención del mismo, y no se conocen qué estudios previos se le han realizado, obstaculizando así el propósito de las instituciones sanitarias que no es otro que brindar una mejor atención.

El seguimiento a los problemas de salud de los pacientes es una acción importante dentro de la APS que se dificulta por el hecho anteriormente explicado, por lo que el diagnóstico y tratamiento de una determinada patología se basa solo en lo que refiere el paciente en algunos casos o un pequeño historial que va incluido en la remisión cuando este es remitido hacia otra especialidad para hacerle un estudio más profundo, no teniendo en cuenta sus antecedentes patológicos o estudios realizados con anterioridad provocando que los diagnósticos no sean lo más acertados posibles y el paciente tenga que ser sometido a diversos estudios para llegar a un conocimiento definitivo del problema que lo aqueja.

Otro de los sectores que se ve afectado son las investigaciones médicas, tanto por la falta del historial clínico como por el formato del mismo, el cual hace más lenta la búsqueda de determinada información dentro de este. El médico debe dedicar más tiempo y esfuerzo en la obtención de estadísticas en detrimento de la atención dedicada al paciente. Esta situación conlleva a que los Departamentos de Estadísticas en todos los niveles, principalmente a nivel nacional no posean información completa y a corto plazo de cada una de sus estructuras inferiores y a la hora de realizar estudios no se disponga de información actualizada de manera rápida en cualquier momento del año. Además, la falta de calidad en la

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

información disponible impone limitaciones a la hora de intentar llevar a cabo una correcta planificación sanitaria.

A esto se agrega que las historias clínicas no siempre contienen la información mínima que permita analizar características de un paciente, tratamientos recibidos, pruebas diagnósticas realizadas y el seguimiento de su curso vital. Los especialistas no tienen conocimiento de los resultados de los exámenes complementarios indicados en el momento, tienen que esperar a que algún personal destinado a ello o el propio paciente los recoja. Esto hace aún más lento el proceso de obtención de un diagnóstico definitivo y tratamiento de la patología. Al mismo tiempo, al realizar diagnósticos de salud sin regirse por estándares, impiden el establecimiento de un lenguaje común.

Todo lo analizado impide realizar diagnósticos certeros, teniendo en cuenta antecedentes patológicos y otros, así como facilitar el tratamiento y seguimiento del paciente en consecuencia con el entorno en el que convive, disminuyendo la calidad de vida de los pacientes. Al no garantizar la oportunidad y consistencia de la información se dificulta la realización de los procesos relacionados con la salud, trayendo como consecuencia disminución de la calidad en la atención médica.

Por tanto, establecer el control de la información relacionada con los pacientes, así como el estado de cumplimiento de las diferentes actividades de forma manual, como se ha venido realizando hasta el momento, representa el empleo de gran parte del tiempo de la jornada laboral en estas tareas y esfuerzo por parte del médico, que sin dudas pudiera ser destinado a labores asistenciales y de prevención.

Por otra parte, uno de los principales problemas al que se enfrenta la APS en los últimos años a nivel internacional es la demanda excesiva. Ésta puede ser definida como la que conlleva masificación y, como consecuencia, una disminución del tiempo de atención deseable para cada paciente o un alargamiento excesivo de la duración de la consulta a demanda, en detrimento de otras actividades del centro. [20]

Esta demanda excesiva se manifiesta sobre todo a través de dos indicadores, el primero sería la presión asistencial que no es más que el número total de pacientes atendidos por el profesional sanitario al día, y el segundo sería la frecuentación que es el número de visitas por paciente al año. [21]

Ambos indicadores están estrechamente relacionados. Por ejemplo, en España la frecuentación ronda las 6,5 consultas por persona al año. La Presión Asistencial (PA) es muy variable de unas comunidades a

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

otras, pero en general anda claramente por encima de los 25 a 30 pacientes por profesional al día que consideran aceptables la mayoría de sociedades científicas para un Centro de Salud (CS) urbano. Por encima de estas cifras se podría hablar de exceso de PA. [22]

La sobrecarga asistencial tiene como consecuencias principales una disminución del tiempo de consulta dedicado al paciente, muy por debajo actualmente de los 10 minutos que reclaman la mayoría de los profesionales, incapacidad de desarrollar programas de prevención y promoción de la salud y una mayor prescripción, lo que a su vez genera un aumento de la frecuentación por parte del usuario, con lo que se entra en una espiral de deterioro progresivo de la calidad de la asistencia que se presta en los centros de salud. [23]

Tres son los factores claves en el exceso de demanda, la ratio profesional/habitante, debido al insuficiente número de médicos; los pacientes hiperfrecuentadores, sobre los que las técnicas de citación en visitas programadas parecen haber fracasado, al haber demostrado diversos trabajos que estas tácticas generan una frecuentación mayor y aumentan el número de frecuentadores inducidos; y la burocracia en la consulta que hoy en día puede rondar el 20-40% de las consultas diarias, a finales de la década anterior se pensó que la progresiva informatización de las consultas atenuaría en gran medida este problema, pero la presión asistencial por demandas administrativas sigue siendo elevada. [24]

Expertos en Atención Primaria de Salud enfatizan sobre las consecuencias de disponer de poco tiempo para consulta, o la importancia que tiene para la satisfacción del paciente el tiempo que se le dedica, trayendo consigo una serie de problemas, como son:

- Mayor facilidad de cometer errores médicos, tanto por la prisa como por el cansancio.
- Escaso tiempo para registrar adecuadamente los datos recogidos y con ello aumentar las posibilidades de cometer errores, y, por supuesto, mayor dificultad para realizar trabajos de investigación, o programas de evaluación.
- Dificultad en el seguimiento de las patologías.
- Insatisfacción generalizada y cansancio profesional.

Pero también, para mitigar todas estas consecuencias, a nivel mundial se toman medidas, estrategias y planificaciones, así como se introducen tecnologías para lograr una mejor calidad de la atención médica del sector primario.

1.5. Tecnologías

Las tecnologías a usar para dar cumplimiento al objetivo general de la investigación serán las definidas por el CESIM para el desarrollo de sistemas. A continuación se exponen definiciones, características y ventajas de su utilización.

1.5.1. Java EE 5.0

Java Platform, Enterprise Edition es una plataforma de programación para desarrollar y ejecutar software de aplicaciones en Lenguaje de programación Java con arquitectura de N niveles distribuida, basándose ampliamente en componentes de software modulares ejecutándose sobre un servidor de aplicaciones. [25]

Es un estándar para el desarrollo de aplicaciones empresariales portables, robustas, escalables y seguras usando tecnología Java. Uno de los beneficios de Java EE como plataforma es que es posible empezar con poco o ningún coste.

1.5.2. Estilos Arquitectónicos

Ateniendo a la recomendación IEEE-1471, arquitectura de software se entenderá como “la organización fundamental de un sistema encarnada en sus componentes, las relaciones de los componentes con cada uno de los otros y con el entorno, y los principios que orientan su diseño y evolución”.

1.5.2.1. Arquitectura en Capas

La programación por capas es un estilo arquitectónico. La ventaja principal de este estilo, es que el desarrollo se puede llevar a cabo en varios niveles y en caso de algún cambio solo se modifica al nivel requerido sin tener que revisar entre código mezclado. Entre las más usadas se encuentra 3 capas la cual se compone de:

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

Capa de presentación: Es la que ve el usuario, presenta el sistema al usuario, le comunica la información al sistema y captura la información del usuario.

Capa de negocio: Es donde residen los programas que se ejecutan, se reciben las peticiones del usuario y se envían las respuestas tras el proceso. Se denomina capa de negocio (e incluso de lógica del negocio) pues es aquí donde se establecen todas las reglas que deben cumplirse.

Capa de datos: Es donde residen los datos. Está formada por uno o más gestores de bases de datos que realizan todo el almacenamiento de datos. [26]

1.5.2.2. MVC

El Modelo Vista Controlador (MVC) es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos. Se ve frecuentemente en aplicaciones web. Los elementos de este patrón son:

Modelo: Esta es la representación específica de la información con la cual el sistema opera. La lógica de datos asegura la integridad de estos y permite derivar nuevos datos.

Vista: Este presenta el modelo en un formato adecuado para interactuar, usualmente la interfaz de usuario.

Controlador: Este responde a eventos, usualmente acciones del usuario e invoca cambios en el modelo y probablemente en la vista.

1.5.3. Lenguajes

1.5.3.1. Java

Java es un lenguaje de programación orientado a objetos desarrollado por Sun Microsystems. El lenguaje toma mucha de su sintaxis de C y C++, pero tiene un modelo de objetos más simple y elimina herramientas de bajo nivel, que suelen inducir a muchos errores, como la manipulación directa de punteros o memoria. Fue liberado bajo la licencia GNU GPL. Es fácil de usar y permite la ejecución de un mismo programa en múltiples sistemas operativos.

1.5.3.2. XHTML

XHTML, acrónimo en inglés de eXtensible Hypertext Markup Language (lenguaje extensible de marcado de hipertexto), es el lenguaje de marcado pensado para sustituir a HTML como estándar para las páginas web. XHTML es la versión XML de HTML, por lo que tiene, básicamente, las mismas funcionalidades, pero cumple las especificaciones, más estrictas, de XML. [27]

1.5.3.3. XML

XML, (lenguaje de marcas extensible), es un metalenguaje extensible de etiquetas. No es un lenguaje en particular, sino una manera de definir lenguajes para diferentes necesidades. No ha sido creado sólo para su aplicación en Internet, sino que se propone como un estándar para el intercambio de información estructurada entre diferentes plataformas. Se puede usar en bases de datos, editores de texto, hojas de cálculo y casi cualquier cosa imaginable.

1.5.3.4. EL

Es el lenguaje de expresión que provee JSF, es utilizado en las páginas de las aplicaciones web con el fin de acceder a las entidades java. [28]

1.5.3.5. HQL

HQL es un lenguaje de consulta de gran alcance usado por Hibernate, es similar en apariencia a SQL. En comparación con SQL, sin embargo, HQL es completamente orientado a objetos y comprende nociones como herencia, polimorfismo y asociación. [29]

1.5.3.6. UML 2.0

UML es un lenguaje para visualizar, especificar, construir y documentar los artefactos de un sistema que involucra una gran cantidad de software. Permite modelar sistemas con tecnología orientada a objetos. Es importante resaltar que UML es un "lenguaje" para especificar y no para describir métodos o procesos. [19]

1.5.4. Metodología de Desarrollo

Una buena metodología de software pretende reducir costos y retrasos de proyectos, así como mejorar la calidad del software. La metodología de desarrollo cobra gran importancia en proyectos empresariales, pues al no utilizarla adecuadamente se puede desembocar en la frustración del equipo de desarrollo y en la insatisfacción de los clientes por lo que su uso es necesario para controlar el ciclo de vida de un proyecto.

El ciclo de vida básico a seguir para el desarrollo de la solución consta de 9 flujos de trabajo: Estudio Preliminar, Modelación del Negocio, Requisitos, Análisis y Diseño, Implementación, Pruebas Internas, Pruebas de Liberación, Despliegue y Soporte; los cuales se encuentran descritos en el IPP-3510:2009 Libro de Proceso para la Administración de Requisitos, realizado por la universidad como parte del programa de mejoras que se está llevando a cabo actualmente para alcanzar el nivel 2 de CMMI (Capability Maturity Model Integration).

Este documento establece el ciclo de vida a seguir asociado a los proyectos involucrados en el proceso de mejora e incluye además la definición de roles, sus responsabilidades y las habilidades en la ejecución de las actividades de los distintos procesos y los productos típicos de trabajos que se obtienen como resultado de la ejecución de dichas actividades. La información contemplada en cada producto típico de trabajo es registrada según lo definido en la metodología RUP (Rational Unified Process o Proceso Unificado de Desarrollo) y se usa la notación BPMN (Business Process Management) para el modelado de los procesos de negocio.

1.5.5. Servidor de Base de Datos

Los Sistemas de gestión de base de datos son un tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan. Se compone de un lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta.

1.5.5.1. PostgreSQL 8.3

PostgreSQL es un servidor de base de datos relacional orientada a objetos de software libre, liberado bajo la licencia BSD. Está considerado como la base de datos de código abierto más avanzada del mundo. Proporciona un gran número de características que normalmente sólo se encontraban en las bases de

datos comerciales tales como DB2 u Oracle. PostgreSQL aproxima los datos a un modelo objeto-relacional, y es capaz de manejar complejas rutinas y reglas.

1.5.6. Servidor de Aplicación

En informática, se denomina servidor de aplicaciones a un servidor en una red de computadores que ejecuta ciertas aplicaciones. Usualmente se trata de un dispositivo de software que proporciona servicios de aplicación a las computadoras cliente. Un servidor de aplicaciones generalmente gestiona la mayor parte o la totalidad de las funciones de lógica de negocio y de acceso a los datos de la aplicación.

1.5.6.1. JBoss 4.2

JBoss es un servidor de aplicaciones Java EE de código abierto implementado en Java solamente. Al estar basado en Java, JBoss puede ser utilizado en cualquier sistema operativo que lo soporte. Su robusta pero flexible arquitectura de código abierto junto con su bajo coste ha convertido rápidamente a JBoss AS en la solución de middleware más popular entre los desarrolladores, los proveedores independientes de software y las empresas corporativas.

1.5.7. Frameworks

Un framework es una estructura de soporte conceptual y tecnológica definida mediante la cual otro proyecto de software puede ser organizado y desarrollado. Provee una estructura y una metodología de trabajo la cual extiende o utiliza las aplicaciones del dominio. Son diseñados con la intención de facilitar el desarrollo de software, permitiendo a los diseñadores y programadores pasar más tiempo identificando requerimientos de software que tratando con los tediosos detalles de bajo nivel de proveer un sistema funcional.

1.5.7.1. EJB 3.0

EJB 3.0 es un framework basado en Plain Old Java Objects (POJO) para servicios de negocios y persistencia de base de datos. El objetivo de los EJBs es dotar al programador de un modelo que le permita abstraerse de los problemas generales de una aplicación empresarial (conurrencia, transacciones, persistencia, seguridad,...) 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. [31]

1.5.7.2. JSF 2.0

JSF (Java Server Faces) es un framework de desarrollo basado en el patrón MVC (Modelo Vista Controlador). Desarrollado a través del Java Community Process, la tecnología JavaServer Faces establece el estándar para la construcción de interfaces de usuario del lado del servidor. Las API JavaServer Faces se están diseñando de manera que puedan ser aprovechados por herramientas que harán que el desarrollo de aplicaciones web sea más fácil.

1.5.7.3. JPA

Java Persistence API, más conocida por su sigla JPA, es la API de persistencia desarrollada para la plataforma Java EE e incluida en el estándar EJB3. Esta API busca unificar la manera en que funcionan las utilidades que proveen un mapeo objeto-relacional. El objetivo que persigue el diseño de esta API es no perder las ventajas de la orientación a objetos al interactuar con una base de datos, como sí pasaba con EJB2, y permitir usar objetos regulares (conocidos como POJOs).

1.5.7.4. Hibernate 3.2

Hibernate es una herramienta de Mapeo objeto-relacional para la plataforma Java que facilita el mapeo de atributos entre una base de datos relacional tradicional y el modelo de objetos de una aplicación, mediante archivos declarativos (XML) que permiten establecer estas relaciones. Es software libre, distribuido bajo los términos de la licencia GNU LGPL.

1.5.7.5. SEAM

SEAM es una potente plataforma de desarrollo de código abierto para construir aplicaciones ricas de Internet en Java. Seam integra tecnologías como JavaScript asíncrono y XML (AJAX), JavaServer Faces (JSF), Java Persistence Api (JPa), Enterprise Java Beans (EJB 3.0) y Business Process Management (BPM). [32]

Seam elimina la capa artificial que existe entre EJB 3.0 y JSF y provee un consistente sistema de anotaciones para integrar estos dos frameworks. Comparada con aplicaciones desarrolladas en otros frameworks, las aplicaciones Seam son conceptualmente simples y requieren significativamente menos código (en Java y en XML) para obtener las mismas funcionalidades.

1.5.7.6. Facelets

JavaServer Facelets es un framework para plantillas centrado en la tecnología JSF (JavaServer Faces), por lo cual se integran de manera muy fácil. Facelets llena el vacío entre JSP y JSF, siendo una tecnología centrada en crear árboles de componentes y estar relacionado con el complejo ciclo de vida JSF. [33]

1.5.7.7. RichFaces 3.1.3

RichFaces es una librería de componentes visuales para JSF. Además, RichFaces posee un framework avanzado para la integración de funcionalidades Ajax en dichos componentes visuales, mediante el soporte de la librería Ajax4JSF. Se integra perfectamente en el ciclo de vida de JSF y contiene un set de componentes visuales, los más comunes para el desarrollo de una aplicación web rica (Rich Internet Application), con un número bastante amplio que cubren casi todas nuestras necesidades. [34]

1.5.8. Herramientas

1.5.8.1. Eclipse Galileo

Eclipse es un entorno de desarrollo integrado de código abierto multiplataforma. La definición que da el proyecto Eclipse acerca de su software es: "una especie de herramienta universal - un IDE abierto y extensible para todo y nada en particular". Eclipse es, en el fondo, únicamente un almacén (workbench) sobre el que se pueden montar herramientas de desarrollo para cualquier lenguaje, mediante la implementación de los plugins adecuados. [35]

1.5.8.2. Visual Paradigm 6.4

Visual Paradigm para UML es una herramienta UML profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. El software de modelado UML ayuda a una más rápida construcción de aplicaciones de calidad, mejores y a un menor coste. VP-UML aporta a los desarrolladores de software una plataforma de desarrollo puntera para construir aplicaciones de calidad, mejores y más baratas con rapidez. Aporta una excelente interoperabilidad con otras herramientas CASE y muchos de los entornos IDE líderes del mercado. [36]

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

En este capítulo se abordaron conceptos generales referentes al Sistema Nacional de Salud y la Atención Primaria de Salud, así como todo el proceso de informatización que se lleva actualmente en el mundo, los cuales resultan significativos para una mejor comprensión de la investigación. Se realizó un análisis de las tecnologías involucradas en el desarrollo del sistema propuesto, fundamentándose también la elección del lenguaje de programación, el sistema gestor de bases de datos, la metodología y los frameworks a utilizar. Además, se describen las herramientas a utilizar; teniendo en cuenta las políticas definidas por el CESIM para el desarrollo de sistemas. También se analizaron algunos de los sistemas existentes en Cuba y el mundo relacionados con la solución propuesta y que de una forma u otra tributan al por qué de la investigación.

Capítulo 2. Características del Sistema

El producto final de cualquier fábrica es el resultado del proceso de elaboración del que fue objeto, o sea, los pasos que se siguieron para lograr terminarlo con éxito, lo mismo ocurre con el software y su proceso de desarrollo. Para cualquier producto es de suma importancia la fase de inicio, el principio de todas las acciones, pues es bien sabido que lo que mal empieza, mal a acaba. Así en el software y su ciclo de desarrollo es de gran importancia el conocimiento del negocio, tener bien claro lo que el sistema debe hacer y las cualidades que lo harán atractivo. El pasar por alto alguna de estas premisas, ha sido siempre la causa de fracaso de muchos proyectos o la baja calidad de los productos nacidos en los mismos.

Este capítulo está dedicado a describir las características generales del sistema. Además se describen los principales procesos del negocio identificados y se especifican los requisitos funcionales y no funcionales de la solución.

2.1. Modelo de Negocio

El modelado del negocio es una técnica para comprender los procesos del negocio de la organización. Los propósitos que se persiguen al realizarse este modelo son: entender la estructura y la dinámica de la organización, entender los problemas actuales e identificar mejoras potenciales, asegurarse de que los clientes, usuarios finales y desarrolladores tengan una idea común de la organización y derivar los requerimientos del sistema a partir del modelo de negocio que se obtenga. A continuación se describe el desempeño de los roles en los diferentes procesos de negocio.

Rol	Función
Médico	Es el encargado de realizar la consulta, atender, diagnosticar, tratar y darle seguimiento al paciente.
Enfermera	Realiza actividades dentro de la consulta siguiendo las indicaciones del médico como pueden ser pesar y tallar a los pacientes, entre otras.
Recepcionista	Es la encargada de controlar los turnos así

CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

	como de validar y registrar los certificados médicos.
--	---

2.1.2. Procesos de Negocio

El principal proceso que realizan los diferentes especialistas en cada una de las instituciones de APS, diariamente, es: la atención al paciente mediante consultas, interconsultas y terrenos. Para comprender mejor como se llevan a cabo este y otros procesos que complementan el mismo a continuación se explican las actividades realizadas en diferentes áreas dentro de las instituciones de APS para el proceso principal identificado: Realizar Consulta, los demás se pueden encontrar en el expediente de proyecto.

Los procesos identificados son los siguientes:

- Realizar Consulta.
- Validar Certificados.
- Validar Dietas.

2.1.3. Descripción de los Procesos

Acrónimos:

CM Certificado Médico

CTMC Certificados de Tratamiento con Medicamento Controlado

HCI Historia Clínica Individual

HC Hoja de Cargo

CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

2.1.3.1. P1 Realizar Consulta

Nombre:	P1 Realizar Consulta
Objetivos:	Consultar y tratar al paciente.
Evento(s) que lo generan:	Presencia del paciente en consulta con síntomas o evidencias de enfermedad del corazón, urología, angiología, reuma, endocrinas, otorrinos, etc.
Precondiciones:	Debe haber sacado un turno médico.
Poscondiciones:	Diagnosticar al paciente.
Reglas de Negocio:	1,2,3,4,5,7,10,11,12 ver documento de referencia [37]
Responsables:	Médico
Clientes internos:	Maikel David Ruenes Correga y Johander León Garcés
Clientes externos:	No procede
Entradas:	Hoja de cargo, Remisión, HCI
Salidas:	Remisión de Ingreso, CM, CTMC, Receta Médica, Método, Indicación de Examen, Modelo Remisión, Orden de Cita.
Actividades:	Act.1 Identificar al Paciente. Act.2 Realizar Interrogatorio. Act.3 Realizar Examen Físico. Act.4 Revisar Resultados de Exámenes Complementarios Act.5 Indicar Exámenes Complementarios. Act.6 Emitir Diagnóstico. Act.6.1 Indicar Ingreso. Act.8 Indicar Tratamiento. Act.9 Dar seguimiento.

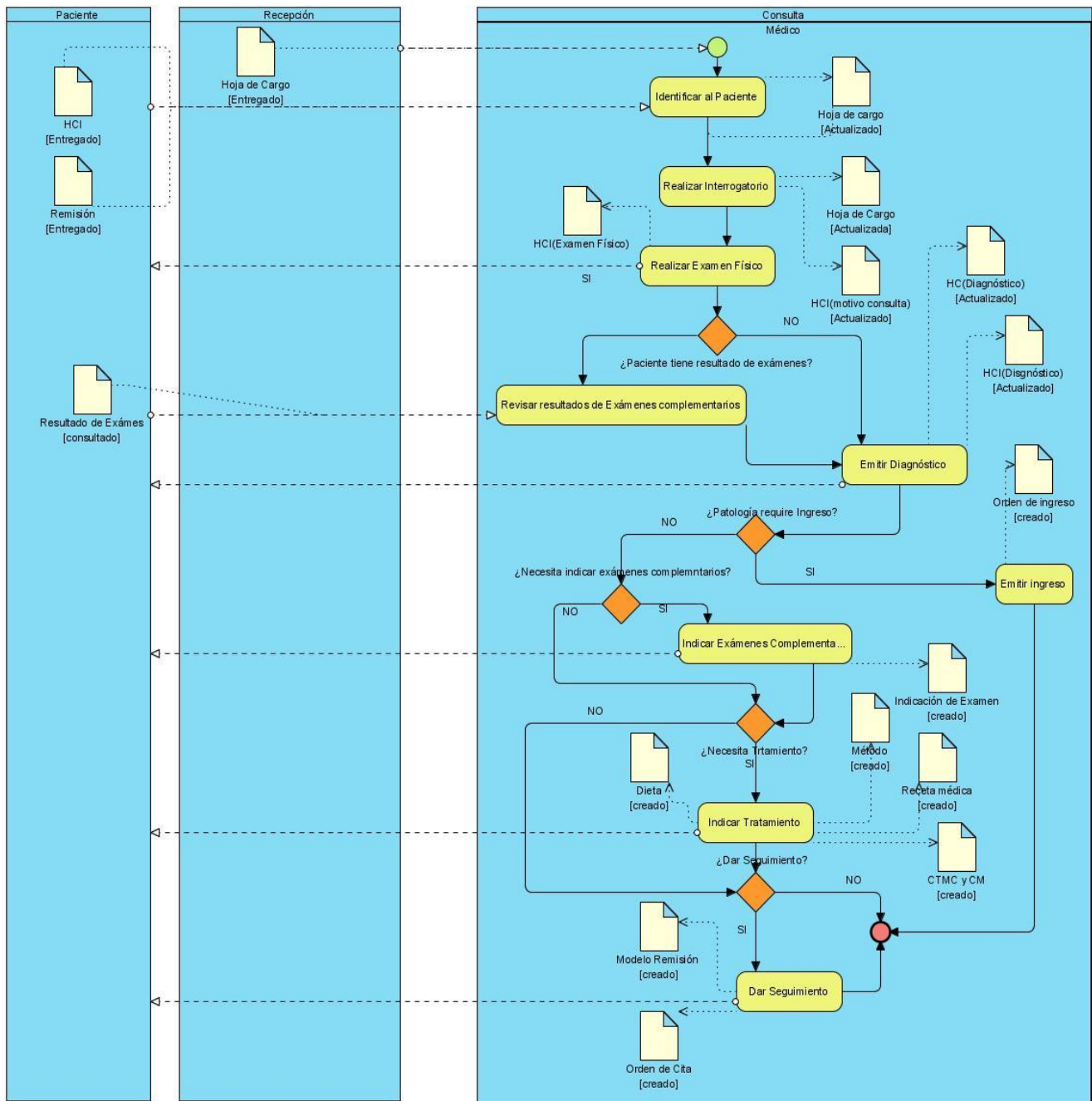
2.1.4. Diagramas de Procesos

Para cada proceso se define un diagrama que está compuesto por calles, que representan los roles del negocio. En estas se muestran las principales acciones que se realizan en un orden lógico y que responden al proceso de negocio descrito. Por otra parte se incluyen las entidades que son artefactos que

CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

se generan como parte de las actividades, los cuales representan la entrada o salida de información en la realización de cada una de ellas.

2.1.4.1. P1 Realizar Consulta:



2.1.4.1.1. Descripción del Flujo Básico

1. Identificar al paciente: Se solicita al paciente la remisión y la HCI si viene por primera vez a la consulta, en caso contrario solo la HCI; no se procede a tomar sus datos personales puesto que ya el mismo viene con un turno sacado con antelación (ver P4 Planificar Consulta en el expediente de proyecto), por lo que los datos como: nombre, apellidos, edad, sexo y dirección ya han sido previamente registrados en la hoja de cargo, en esta también se registra la fecha y nombre del médico.

Responsable: Médico

Entradas: HC, Remisión, HCI

Salidas: HC

2. Realizar Interrogatorio: Se le pregunta al paciente el motivo de su visita y cuáles son los síntomas que presenta, se registra en la HCI y en la HC.

Responsable: Médico

Entradas: No procede

Salidas: HCI (Motivo de Consulta), HC.

3. Realizar Examen Físico: Se le realiza a cada paciente cuando llega a la consulta, este depende del interrogatorio realizado, es registrado en la HCI. Entre los tipos de exámenes físicos a realizar, se encuentran: El médico tomará las siguientes mediciones físicas: Peso, Talla, Presión Arterial Sistólica, Presión Arterial Diastólica, y Frecuencia Cardíaca. Se calcula el Índice de masa corporal y se clasifica al paciente según este. Observará Mucosas y hará Auscultación Pulmonar, Auscultación Cardíaca, Palpación Abdominal, Palpación Renal. Se anotarán, además otras observaciones clínicas.

Responsable: Médico

Entradas: No procede

Salidas: HCI (Examen Físico)

4. Revisar Resultados de Exámenes Complementarios: En caso que el paciente acuda con resultados de análisis que pueden haber sido indicados por otro especialista o por el mismo, el médico procede a revisar dichos resultados y los registra en la HCI.

CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

Responsable: Médico

Entradas: Resultado de exámenes complementarios.

Salidas: HCI (Resultado de exámenes complementarios)

5. Indicar Exámenes Complementarios: El médico si lo cree necesario indica un grupo de exámenes complementarios. Esta indicación contiene fecha, datos del paciente (nombre y apellidos), nombre del médico y nombre del análisis.

Responsable: Médico

Entradas: No procede

Salidas: Indicación de Examen.

6. Emitir Diagnóstico: Si el paciente acude por primera vez a la consulta y acorde al interrogatorio, el examen físico, la remisión y/o resultados de análisis complementarios, el médico emite un diagnóstico que puede ser presuntivo o confirmado. En otro caso, acorde al interrogatorio, el examen físico y/o resultados de análisis complementarios, el médico emite un diagnóstico que puede ser presuntivo o confirmado, el cual es registrado en la hoja de cargo y en la HCI.

Responsable: Médico

Entradas: Remisión, HCI (Motivo de Consulta), HCI (Examen Físico), Resultado de Análisis Complementarios.

Salidas: HCI (Diagnóstico), HC (Diagnóstico).

7. Indicar Tratamiento: En dependencia del diagnóstico emitido por el médico se le puede poner, cambiar o derogar un tratamiento al paciente, el cual es registrado en su HCI. El médico le entrega al paciente una o varias Recetas con la medicación requerida y un Método indicando la forma de realizar el tratamiento. También el médico si lo estima conveniente le puede indicar un Certificado Médico al paciente (Certificado Médico o Certificado de Tratamiento con Medicamentos Controlados), o en algunos casos indicarle Dieta. Tanto la dieta como el certificado deben ser validados, ver P3 Validar Dieta y P2 Validar Certificado Médico en el expediente de proyecto.

Responsable: Médico

Entradas: No procede

CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

Salidas: Receta Médica, Método, HCl, Certificado Médico, Certificado de Tratamiento con Medicamentos Controlados, Dieta.

8. Dar Seguimiento: El médico según todo lo visto durante la consulta decide si el paciente necesita ser valorado además de él por otro especialista (Interconsulta), o si necesita una nueva cita o no (seguimiento). En correspondencia con la patología del paciente el médico establece el período de tiempo necesario para la nueva consulta en una orden que le entrega para sacar el turno, ver P4 Planificar Consulta. En el caso de remitirlo a otra especialidad llena un modelo de remisión donde plasma los datos personales del paciente (nombre, apellidos, sexo, dirección) y el motivo de la remisión (Diagnóstico).

Responsable: Médico

Entradas: No procede

Salidas: Modelo de Remisión, Orden de Cita.

2.1.4.1.2. Descripción de los Flujos Paralelos

6.1. Indicar Ingreso: El médico le da una orden de ingreso al paciente para proceder al ingreso del mismo si es necesario, esta orden contiene: datos del paciente (nombre, apellidos, sexo y edad), anamnesis (historia de la enfermedad), examen físico, diagnóstico, exámenes complementarios, nombre del médico y firma.

Responsable: Médico

Entradas: No procede

Salidas: Orden de Ingreso.

2.2. Propuesta de Sistema

El punto fundamental a tener en cuenta al realizar cualquier acción es el establecer y planificar que se quiere hacer y cómo llevarlo a cabo. Una vez trazados los objetivos y las estrategias para lograr los mismos se pone en marcha un conjunto de acciones planificadas con anterioridad, pero cualquier cambio en nuestro objetivo hará cambiar toda o gran parte de la estrategia elegida por lo que se retornaría el mismo punto de inicio y en muchos casos es necesario desechar todo lo realizado hasta el momento. No es de extrañar que esta sea la tarea más importante en cualquier empresa que se emprenda, lo mismo ocurre con el software.

CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

Para el software es más costoso un cambio de los objetivos, por tanto es de vital importancia conocer y acordar con los clientes los requisitos funcionales y no funcionales del sistema, que se tenga un entendimiento entre los integrantes del proyecto y los clientes de lo que el sistema debe hacer y hasta donde se va a llegar, así como las estrategias y tácticas que se emplearán en cada fase de desarrollo.

2.2.1. Especificación de Requerimientos de Software

Un requerimiento según la IEEE Standard Glossary of Software Engineering Terminology se define como: “condición o capacidad que necesita un usuario para resolver un problema o lograr un objetivo, condición o capacidad que tiene que ser alcanzada o poseída por un sistema o componente de un sistema para satisfacer un contrato, estándar, u otro documento impuesto formalmente y una representación documentada de una condición o capacidad”. [38]

Todas las ideas que los clientes, usuarios y miembros del equipo de proyecto tengan acerca de lo que debe hacer el sistema, deben ser analizadas como candidatas a requisitos. Los requisitos se pueden clasificar en: funcionales y no funcionales.

2.2.1.1. Requisitos Funcionales

Los requerimientos funcionales son capacidades o condiciones que el sistema debe cumplir, los mismos no alteran la funcionalidad del producto, esto quiere decir que los requerimientos funcionales se mantienen invariables sin importar con que propiedades o cualidades se relacionen. A partir del estudio e investigaciones realizadas en el negocio del problema, se obtuvieron los siguientes requerimientos funcionales que ha de cumplir el sistema, con vista a la satisfacción del cliente:

R1	Crear Hoja de Cargo	Si no existe una hoja de cargo para esa fecha se crea una.
R2	Listar Especialidades	Se muestra un listado con las especialidades que puede realizar consulta ese médico en esa entidad
R3	Listar Citas por Especialidad	Permite mostrar las citas disponibles para la especialidad seleccionada en el R2
R4	Crear Consulta	Se crea la consulta para el paciente que fue

CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

		seleccionado en el listado de citas del R3 y se agrega a la hoja de cargo del R1. Al seleccionar el paciente a consultar se activarán las pestañas: Subjetivo, Objetivo, Evaluación y Plan (Correspondientes al modelo SOEP) y Vista Previa que permite visualizar los datos recogidos a lo largo de la consulta y terminar la misma.
R5	Listar Consultas de Hoja de Cargo	Se muestran los detalles de los pacientes consultados en las consultas de esa hoja de cargo creada en el R1
R6	Listar Antecedentes Patológicos	Se listan los antecedentes patológicos del paciente
R7	Listar Problemas Crónico	Se listan los problemas crónicos que ha tenido el paciente
R8	Registrar Estilos de Vida	Se registran los estilos de vida y la frecuencia con que lo ejecuta el paciente
R9	Listar Estilos de Vida	Se listan los estilos de vida
R10	Listar Frecuencia de Estilos de Vida	Se listan las frecuencias asociadas al estilo de vida seleccionado en el R9
R11	Buscar Problema de Salud	Se busca un problema de salud en los codificadores CIAP y en CPAPS
R12	Registrar Motivo de Consulta	Permite registrar el problema de salud buscado en el R11 como un motivo de consulta
R13	Mostrar Resultados de exámenes Complementarios	Se muestra la lista de exámenes realizados y sus resultados
R12	Buscar Examen Físico	Permite realizar una búsqueda de los exámenes físicos
R13	Registrar Resultado de Examen Físico	Permite registrar los resultados del examen

CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

		físico seleccionado en el R12
R14	Registrar Problema de Salud	Permite registrar el problema de salud seleccionado en el R11
R15	Clasificar Problema de Salud	Permite clasificar el problema de salud registrado en el R14 y establecer su duración, asociación y si está confirmado o es presuntivo
R16	Listar Diagnósticos	Permite listar los problemas de salud registrados en el R14
R17	Registrar Tipo de Tratamiento	Permite registrar el tipo de tratamiento que se le dará a un problema de salud
R18	Buscar Medicamento	Permite realizar una búsqueda por el estándar ATC de medicamentos
R19	Registrar Medicamento	Permite registra el medicamento buscado en el R18
R20	Registrar Indicación	Permite registrar la indicación del medicamento seleccionado en el R19
R20.1	Listar Frecuencia	Lista las frecuencias con que se le indicara el medicamento
R20.2	Listar Periodo	Lista el periodo con que se le indicara el medicamento
R20.3	Calcular Fecha de Suspensión	Permite calcular la fecha de suspensión a partir del periodo seleccionado
R21	Listar Diagnósticos y Tratamientos	Se lista los diagnósticos con los tratamientos asociados al mismo
R22	Registrar Tratamiento	Se registra el Tratamiento y se le adiciona la indicación
R23	Buscar Exámenes Complementarios	Permite realizar una búsqueda de exámenes complementarios por departamentos y servicios

CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

R24	Registrar Examen Complementario	Permite realizar una solicitud del examen complementario seleccionado en el R23
R25	Gestionar Seguimiento	
R25.1	Agregar a Seguimiento	Permite agregar la consulta en curso al seguimiento seleccionado, agregarle una nota evolutiva y planificar una nueva consulta
R25.2	Terminar Seguimiento	Permite terminar el seguimiento seleccionado con la consulta en curso y agregarle una nota evolutiva.
R25.3	Iniciar Seguimiento	Permite iniciar un seguimiento con la consulta en curso, agregarle una nota evolutiva y planificar una nueva consulta
R25.4	Listar Seguimiento	Permite listar los seguimientos en curso del paciente
RF 26	Gestionar causas de alta	Permite buscar, modificar, eliminar y adicionar causas de alta (causas por las que un paciente puede ser dado de alta de un determinado seguimiento, estas causas pueden ser por muerte, porque el paciente ya fue curado o las que sean establecidas por el médico)
RF 26.1	Buscar causa de alta	Permite buscar una causa de alta
RF 26.2	Agregar causa de alta	Permite agregar una causa de alta
RF 26.3	Modificar causa de alta	Permite modificar una causa de alta
RF 26.4	Eliminar causa de alta	Permite eliminar una causa de alta
RF 27	Gestionar duración de problema de salud	Permite buscar, modificar, eliminar y adicionar una clasificación de problema de salud según duración (un problema puede ser crónico o agudo según duración)

CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

RF 27.1	Buscar duración de problema de salud	Permite buscar una duración de problema de salud
RF 27.2	Agregar duración de problema de salud	Permite agregar una duración de problema de salud
RF 27.3	Modificar duración de problema de salud	Permite modificar una duración de problema de salud
RF 27.4	Eliminar duración de problema de salud	Permite eliminar una duración de problema de salud
RF 28	Gestionar registro problema de salud	Permite buscar, modificar, eliminar y adicionar una clasificación de problema de salud según registro (un problema puede ser asociado según registro)
RF 28.1	Buscar registro del problema de salud	Permite buscar un registro de problema de salud
RF 28.2	Agregar registro del problema de salud	Permite agregar un registro de problema de salud
RF 28.3	Modificar registro del problema de salud	Permite modificar un registro de problema de salud
RF 28.4	Eliminar registro del problema de salud	Permite eliminar un registro de problema de salud
RF 29	Gestionar aparición de problema de salud	Permite buscar, modificar, eliminar y adicionar una clasificación de problema de salud según aparición (un problema puede ser nuevo o conocido según aparición)
RF 29.1	Buscar aparición de problema de salud	Permite buscar una aparición de problema de salud
RF 29.2	Agregar aparición de problema de salud	Permite agregar una aparición de problema de salud
RF 29.3	Modificar aparición de problema de salud	Permite modificar una aparición de problema de salud

CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

RF 29.4	Eliminar aparición de problema de salud	Permite eliminar una aparición de problema de salud
RF 30	Gestionar examen físico	Permite buscar, modificar, eliminar y adicionar un examen físico (ej.: peso, talla)
RF 30.1	Buscar examen físico.	Permite buscar un examen físico
RF 30.2	Agregar examen físico	Permite agregar un examen físico
RF 30.3	Modificar examen físico	Permite modificar un examen físico
RF 30.4	Eliminar examen físico	Permite eliminar un examen físico
RF 31	Gestionar estado examen físico	Permite buscar, modificar, eliminar y adicionar un estado de examen físico (ej.: normal, anormal)
RF 31.1	Buscar estado examen físico	Permite buscar un estado de examen físico
RF 31.2	Agregar estado examen físico	Permite agregar un estado de examen físico
RF 31.3	Modificar estado examen físico	Permite modificar un estado de examen físico
RF 31.4	Eliminar estado examen físico	Permite eliminar un estado de examen físico

2.2.1.2. Requerimientos no Funcionales

Los requerimientos no 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. En muchos casos son fundamentales en el éxito del producto. Normalmente están vinculados a requerimientos funcionales, es decir una vez se conozca lo que el sistema debe hacer se puede determinar cómo ha de comportarse, qué cualidades debe tener o cuán rápido o grande debe ser. [39]

Estos requerimientos en muchos casos son los que marcan la diferencia a la hora de realizar una elección de un software si las propuestas cumplen con los mismos requerimientos funcionales. Así, el que un producto sea más seguro, más usable, más rápido o más conveniente puede ser lo que haga que más usuarios deseen comprarlo y por ende se convierta en un éxito.

CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

2.2.1.2.1. Requerimientos de Usabilidad

El sistema estará diseñado de manera que los usuarios adquieran las habilidades necesarias para explotarlo en un tiempo reducido.

2.2.1.2.2. Requerimientos de Fiabilidad

En los servidores de los policlínicos y en el Centro de Datos Nacional se garantizará una arquitectura de máxima disponibilidad, tanto de servidores de aplicación como de bases de datos. Se garantizarán además políticas de respaldo a toda la información, evitando pérdidas en caso de desastres ajenos al sistema.

Las informaciones médicas relacionadas con los pacientes y que vayan a ser intercambiadas con otros policlínicos por la red pública, viajarán cifradas para evitar accesos o modificaciones no autorizadas.

Se mantendrá la seguridad y el control a nivel de usuario, garantizando el acceso de los mismos sólo a los niveles establecidos de acuerdo a la función que realizan. Las contraseñas podrán cambiarse sólo por el propio usuario o por el administrador del sistema.

Se registrarán todas las acciones que se realizan, llevando el control de las actividades de cada usuario en todo momento.

El sistema implementará un mecanismo de auditoría para el registro de todos los accesos efectuados por los usuarios, proporcionando un registro de actividades (log) de cada usuario en el sistema.

Ninguna información que se haya ingresado en el sistema será eliminada físicamente de la BD, independientemente de que para el sistema, este elemento ya no exista.

El sistema permitirá la recuperación de la información de la base de datos a partir de los respaldos o salvadas realizadas.

2.2.1.2.3. Requerimientos de Eficiencia

El sistema minimizará el volumen de datos en las peticiones y además optimizará el uso de recursos críticos como la memoria. Para ello se potenciará como regla guardar en la memoria caché datos y recursos de alta demanda.

CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

El sistema respetará buenas prácticas de programación para incrementar el rendimiento en operaciones costosas para la máquina virtual como la creación de objetos. Se deberá usar siempre que sea posible el patrón de diseño Singleton, destruir referencias que ya no estén siendo usadas, optimizar el trabajo con cadenas, entre otras buenas prácticas que ayudan a mejorar el rendimiento.

2.2.1.2.4. Restricciones de Diseño

La capa de presentación contendrá todas las vistas y la lógica de la presentación. El flujo web se manejará de forma declarativa y basándose en definiciones de procesos del negocio. La capa del negocio mantendrá el estado de las conversaciones y procesos del negocio que concurrentemente pueden estar siendo ejecutados por cada usuario. La capa de acceso a datos contendrá las entidades y los objetos de acceso a datos correspondientes a las mismas. El acceso a datos está basado en el estándar JPA y particularmente en la implementación del motor de persistencia Hibernate.

2.2.1.2.5. Requisitos para la Documentación de Usuarios en Línea y Ayuda del Sistema.

Se posibilitará el uso de ayudas dinámicas y tutoriales en línea sobre el funcionamiento del sistema.

2.2.1.2.6. Requerimientos de Interfaz

➤ Interfaces de Usuario

Las ventanas del sistema contendrán los datos claros y bien estructurados, además de permitir la interpretación correcta de la información. La interfaz contará con teclas de función y menús desplegables que faciliten y aceleren su utilización. La entrada de datos incorrecta será detectada claramente e informada al usuario. Todos los textos y mensajes en pantalla aparecerán en idioma español.

2.2.1.2.7. Requerimientos de rendimiento

El sistema minimizará el volumen de datos en las peticiones y además optimizará el uso de recursos críticos como la memoria.

El sistema respetará buenas prácticas de programación para incrementar el rendimiento en operaciones costosas para la máquina virtual como la creación de objetos.

CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

2.2.1.2.8. Requerimientos de Soporte

Se permitirá la creación de usuarios, otorgamiento de privilegios y roles, asignación de perfiles y activación de permisos por direcciones IP. Se permitirá realizar copias de seguridad de la base de datos hacia otro dispositivo de almacenamiento externo, además de recuperar la base de datos a partir de los respaldos realizados. Se permitirá el chequeo de las operaciones y acceso de los usuarios al sistema. Se permitirá establecer parámetros de configuración del sistema y actualización de nomencladores.

2.2.1.2.9. Requerimientos de Hardware

➤ Servidores

La solución estará conformada, fundamentalmente, por servidores de alta capacidad de procesamiento y redundancia, que permitan garantizar movilidad y residencia de la información y las aplicaciones bajo esquemas seguros y confiables. Se recomienda que los servidores de base de datos posean los siguientes requerimientos: 1 DL380 G5, Procesador Intel® Xeon® 5140 Dual - Core 4GB de memoria y 2x72GB de disco y sistema operativo Linux y los servidores de aplicaciones: 2 DL380 G5, Procesador Intel® Xeon® 5140 Dual - Core 4GB de memoria y 2x72GB de disco y sistema operativo Linux.

2.2.1.2.10. Requerimientos de Software

El sistema debe correr en sistemas operativos Windows, Unix y Linux, utilizando la plataforma JAVA (Java Virtual Machine, JBoss AS y PostgreSQL). El sistema deberá disponer de un navegador web, estos pueden ser IE 7, Opera 9, Google Chrome 1 y Firefox 2 o versiones superiores de estos.

Con la culminación de este capítulo se sentaron las bases para el exitoso desarrollo de las Disciplinas de Diseño e Implementación del sistema. Se identificaron y describieron cada uno de los procesos del negocio y las actividades que son desarrolladas para llevar a feliz término cada uno, haciendo énfasis en el proceso Realizar Consulta. Además, se obtuvo una perspectiva más amplia del sistema a desarrollar, identificándose los requisitos funcionales y no funcionales que deberán ser cumplidos.

Capítulo 3. Diseño del Sistema

Para llevar a cabo la construcción de un producto de software exitoso se torna de vital importancia el ser muy cuidadosos y prestar mucha atención a cada uno de los pasos dentro del ciclo de vida de desarrollo del proyecto, fundamentalmente los relacionados con el flujo de trabajo de Diseño cuyo objetivo principal es modelar el sistema, encontrando su forma para que soporte todos sus requisitos, tanto funcionales como no funcionales.

Este capítulo está dedicado a describir el diseño de la solución propuesta con el objetivo de transformar los requisitos funcionales en un diseño de clases, teniéndose en cuenta los aspectos relacionados con los requisitos no funcionales. Se construye el Modelo de Datos, los diagramas de clases del diseño y de interacción, describiéndose todas las clases involucradas. Además, se definen los principios de diseño y los distintos patrones a utilizar en el diseño del sistema.

3.1. Diseño

El diseño orientado a objetos es una manera de pensar para el desarrollo de los sistemas de información que se encuentran en el mundo real y un buen mecanismo para afrontar la complejidad de un software.

Los propósitos fundamentales del flujo de trabajo de diseño son:

- Transformar los requerimientos en un diseño de cómo el sistema debe ser.
- Desarrollar una arquitectura robusta del sistema.
- Adaptar el diseño para que se corresponda con el entorno de implementación, diseñando sus funcionalidades.

En concreto, el diseño tiene el propósito de formar la base para la implementación, desarrollar la arquitectura, adaptar el diseño para que sea consistente con el entorno de implementación, dándole soporte a todos los requisitos, incluyendo los no funcionales y las restricciones que se le imponen y descomponer los trabajos a desarrollar en partes más manejables que puedan ser llevadas a cabo por diferentes equipos de desarrollo, teniendo en cuenta aspectos como: lenguaje de programación a utilizar, sistema operativo donde se podrá ejecutar la aplicación y tecnologías a emplear.

3.2. Modelo de Diseño

El modelo de diseño es un modelo de objetos que describe la realización física de los casos de uso centrándose en como los requisitos funcionales y no funcionales, junto con otras restricciones relacionadas con los lenguajes de programación, tienen impacto en el sistema a considerar. Sirve de abstracción para actividades de implementación, mediante la captura de los requisitos o subsistemas individuales, interfaces y clases. En la elaboración del diseño se utilizan un grupo de patrones o modelos para tener una comprensión más simple de la arquitectura del sistema. [40]

3.3. Definición de Elementos del Diseño

En el Modelo de diseño se establece la realización de las funcionalidades en clases teniendo en cuenta elementos como el lenguaje de programación, la arquitectura definida, etc. Está compuesto por los diagramas de clases del diseño así como los diagramas de interacción que pueden ser de secuencia o colaboración. Se definió la realización de un diagrama de clases por cada funcionalidad y uno por cada gestor para el caso de los codificadores. El diagrama de clases responderá a la arquitectura definida que tiene como característica principal el uso del patrón MVC. A continuación se dará una breve explicación del patrón MVC y como se ve reflejado en la solución.

MVC

Entre los patrones más usados en la construcción de aplicaciones web y que ha demostrado ser fundamental se encuentra el Modelo-Vista-Control (MVC). Este es un patrón de arquitectura que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos: la vista, el modelo y el controlador.

Modelo: Constituye la representación específica de la información con la cual el sistema opera.

Vista: Presenta el modelo en un formato adecuado para interactuar, usualmente la interfaz de usuario.

Controlador: Responde a eventos, usualmente acciones del usuario e invoca cambios en el modelo y probablemente en la vista.

Todo el sistema está basado en el patrón Model 2, que no es más que el patrón MVC definido para las aplicaciones Java EE y que tiene como característica que usa un único servlet como controlador para toda

la aplicación, atendiendo al patrón Front Controller (controlador frontal o fachada). El poder centralizar en un solo punto servicios como la gestión de conexiones a base de datos, comprobaciones de seguridad o gestión de errores favorecen que la aplicación sea mucho más robusta y aísla de todos estos aspectos al resto de los componentes.

En la aplicación, las vistas son todas las clases interfaz con las que interactúa directamente el usuario y están construidas mayormente con componentes JSF los cuales hacen peticiones al controlador mediante eventos, son controladas por el servlet controlador de JSF que es el encargado del ciclo de vida y de transportar los datos provenientes de las controladoras, actualizando la interfaz de usuario.

Forman parte de las vistas además, los frameworks RichFaces con la librería Ajax4JSF y Facelets (vista estática). El controlador que engloba toda la lógica del negocio, está representado por las clases controladoras y usa como framework SEAM, el cual es el encargado de unificar las tecnologías JSF, EJB3 y JPA facilitando la comunicación entre la presentación y el negocio. En el modelo se encuentran los EJB y para el acceso a los datos se usa el framework Hibernate que facilita una total independencia del gestor de Base de datos y provee de un lenguaje de consultas totalmente orientado a objetos; también se usa JPA; principalmente la interfaz EntityManager para persistir los datos.

Como resultado de la utilización del patrón MVC y debido a que la solución es una aplicación web para la realización de los diferentes diagramas se definió el uso de las facilidades de extensión brindadas por el UML específicamente: estereotipos web.

Los estereotipos web como se les conocen hoy en día son resultado de la extensión propuesta por Jim Conallen en 1998 al UML para el diseño aplicaciones Web, siendo los elementos más significativos los estereotipos: Server Page, Client Page, y Form. El diseño de la solución será modelado con dichos estereotipos por lo que a continuación se brinda una explicación breve de cada uno de ellos:

<<Server Page>>: Representa la clase que tiene código que se ejecuta en el servidor, la cual se encarga de construir o generar el resultado HTML.

<<Client Page>>: Es una página Web con formato XHTML. Son interpretadas por el navegador. Cada página cliente es construida por una sola página de servidor.

<<HTML Form>>: Es una colección de elementos de entrada que están contenidos en la página cliente.

Sus atributos son los elementos de entrada del formulario (input, textareas, radiobuttons, checkboxes, hiddenfields, entre otros).

Es importante destacar que una página cliente es construida por una sola página servidora; esta a su vez, puede completar su funcionamiento incluyendo algunas de las librerías que componen el framework JSF.

3.4. Uso de Patrones

En la elaboración del diseño se encontrarán frecuentemente con problemas que se repiten o son análogos a otros existentes y cuya solución puede ser dada de forma muy similar a alguna que han visto con anterioridad. Para dar respuesta a estos problemas surgen los llamados patrones. Estos dan soluciones simples y elegantes a problemas específicos. Son muy usados en el mundo de la informática ya que permiten tener una comprensión más simple de la arquitectura del sistema y facilitan el proceso de desarrollo.

Los patrones de diseño, no son más que la descripción de un problema y la solución del mismo, de forma que se pueda utilizar en diferentes contextos dando respuesta a interrogantes comunes. Estos permiten dar flexibilidad y extensibilidad al diseño en cuestión y son una técnica para flexibilizar el código haciéndolo satisfacer ciertos criterios. Además, contribuyen a reutilizar diseño, identificando aspectos claves de la estructura que pueden ser aplicados en una gran cantidad de situaciones. A continuación se describen los patrones utilizados durante el diseño de la solución propuesta.

3.4.1. Patrones GRASP

Un importante papel en el diseño realizado, lo tuvieron los patrones generales de software para asignación de responsabilidades (GRASP, en inglés). Estos son un conjunto de patrones que se basan en los principios generales de asignar responsabilidades para el diseño eficiente del software orientado a objetos. Se utilizan con el objetivo de determinar las responsabilidades que tienen las diferentes clases que se definieron en el diseño.

Dentro de este grupo se identifican principalmente cinco patrones muy utilizados: experto, creador, alta cohesión, bajo acoplamiento y controlador que son utilizados frecuentemente cuando se requiere implementar una clase que conozca o realice las funcionalidades de acuerdo a la información que contiene en el sistema y su relación de dependencia con otras clases u objetos. [29]

3.4.1.1. Alta Cohesión

Una clase presenta alta cohesión cuando el objeto tiene delimitadas sus responsabilidades, es decir, no se le asignan responsabilidades que no estén dentro de sus funciones por lo que no se encuentra sobrecargada. La cohesión de una clase significa cuan relacionadas y enfocadas están las acciones de la misma, si se implementa el sistema con baja cohesión entonces se complican sus clases, su organización y la posibilidad de mantenerlo en el futuro.

La solución propuesta se diseñó asignando responsabilidades de modo que la cohesión sea alta, ejemplo de esto lo representan cada una de las clases del negocio, las cuales tienen bien delimitadas sus responsabilidades, no encontrándose sobrecargadas en ningún momento, permitiendo de esta forma una mayor eficiencia y generando un bajo acoplamiento.

3.4.1.2. Bajo Acoplamiento

El patrón bajo acoplamiento está estrechamente vinculado al patrón alta cohesión y hace referencia a que un objeto tenga una dependencia escasa con el resto del sistema (que las clases estén lo menos ligadas entre sí que se pueda). Esto posibilita que se puedan realizar modificaciones en diferentes partes del programa sin necesidad de que los objetos relacionados se vean afectados en gran medida lo que trae como consecuencia que sea más fácil dar mantenimiento al software y que sea flexible a los cambios, potenciándose la reutilización. Una clase con bajo o débil acoplamiento no depende de "muchas otras".

3.4.1.3. Controlador

Este patrón facilita la centralización de actividades (validaciones, seguridad, etc.) consiste en asignar la responsabilidad de controlar el flujo de eventos del sistema a clases específicas. El controlador es el que recibe los datos del usuario y el que los envía a las distintas clases según el método llamado. Un error muy común es asignarle demasiada responsabilidad y alto nivel de acoplamiento con el resto de los componentes del sistema.

3.4.1.4. Experto y Creador

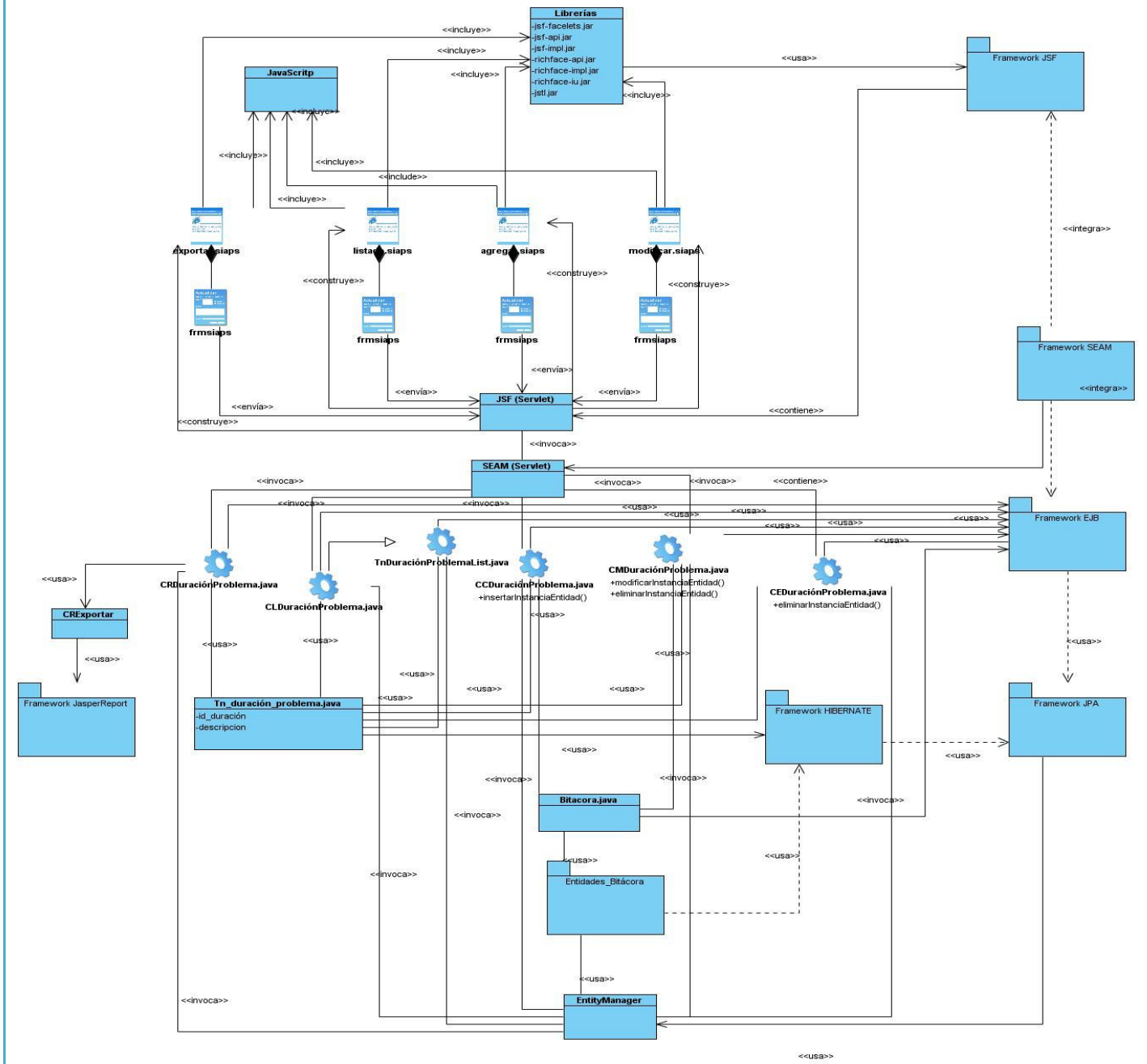
Además de los patrones explicados anteriormente también se pusieron en práctica el patrón Experto y el Creador, obteniéndose como resultado que a cada clase del diseño se les fueron asignadas tareas acorde a la información que poseía, y crearon instancias acorde a la responsabilidad que cada clase tenía. Como

resultado, se logró conservar el encapsulamiento, pues los objetos logran valerse de su propia información para realizar las tareas que se les solicita.

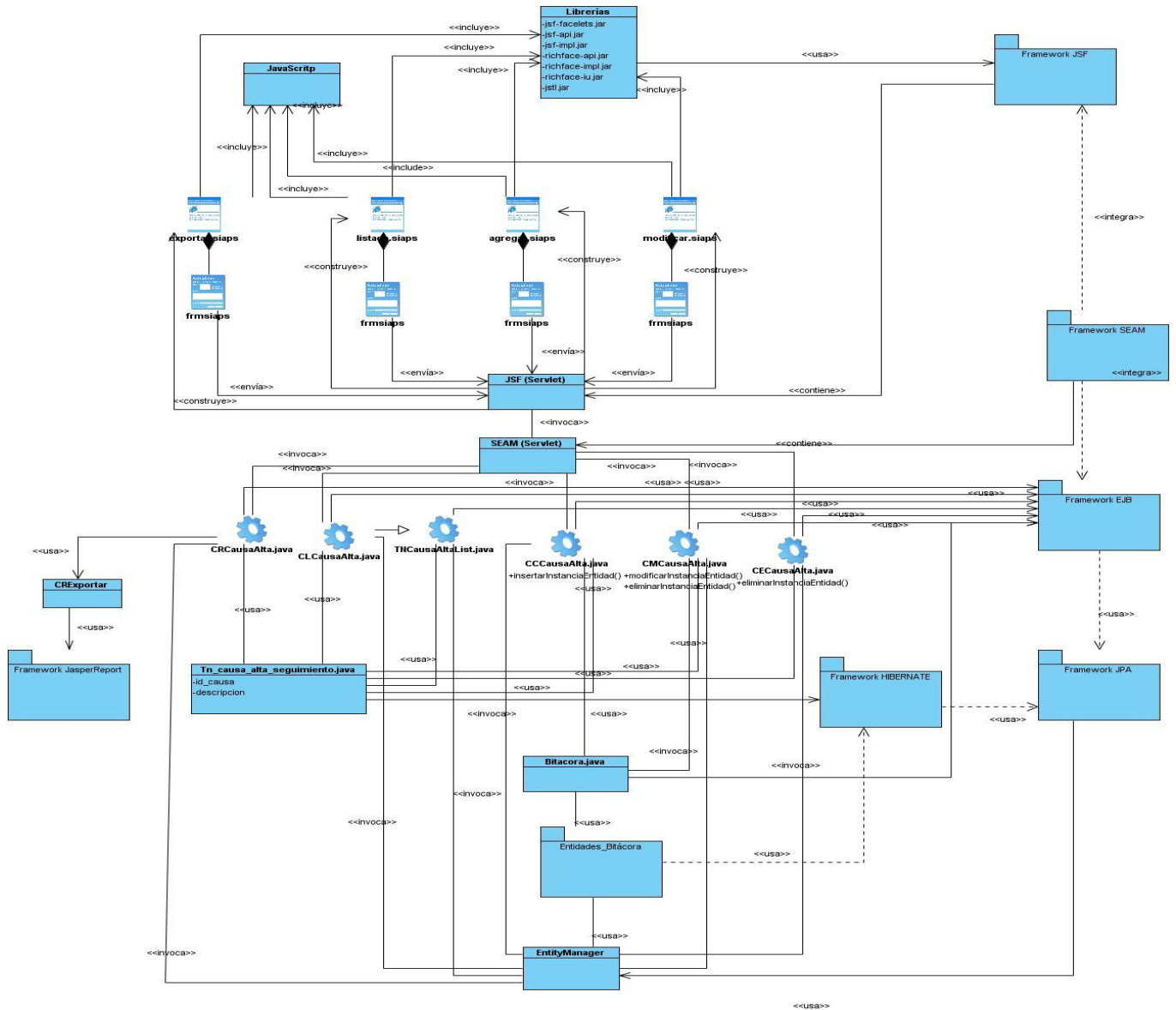
3.5. Diagramas de Clases del Diseño

El Diagrama de Clases es el artefacto principal en el diseño de un sistema, ya que brindan un mayor acercamiento a la forma y contenido de la solución propuesta. Este es usado para modelar la vista estática del sistema y contiene clases, interfaces, colaboraciones y relaciones. Son importantes no sólo para visualizar, especificar y documentar modelos estructurales, sino también para construir sistemas ejecutables, aplicando ingeniería directa e inversa. A continuación se muestran algunos de los diagramas de clases del diseño, para ver los restantes remitirse al expediente de proyecto.

Gestionar Duración de Problema



Gestionar Causa de Alta



3.5.1. Descripción de las Clases del Diseño

A continuación se describen algunas de las clases que fueron identificadas para la futura implementación del sistema, con el objetivo de lograr una mejor comprensión del funcionamiento que tendrá el mismo. Estas descripciones le permitirán entender mejor los diagramas mostrados anteriormente. Para ver la descripción del diagrama de clases del diseño Realizar Consulta así como de los diagramas restantes, remitirse al expediente de proyecto.

3.5.1.1. Gestionar Duración de Problema

3.5.1.1.1. Vistas

Nombre: listar.siaps

Propósito: Proveer la interacción con el usuario.

Descripción: La clase listar.siaps es una página web que se ejecuta del lado del cliente sobre un navegador. Permite listar los datos que están en la base de datos, además de mostrar información útil al usuario. Posee un conjunto de validaciones en JavaScript que permiten no realizar peticiones innecesarias y por lo tanto se incrementa su usabilidad. Utiliza diferentes librerías basadas en el Framework JSF. *Esta vista en particular le permite visualizar las Duraciones de Problema(descripción) que están en la base de datos mediante el llamado a la clase CLDuracionProblemaSalud, además de proporcionar las opciones Agregar, Modificar y Eliminar una Duración de Problema así como Exportar el listado.*

Tabla 3.1 Descripción de la vista Listar Duración de Problema.

Nombre: agregar.siaps

Propósito: Proveer la interacción con el usuario.

Descripción: La clase agregar.siaps es una página web que se ejecuta del lado del cliente sobre un navegador. Permite capturar los datos que serán persistidos en la base de datos, además de mostrar información útil al usuario. Posee un conjunto de validaciones en JavaScript que permite no realizar peticiones innecesarias y por lo tanto se incrementa su usabilidad. Utiliza diferentes librerías basadas en el Framework JSF. *Esta clase para persistir los datos (descripción de Duración de Problema) hace un llamado a la clase CCProblemaSalud.*

Tabla 3.2 Descripción de la vista Agregar Duración de Problema.

Nombre: modificar.siaps

Propósito: Proveer la interacción con el usuario.

Descripción: La clase modificar.siaps es una página web que se ejecuta del lado del cliente sobre un navegador. Permite capturar los datos que serán persistidos en la base de datos (*descripción de Duración de Problema*), además de mostrar información útil al usuario. Posee un conjunto de validaciones en Java Script que permiten no realizar peticiones innecesarias y por lo tanto se incrementa su usabilidad. Utiliza diferentes librerías basadas en el Framework JSF. *Esta clase para persistir los nuevos datos (descripción de Duración de Problema) hace un llamado a la clase CMProblemaSalud.*

Tabla 3.3 Descripción de la vista Modificar Duración de Problema.

3.5.1.1.2. Control

Nombre: CLDuracionProblemaSalud.java

Propósito: Proveer una respuesta a las peticiones realizadas en la vista.

Descripción: La clase CLDuracionProblemaSalud.java es una clase que se ejecuta del lado del servidor. Permite darle respuesta a las peticiones que se desencadenan en la vista a través de los métodos que contiene. Hace uso del Framework EJB que encapsula la lógica de negocio, integrándose con la vista a través del Framework SEAM. *Esta clase en particular permite listar todas las Duraciones de Problema (descripciones) existentes en la base de datos.*

Tabla 3.4 Descripción de la clase controladora Listar Duración de Problema.

Nombre: CCDuracionProblemaSalud.java

Propósito: Proveer una respuesta a las peticiones realizadas en la vista.

Descripción: La clase CCDuracionProblemaSalud.java es una clase que se ejecuta del lado del servidor. Permite darle respuesta a las peticiones que se desencadenan en la vista a través de los métodos que contiene. Hace uso del Framework EJB que encapsula la lógica de negocio, integrándose con la vista a través del Framework SEAM. *Esta clase en particular permite agregar una Duración de Problema (descripción) al Listado de Duración de Problema.*

Tabla 3.5 Descripción de la clase controladora Agregar Duración de Problema.

Nombre: CMDuracionProblemaSalud.java

Propósito: Proveer una respuesta a las peticiones realizadas en la vista.

Descripción: La clase CMDuracionProblemaSalud.java es una clase que se ejecuta del lado del servidor. Permite darle respuesta a las peticiones que se desencadenan en la vista a través de los métodos que contiene. Hace uso del Framework EJB que encapsula la lógica de negocio, integrándose con la vista a través del Framework SEAM. *Esta clase en particular permite modificar o eliminar la descripción de Duración de Problema seleccionado en el Listado de Duración de Problema.*

Tabla 3.6 Descripción de la clase controladora Modificar Duración de Problema.

Nombre: CRDuracionProblemaSalud.java

Propósito: Proveer una respuesta a las peticiones realizadas en la vista.

Descripción: La clase CRDuracionProblemaSalud.java es una clase que se ejecuta del lado del servidor. Permite darle respuesta a las peticiones que se desencadenan en la vista a través de los métodos que contiene. Hace uso del Framework EJB que encapsula la lógica de negocio, integrándose con la vista a través del Framework SEAM. *Esta clase en particular permite exportar a diferentes formatos (pdf, excel) el Listado de Duración de Problema.*

Tabla 3.7 Descripción de la clase controladora Exportar Listado de Duración de Problema.

3.5.1.1.3. Modelo

Nombre: TnDuracionProblema.java

Propósito: Proveer el mapeo con la base de datos.

Descripción: La clase TnDuracionProblemaNombre.java es una clase que se ejecuta del lado del servidor. Una entidad representa una tabla en el modelo de datos relacional y cada instancia de esta entidad corresponde a un registro en esa tabla. El estado de persistencia de una entidad se representa a través de campos persistentes o propiedades persistentes, *en este caso solo cuenta con los campos descripción (de la duración del problema), cid y versión.* Estos campos o propiedades usan anotaciones para el mapeo de estos objetos en el modelo de base de datos. Hace uso del Framework Hibernate y JPA.

Tabla 3.8 Descripción de la clase java Duración de Problema.

3.6. Modelo de Datos

El modelo de datos es uno de los artefactos más importante dentro del diseño. Este modelo proporciona una representación visual y física de los datos persistentes del sistema, que en el futuro serán la base de datos. Se obtiene a partir del diagrama de clases persistentes y su forma se expresa mediante un diagrama de UML, siendo sus elementos esenciales las entidades, atributos y relaciones entre entidades. Para ver el Modelo de Datos del sistema remitirse al expediente de proyecto.

3.6.1. Descripción de Tablas y Atributos

Describir las tablas y atributos que conforman el modelo de datos constituye un elemento esencial. El objetivo principal es hacer comprender con claridad cada uno de los elementos (entidades, atributos y relaciones) que conforman el modelo. En el expediente de proyecto podrá encontrar toda la información bien detallada sobre el modelo de datos así como las descripciones realizadas.

En este capítulo se detallaron los elementos del diseño quedando representados los resultados obtenidos durante el desarrollo del flujo de trabajo de diseño. Se definieron los diagramas de clases del diseño, los diagramas de interacción, la descripción de los patrones utilizados y el modelo lógico de datos así como la descripción de cada una de las clases del diseño; tablas y atributos de la base de datos con el objetivo de entender mejor el sistema, empleando para ello el Lenguaje unificado de Modelado (UML) y la herramienta case Visual Paradigm. Además, los artefactos obtenidos en el desarrollo del diseño propuesto constituyen la entrada fundamental para el desarrollo del Flujo de Trabajo de Implementación. A partir de este punto se puede comenzar a construir la aplicación, teniendo en cuenta todos los requerimientos y funcionalidades derivados de esta etapa.

Capítulo 4. Implementación

Este capítulo constituye la continuidad al Modelo de Diseño, en el se presenta el Diagrama de Despliegue de la solución propuesta, se describen los estándares de codificación y el tratamiento de errores en la aplicación.

4.1 Propuesta de Integración

En la actualidad ningún sistema se encuentra de forma aislada, por lo que se hace necesario pensar en su integración con otros sistemas en funcionamiento o en vías de desarrollo. El Componente de Consulta Clínica mantiene estrecha relación con los siguientes módulos: el módulo de Configuración, encargado de gestionar todas las configuraciones, de temas, roles, usuarios, funcionalidades así como la seguridad del sistema entre otras, brinda información útil y necesaria al resto de los módulos; el módulo de Medicina Familiar para obtener los datos relacionados a los pacientes como son: antecedentes patológicos familiares y otros; el módulo Medios Diagnósticos para solicitar la realización de exámenes complementarios y la obtención de los resultados, entre otros.

4.2. Diagrama de Despliegue

El Diagrama de Despliegue es un tipo de diagrama del Lenguaje Unificado de Modelado que se utiliza para modelar el hardware utilizado en las implementaciones de sistemas y las relaciones entre sus componentes. Los elementos usados por este tipo de diagrama son los nodos, componentes y asociaciones. El objetivo de los diagramas de despliegue es mostrar cómo y dónde se desplegará el sistema. Los mismos muestran las relaciones físicas de los distintos nodos que componen un sistema y el reparto de los componentes sobre dichos nodos. El Diagrama de despliegue de la solución propuesta es el siguiente:

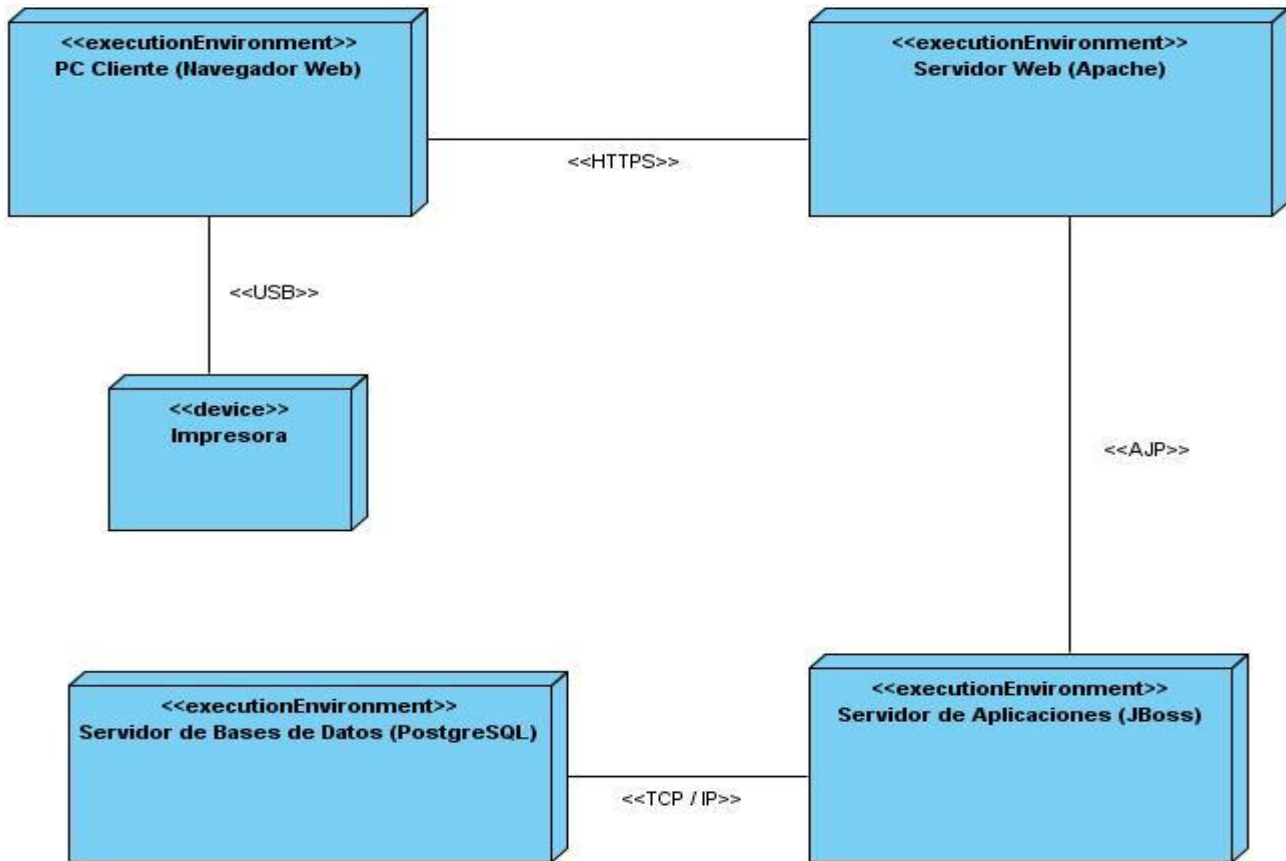


Figura 4.1 Diagrama de Despliegue.

Descripción de los Elementos

4.2.1.1. Dispositivos:

Servidor Web: Servidor Web Apache que provee el servicio de interfaz al usuario final mediante un portal convencional y otro WAP para dispositivos móviles, pues es el ordenador que estará como fachada a Internet, al mismo tiempo será el puente o proxy para entrar al clúster de servidor (res) de aplicaciones que proporciona Jboss. En escenarios híbridos brindará la ejecución y actualización de la solución local mediante la tecnología Java Web Start.

Servidor(es) de Aplicaciones: Servidor de Aplicaciones Jboss 4.2 certificado por SUN para el estándar JEE5, hospedará la solución integrada, proveerá de un clúster para balancear la carga de peticiones hechas por los usuarios garantizando de esta manera disponibilidad de la información mostrada.

Servidor de Bases de Datos: Servidor de Datos PostgreSQL 8.3, en el cual residirá toda la información operacional, que será alimento del repositorio central de historias clínicas electrónicas.

Estaciones Clientes: Serán las estaciones de los usuarios, las cuales servirán para acceder al sistema web o integrado mediante el navegador o al sistema local o de escritorio mediante un ícono en el escritorio del ordenador.

4.2.1.2. Procesadores:

Servidor(es) de Aplicacion(es) Recomendado(s):

Sistema Operativo Linux
2 DL380 G5,
Procesador Intel® Xeon® 5140 Dual - Core
4GB de memoria
2x72GB de disco

Servidor de Base de Datos Recomendado:

Sistema Operativo Linux
1 DL380 G5
Procesador Intel® Xeon® 5140 Dual – Core
4GB de memoria
2x72GB de disco

Servidor Web Recomendado:

Sistema Operativo Linux
1 DL380 G5
Procesador Intel® Xeon® 5140 Dual - Core
2 GB de memoria

2x72GB de disco

4.2.2. Descripción de Elementos e Interfaces de Comunicación

La interfaz de comunicación será IHE por brindar el marco técnico común para realizar la integración de sistemas de salud heterogéneos y los perfiles de integración correspondientes, de forma que se asegure la interoperabilidad y para el marcaje de los documentos clínicos HL7/CDA, ya que CDA es la arquitectura clínica de documentos de HL7 estandarizada y basado en XML que especifica la estructura y semántica de documentos clínicos para el propósito de facilitar su intercambio en un entorno de interoperabilidad.

4.3 Tratamiento de Excepciones

Las excepciones son situaciones que pueden presentarse y que requieren un tratamiento especial. Dichas situaciones pueden ocurrir dentro del programa durante su ejecución, pudiendo interrumpir el correcto funcionamiento del mismo.

Para el tratamiento de las excepciones en el sistema se usaron todas las potencialidades que brinda la plataforma Java en este aspecto. Con el uso de las diferentes tecnologías y la integración de las mismas es posible capturar y controlar situaciones anómalas en diferentes puntos de la aplicación. El framework Seam tiene un conjunto de elementos predefinidos que permiten el tratamiento de las excepciones desde las clases controladoras. Permite además, mediante el fichero de configuración page.xml, todo un flujo de navegación basado en excepciones. También se cuenta, en las páginas clientes, con un conjunto de componentes llamados validadores los cuales permiten establecer tipos de datos y formatos controlando que el envío de los activos al servidor sean los esperados.

A continuación se muestra la arquitectura que JPA tiene para el control de las excepciones y la cual se usa a todo lo largo de la implementación del sistema:

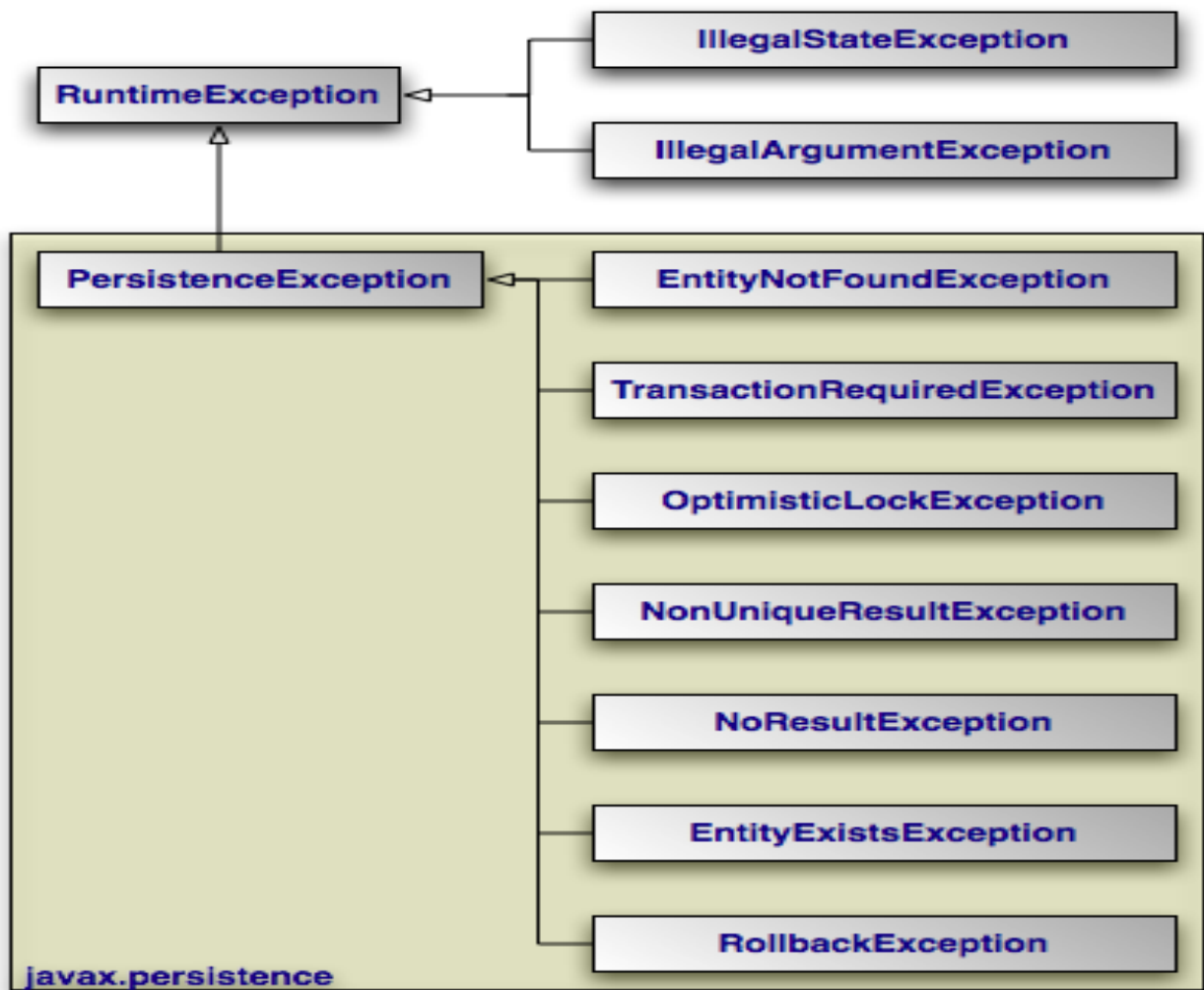


Figura 4.2 Arquitectura de JPA para tratamiento de excepciones.

4.4 Seguridad

Para que cualquier sistema informático sea calificado como seguro debe contar con un correcto equilibrio entre las siguientes características:

Integridad: La información sólo puede ser modificada por quien está autorizado y de manera controlada.

Confidencialidad: La información sólo debe ser legible para los autorizados.

Disponibilidad: La información debe estar disponible cuando se necesita.

Irrefutabilidad (No repudio): El uso y/o modificación de la información por parte de un usuario debe ser irrefutable, es decir, que el usuario no puede negar dicha acción.

En correspondencia con dichas características se plantean a continuación un conjunto de acciones que, llevadas a cabo, permitirá a los usuarios finales disfrutar de un software seguro:

- Se mantendrá la seguridad y el control a nivel de usuario (*Ver figura 4.3*), garantizando el acceso de los mismos sólo a los niveles establecidos de acuerdo a la función que realizan.
- Las contraseñas podrán cambiarse solo por el propio usuario o por el administrador del sistema.
- Se mantendrá un segundo nivel de seguridad a nivel de estaciones de trabajo, garantizando sólo la ejecución de las aplicaciones que hayan sido definidas para la estación en cuestión.
- Se registrarán todas las acciones que se realizan, llevando el control de las actividades de cada usuario en todo momento.
- Se establecerán mecanismos de control y verificación para los procesos susceptibles de fraude. Los mecanismos serán capaces de informar al personal autorizado sobre posibles irregularidades que den indicios sobre la introducción de información falseada.
- El sistema implementará un mecanismo de auditoría para el registro de todos los accesos efectuados por los usuarios, proporcionando un registro de actividades (log) de cada usuario en el sistema.
- Ninguna información que se haya ingresado en el sistema será eliminada físicamente de la BD, independientemente de que para el subsistema, este elemento ya no exista.
- Las informaciones médicas relacionadas con los pacientes y que vayan a ser intercambiadas con otros policlínicos por la red pública, viajarán cifradas para evitar accesos o modificaciones no autorizadas.

- El sistema permitirá la recuperación de la información de la base de datos a partir de los respaldos o salvadas realizadas.



The image shows a login screen for the 'alassIAPS' system. At the top, there is a green header with the logo and the text 'alassIAPS SISTEMA INTEGRAL PARA LA ATENCIÓN PRIMARIA DE SALUD'. Below the header, there are two input fields: one for 'Usuario' and one for 'Contraseña'. To the right of the 'Contraseña' field is a button labeled 'aceptar'.

Figura 4.3 Acceso al Sistema.

4.5 Estándares de Codificación

En la implementación de un sistema informático, estandarizar la codificación es muy importante pues por lo general ningún software lo mantiene toda su vida el autor original y el uso de estándares de codificación mejoran la lectura y comprensión del software, permitiendo entender código nuevo de forma más rápida. Por esta razón es necesario que, cada persona que escribe software siga la convención que estandariza la implementación del mismo. Teniendo en cuenta todo lo expuesto a continuación se plantean una serie de estrategias de codificación a utilizar para la implementación del Componente de Consulta Clínica del Subsistema Web del Sistema Integral para la Atención Primaria de Salud alas SIAPS.

4.5.1. Declaraciones de Clases e Interfaces.

La siguiente tabla describe las partes de la declaración de una clase o interface, en el orden en que deberían aparecer.

Número	Partes de la Declaración de una Clase o Interface	Descripción
1	Comentario de documentación de la clase o interface. (<i>/** ... */</i>)	
2	Sentencia class o interface.	
3	Comentario de implementación de la clase o interface si fuera necesario. (<i>/* ... */</i>)	Este comentario debe contener cualquier información aplicable a toda la clase o interface que no era apropiada para estar en los comentarios de documentación de la clase o interface.
4	Variables de clase (static).	Primero las variables de clase public, después las protected, después las de nivel de paquete (sin modificador de acceso), y después las private.
5	Variables de instancia.	Primero las public, después las protected, después las de nivel de paquete (sin modificador de acceso), y después las private.
6	Constructores.	
7	Métodos.	Estos métodos se deben agrupar por funcionalidad más que por visión o accesibilidad. Por ejemplo, un método de clase privado puede estar entre dos métodos públicos de instancia. El objetivo es hacer el código más legible y comprensible.

Indentación

La indentación tiene como objetivo lograr una estructura uniforme para los bloques de código así como para los diferentes niveles de anidamiento. Se deben emplear cuatro espacios como unidad de indentación (ver opción de formateo del eclipse). La construcción exacta de la indentación (espacios en blanco contra tabuladores) no se especifica. Los tabuladores deben ser exactamente cada 8 espacios.

Longitud de la Línea

Evitar las líneas de más de 80 caracteres, ya que no son manejadas bien por muchas terminales y herramientas.

Formatos de los Comentarios de Implementación

Los programas pueden tener cuatro estilos de comentarios de implementación: de bloque, de una línea, de remolque y de fin de línea.

Comentarios de Bloque

Los comentarios de bloque se usan para dar descripciones de ficheros, métodos, estructuras de datos y algoritmos. Los comentarios de bloque se podrán usar al comienzo de cada fichero o antes de cada método. También se pueden usar en otro lugares, tales como el interior de los métodos. Los comentarios de bloque en el interior de una función o método deben ser indentados al mismo nivel que el código que describen.

Un comentario de bloque debe ir precedido por una línea en blanco que lo separe del resto del código.

```
/*  
 * Aquí hay un comentario de bloque.  
 */
```

Los comentarios de bloque pueden comenzar con /*, que es reconocido por indent (1) como el comienzo de un comentario de bloque que no debe ser reformateado. Ejemplo:

```
/*-
 * Aqui tenemos un comentario de bloque con cierto
 * formato especial que quiero que ignore indent(1).
 *
 * uno
 *   dos
 *     tres
 */
```

Nota: Si no se usa indent (1), no se tiene que usar /* en el código o hacer cualquier otra concesión a la posibilidad de que alguien ejecute indent (1) sobre él.

Comentarios de una Línea

Pueden aparecer comentarios cortos de una única línea al nivel del código que siguen. Si un comentario no se puede escribir en una línea, debe seguir el formato de los comentarios de bloque. Un comentario de una sola línea debe ir precedido de una línea en blanco.

```
if (condicion) {
    /* Código de la condicion. */
    ...
}
```

Comentarios de Remolque

Pueden aparecer comentarios muy pequeños en la misma línea que describen, pero deben ser movidos lo suficientemente lejos para separarlos de las sentencias. Si más de un comentario corto aparece en el mismo trozo de código, deben ser indentados con la misma profundidad.

Aquí se muestra un ejemplo de comentario de remolque:

```
if (a == 2) {
    return TRUE;           /* caso especial */
} else {
    return isPrime(a);    /* caso gerenal */
}
```

Comentarios de Fin de Línea

El delimitador de comentario // puede convertir en comentario una línea completa o una parte de una línea. No debe ser usado para hacer comentarios de varias líneas consecutivas; sin embargo, puede usarse en líneas consecutivas para comentar secciones de código.

Aquí tienes ejemplos de los tres estilos:

```
if (foo > 1) {  
  // Hacer algo.  
  ...  
}  
else {  
  return false; // Explicar aqui por que.  
}  
//if (bar > 1) {  
//  
// // Hacer algo.  
// ...  
//}  
//else {  
// return false;  
//}
```

Con la culminación del presente capítulo se concluye con el flujo de trabajo de Implementación, cumpliéndose con el objetivo general de la investigación de desarrollar el Módulo Clínico Quirúrgico. Se describen los elementos necesarios para que el sistema sea desplegado así como los estándares de codificación usados durante la implementación del sistema y el manejo de excepciones en la aplicación.

Conclusiones

La realización del presente trabajo ha posibilitado de manera general cumplir con los objetivos propuestos determinándose la necesidad del desarrollo de un sistema informático para gestionar la información generada durante las consultas clínicas de APS, por lo que se pueden plantear las siguientes conclusiones:

- Los sistemas analizados para la gestión de la información relacionada con las consultas clínicas no satisfacen las necesidades actuales de la APS, por lo que se demostró la necesidad de desarrollar un sistema informático para facilitar la gestión de la información generada en las consultas clínicas de APS.
- El estudio de los modelos actuales de la Historia Clínica Individual, así como los demás documentos generados durante las consultas, permitió definir las funcionalidades del sistema, haciendo mayor énfasis en la Historia Clínica Orientada a Problemas.
- El modelo de Historia Clínica Orientada a Problemas es el más adecuado para registrar la información asociada a los pacientes en la APS.
- Se realizó la implementación de una aplicación web para facilitar la gestión de la información generada durante las consultas clínicas de APS.

Recomendaciones

Las recomendaciones de la investigación están dirigidas a sugerir acciones para complementar el producto obtenido en vista a mejorar el funcionamiento del sistema y con el fin de que se incrementen las funcionalidades brindadas para un mejor uso de éste. Por lo que para el buen desempeño y puesta en marcha de la aplicación se hacen las siguientes recomendaciones:

- Realizar un análisis de las principales consultas que se ejecutan en el entorno de la APS, para implementar un componente de consulta genérica, que tenga en cuenta la diversidad de información que se registra en este nivel de atención.
- Implementar la generación de Clinical Document Architecture (CDA) con el objetivo de añadir mayor interoperabilidad con otros sistemas y niveles de atención.
- Hacer uso de la Firma Digital.

Referencias Bibliográficas

- [1]. Centro de Ayuda bvscuba. [Disponible en: <http://bvsayuda.sld.cu/ayudas/manual/lilacs/conjunto-de-documentos-de-la-metodologia/manual-de-indizacion-de-documentos-para-la-base-de-datos-lilacs/descriptores/categoria-sp-salud-publica>]
- [2] Ídem a la referencia 1
- [3] Ídem a la referencia 1
- [4] **Colectivo de Autores.** *Burocracia en las consultas de Atención Primaria e informatización: ¿hemos mejorado?* Disponible en: [\[http://www.diclib.com/cgi-bin/d1.cgi?l=es&base=es_wiki_10&page=showid&id=41186\]](http://www.diclib.com/cgi-bin/d1.cgi?l=es&base=es_wiki_10&page=showid&id=41186)
- [5] Ídem a la referencia 4
- [6] **Ortega Benítez, A.** *Humanización en atención urgente. ¿Entendemos lo mismo sanitarios, pacientes y familia?* Disponible en: [\[http://www.semes.org/revista/vol16_1/12-16.pdf\]](http://www.semes.org/revista/vol16_1/12-16.pdf)
- [7] **Schelotto Felipe, Doménech Diana.** *Instituto de APS.* Facultad de Medicina. Agosto – Octubre 2006.
- [8] **Trincado Agudo, María Teresa y Bandera Sánchez.** *Desarrollo y perspectivas de la enfermería nefrológica.* Enero-Abril 2005.
- [9] **Moreno Rodríguez, Miguel Ángel.** *El Método Clínico.* Disponible en: [\[http://jornada.sld.cu/miguelconferencia.html\]](http://jornada.sld.cu/miguelconferencia.html)
- [10] Ídem a la referencia 9
- [11] *Medical informatics: computer applications in health care.* 1990: Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA.
- [12] Curso Universitario Sistemas de Información en los Sistemas de Salud. *Historia Clínica Electrónica.* Hospital Italiano de Buenos Aires.

REFERENCIAS BIBLIOGRÁFICAS

[13] Ídem a la referencia 12

[14] Ídem a la referencia 12

[15] Ídem a la referencia 12

[16] Ídem a la referencia 12

[17] **Frómeta Moreno, Yoiler, Brocard Delfino, Yusniel y Suárez Corrales, Maikel.** *Diseño y Servicios Web para el Registro de Población de la Atención Primaria del Sistema de Información para la Salud.* UCI. La Habana : s.n., 2007. Tesis.

[18] **Stusser Beltranena, R. J. y Colectivo de Autores.** (2006) Proyecto Vedado: salud-electrónica en la atención primaria de salud. Diseño y resultados iniciales. [Disponible en: Revista Cubana Medicina General Integral v.22 n.4 Ciudad de La Habana oct.-dic. 2006.

[19] **Delgado Ramos, A; Vidal Ledo, M** (2005) Informática en la salud pública cubana [Consultado el: 6 de febrero de 2010], [Disponible en: http://bvs.sld.cu/revistas/spu/vol32_3_06/spu15306.htm#cargo]

[20] Ídem a la referencia 3.

[21] Ídem a la referencia 3.

[22] Ídem a la referencia 3.

[23] Ídem a la referencia 3.

[24] Ídem a la referencia 3.

[25] **Sun Microsystems.** 1994-2009. [Citado el: 2 de 2 de 2010.] Disponible en: [<http://java.sun.com/javaee/technologies/javaee5.jsp>.]

[26] **Poza Bernal, Yanet, Pons Alonso, Ariel y Osés Sosa, Yoelvis.** *Registro de Actividades Diarias de la Atención Primaria de Salud.* UCI. La Habana : s.n., 2008. Tesis.

[27] *Guía Breve de XHTML.* Disponible en: [<http://www.w3c.es/Divulgacion/Guiasbreves/XHTML>]

REFERENCIAS BIBLIOGRÁFICAS

- [28] **Sun Microsystems.** [En línea]. [Citado el: 4 de febrero de 2010.] Disponible en: [\[http://developers.sun.com/docs/jscreator/help/jsp-jsfel/jsf_expression_language_intro.html\]](http://developers.sun.com/docs/jscreator/help/jsp-jsfel/jsf_expression_language_intro.html)
- [29] Community Documentation. *Hibernate.* Disponible en: [\[http://docs.jboss.org/hibernate/core/3.3/reference/en/html/queryhql.html\]](http://docs.jboss.org/hibernate/core/3.3/reference/en/html/queryhql.html)
- [30] **Rumbaugh James, Jacobson Ivar, Booch Grady.** *El Lenguaje Unificado de Modelado.* Manual de Referencia. Pearson Education. 2007. p18.
- [31] -EJB in Action
- [32] *The Seam Framework - Next generation enterprise Java development.* Disponible en: [\[http://www.seamframework.org/Home\]](http://www.seamframework.org/Home)
- [33] **Liu, Xinyu.** *Developing applications with Facelets, JSF and JS.* Disponible en: [\[http://today.java.net/pub/a/today/2006/08/29/developing-with-facelets-jsf-jsp.html\]](http://today.java.net/pub/a/today/2006/08/29/developing-with-facelets-jsf-jsp.html)
- [34] **Sánchez Suárez, José Manuel.** *Introducción a RichFaces.* Disponible en: [\[http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=richFacesJsfIntro\]](http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=richFacesJsfIntro)
- [35] *Mi primera hora con eclipse*
- [36] **Hernandis, José Alberto.** *Versión Cero.* [En línea]. [Citado el: 3 de febrero de 2010.] Disponible en: [\[http://www.versioncero.com/noticia/210/visual-paradigm-for-uml\]](http://www.versioncero.com/noticia/210/visual-paradigm-for-uml).
- [37] *APS_SIAPS_0116_Rneg.* Disponible en: [\[https://repositorio.cesim.prod.uci.cu/svn/aps/clinicoq/WEB/EXPEDIENTE%20DE%20PROYECTO/INGENIERIA/REQUISITOS/\]](https://repositorio.cesim.prod.uci.cu/svn/aps/clinicoq/WEB/EXPEDIENTE%20DE%20PROYECTO/INGENIERIA/REQUISITOS/)
- [38] **León Garcés, Johander y Jiménez Pomares, Yanies Maday.** *Registro de Problemas de Salud de la Atención Primaria (RPSAP) del Sistema de Información para la Salud.* UCI. La Habana : s.n., 2007. Tesis.
- [39] Ídem a la referencia 38.

REFERENCIAS BIBLIOGRÁFICAS

[40] **Jacobson, Ivar; Booch, Grady; Rumbaugh, James.** *El Proceso Unificado de Desarrollo de Software*. Addison Wesley. Madrid, Pearson Educación, 2000. 458.

[41] **Larman, Craig.** *UML y patrones Tomo I*. Prentice Hall Iberoamericana, 1999.499.

Bibliografía

1. Anales. *Comunicación informatizada entre un centro de salud y su hospital de referencia* [En línea] [Citado el: 13 de 01 de 2010.] Disponible en: [http://www.cfnavarra.es/salud/anales/textos/vol24/n2/orig2a.html]
2. *A vueltas con la presión asistencial* [En línea] [Citado el: 13 de 12 de 2009.] Disponible en: [http://aplamancha.blogspot.com/2009/06/vueltas-con-la-presion-asistencial.html]
3. *APS_SIAPS_0116_Rneg.* Disponible en: [https://repositorio.cesim.prod.uci.cu/svn/aps/clinicoq/WEB/EXPEDIENTE%20DE%20PROYECTO/INGENIERIA/REQUISITOS/]
4. **Bermúdez, Javier.** *Utilizando BPM.* [En línea] 2009. [Citado el: 24 de 1 de 2010.] Disponible en: [http://hpmslab.ing.puc.cl/LinkClick.aspx?fileticket=i9q%2B3gJIn5g%3D&tabid=305&mid=769]
5. **Castillo Reyes, Odalis Amparo y Sánchez Prida, Elizabeth.** *Impacto del programa de Informatización en el Policlínico - Area Olivos I. Sancti Spiritus.* [En línea] [Citado el: 12 de 01 de 2010.] Disponible en: [http://jornada2009.sld.cu/index.php/Jornada/2009/paperviewDownloadInterstitial/46/62]
6. Centro de Ayuda bvscuba. [Disponible en: http://bvsayuda.sld.cu/ayudas/manual/lilacs/conjunto-de-documentos-de-la-metodologia/manual-de-indizacion-de-documentos-para-la-base-de-datos-lilacs/descriptores/categoria-sp-salud-publica]
7. Centro Médico Loreto. [En línea] [Citado el: 6 de 12 de 2009.] Disponible en: [http://www.loretocentromedico.com/09/index.php?option=com_content&view=article&id=9&Itemid=57]
8. **Colectivo de Autores.** *Burocracia en las consultas de Atención Primaria e informatización: ¿hemos mejorado?* Disponible en: [http://www.diclib.com/cgi-bin/d1.cgi?l=es&base=es_wiki_10&page=showid&id=41186]
9. **Contreras, José L.** *La Importancia De Las Tic en Aps.* [Disponible en: http://www.slideshare.net/guest46cb9a/la-importancia-de-las-tic-en-aps]

10. Community Documentation. *Hibernate*. Disponible en:[<http://docs.jboss.org/hibernate/core/3.3/reference/en/html/queryhql.html>]
11. Curso Universitario Sistemas de Información en los Sistemas de Salud. *Historia Clínica Electrónica*. Hospital Italiano de Buenos Aires.
12. **Delgado Ramos, A; Vidal Ledo, M** (2005) Informática en la salud pública cubana [Consultado el: 6 de febrero de 2010], [Disponible en: http://bvs.sld.cu/revistas/spu/vol32_3_06/spu15306.htm#cargo]
13. *EJB in Action*
14. **Frómata Moreno, Yoiler, Brocard Delfino, Yusniel y Suárez Corrales, Maikel**. *Diseño y Servicios Web para el Registro de Población de la Atención Primaria del Sistema de Información para la Salud*. UCI. La Habana : s.n., 2007. Tesis.
15. **Gimeno, Andrés**. Revista española de Economía de la Salud. *A fondo*. [En línea] [Citado el: 10 de 12 de 2009. Disponible en: [http://www.economiadelasalud.com/Ediciones/07/07afondo/07a_fondo.htm]
16. *Guía Breve de XHTML*. Disponible en: [<http://www.w3c.es/Divulgacion/Guiasbreves/XHTML>]
17. **Hernandis, José Alberto**. *Versión Cero*. [En línea]. [Citado el: 3 de febrero de 2010.] Disponible en: [<http://www.versioncero.com/noticia/210/visual-paradigm-for-uml>].
18. **Hookom, Jacob**. *Inside Facelets Part 1: An Introduction*. [En línea]. [Citado el: 3 de febrero de 2010] Disponible en: [http://www.jsfcentral.com/articles/facelets_1.html]
19. **Jacobson, Ivar; Booch, Grady; Rumbaugh, James**. *El Proceso Unificado de Desarrollo de Software*. Addison Wesley. Madrid, Pearson Educación, 2000. 458.
20. *LA CALIDAD DE LA ASISTENCIA PEDIÁTRICA DE ATENCIÓN PRIMARIA EN LOS SISTEMAS SANITARIOS PÚBLICOS ESPAÑOLES*. [En línea] [Citado el: 8 de 12 de 2009.] Disponible en: [<http://www.pediatrasandalucia.org/Docs/Docencia/ConsensoCalidadPediatriaAPAEP.pdf>]
21. **Larma, Craig**. *UML y patrones Tomo I*. Prentice Hall Iberoamericana, 1999.499.

22. **León Garcés, Johander y Jiménez Pomares, Yanies Maday.** *Registro de Problemas de Salud de la Atención Primaria (RPSAP) del Sistema de Información para la Salud.* UCI. La Habana : s.n., 2007. Tesis.
23. **Liu, Xinyu.** *Developing applications with Facelets, JSF and JS.* Disponible en: [<http://today.java.net/pub/a/today/2006/08/29/developing-with-facelets-jsf-jsp.html>]
24. *Mi primera hora con eclipse*
25. **Marín Díaz, Miguel E.** *Capacitación en el área de la Informática en Salud.* Revista cubana de Informática Médica. [Disponible en: http://www.cecam.sld.cu/pages/rcim/revista_10/articulos_hm/capacitacion.htm]
26. *Medical informatics: computer applications in health care.* 1990: Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA.
27. Medicina y Melodía. *Puntos claves de la Atención Primaria de Salud* [En línea] [Citado el: 12 de 01 de 2010.] Disponible en: [<http://medymel.blogspot.com/2009/05/puntos-clave-de-la-atencion-primaria-de.html>]
28. **Minho Conill, E y Rodrigues Fausto, Marcia C.** *Análisis de la Integración de la Atención Primaria de Salud en la Red de Servicios en Europa y América Latina.* Marzo. 2009.
29. **Moreno Rodríguez, Miguel Ángel.** *El Método Clínico.* Disponible en: [<http://jornada.sld.cu/miguelconferencia.html>]
30. **Ortega Benítez, A.** *Humanización en atención urgente. ¿Entendemos lo mismo sanitarios, pacientes y familia?* Disponible en: [http://www.semes.org/revista/vol16_1/12-16.pdf]
31. **Pompa Sourd, Frank.** *Memorias II Congreso Latinoamericano de Ingeniería Biomédica, Habana 2001, Mayo 23-25, La Habana, Cuba. Sistema Informático para la APS (APUS).*
32. **Poza Bernal, Yanet, Pons Alonso, Ariel y Osés Sosa, Yoelvis.** *Registro de Actividades Diarias de la Atención Primaria de Salud.* UCI. La Habana : s.n., 2008. Tesis.

33. **Ramos, Juan Alonso.** Adictos al trabajo. [En línea]. [Citado el: 3 de febrero de 2010.] Disponible en: [http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=Ajax4Jsf].
34. RichFaces. Disponible en: [http://www.juntadeandalucia.es/xwiki/bin/view/MADEJA/RichFaces#ventajas]
35. **Rumbaugh James, Jacobson Ivar, Booch Grady.** *El Lenguaje Unificado de Modelado.* Manual de Referencia. Pearson Education. 2007. p18.
36. **Sánchez Suárez, José Manuel.** *Introducción a RichFaces.* Disponible en: [http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=richFacesJsflIntro]
37. **Schelotto Felipe, Doménech Diana.** *Instituto de APS.* Facultad de Medicina. Agosto – Octubre 2006.
38. **Stusser Beltranena, Rodolfo J. y Rodríguez Díaz, Alfredo.** *La informatización de la atención primaria de salud. Revista Cubana de Medicina General Integral.* 2006. [Disponible en: http://bvs.sld.cu/revistas/mgi/vol22_4_06/mgi12406.htm#cargo]
39. **Suárez González, Héctor.** *Manual Hibernate.* Disponible en: [http://www.javahispano.org/contenidos/es/manual_hibernate/]
40. **Sun Microsystems.** [En línea]. [Citado el: 4 de febrero de 2010.] Disponible en: [http://developers.sun.com/docs/jscreator/help/jsp-jsfel/jsf_expression_language_intro.html]
41. **Sun Microsystems.** 1994-2009. [Citado el: 2 de 2 de 2010.] Disponible en: [http://java.sun.com/javaee/technologies/javaee5.jsp.]
42. *The Seam Framework - Next generation enterprise Java development.* Disponible en: [http://www.seamframework.org/Home]
43. **Trincado Agudo, María Teresa y Bandera Sánchez.** *Desarrollo y perspectivas de la enfermería nefrológica.* Enero-Abril 2005.

Glosario de Términos

Aplicación o Sistema Informático: Programas con los cuales el usuario final interactúa a través de una interfaz y que realizan tareas útiles para éste.

Componente: Parte física y reemplazable de un sistema que se ajusta a, y proporciona la realización de, un conjunto de interfaces.

Informática: Disciplina que estudia el tratamiento automático de la información utilizando dispositivos electrónicos y sistemas computacionales.

Interoperabilidad: Condición necesaria para que los usuarios (humanos o mecánicos) tengan un acceso completo a la información disponible. Entre las iniciativas recientes más destacadas para dotar a la Web de interoperabilidad se encuentran los servicios Web y la Web semántica.

Policlínico: Unidad de salud donde se brindan servicios médicos a una población geográficamente determinada perteneciente al nivel asistencial de APS.

Servicio: Unidad de software que encapsula alguna funcionalidad de negocio y proporciona estas a otros servicios a través de interfaces públicas bien definidas.

Software: Conjunto de programas y procedimientos necesarios para hacer posible la realización de una tarea específica, en contraposición a los componentes físicos del sistema.

Software Libre: Es el software que, una vez obtenido, puede ser usado, copiado, estudiado, modificado y redistribuido libremente.

Subsistema: Agrupación de elementos, de los que algunos constituyen una especificación del comportamiento ofrecido por los elementos contenidos.