



UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS

FACULTAD 7

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

**Título: Componente web Atención al paciente diabético del
Sistema Integral para la Atención Primaria de Salud**

Autores:

Néstor Brito Medina

Klaudia Reyes Castro

Tutores:

Ing. Yosleiby Izquierdo Sánchez

Ing. Humberto Ruiz Salcedo

La Habana, junio del 2011

“Año 53 de la Revolución”

Datos de Contacto

Ing. Yosleiby Izquierdo Sánchez - (yizquierdo@uci.cu): Graduado de Ingeniero en Ciencias Informáticas en el año 2008. Actualmente labora en la Universidad de las 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.

Ing. Humberto Ruiz Salcedo – (hruiz@uci.cu): Graduado de Ingeniero en Ciencias Informáticas en el año 2010 con Título de Oro. Labora en la Universidad de las Ciencias Informáticas (UCI), desempeñándose como profesor de la Facultad No. 7 vinculado a la producción en el Centro de Informática Médica (CESIM). Actualmente es parte del proyecto Sistema Integral para la Atención Primaria de Salud (SIAPS) donde se desarrolla como jefe de Equipo de Desarrollo.

Resumen

El departamento de Atención Primaria de Salud (APS) del Centro de Informática Médica (CESIM) está desarrollando el Sistema Integral para la Atención Primaria de Salud (alás SIAPS). Dentro de sus objetivos se encuentra la informatización del Sistema Nacional de Salud específicamente la consulta integral al paciente diabético para la APS ya que esta juega un papel primordial en las acciones de promoción de salud que preconiza a estilos de vida saludables.

A raíz de la importancia de la información que se genera en la consulta integral al paciente diabético y a que actualmente no existe ningún sistema informático que lleve todo el control de la misma, se propone el desarrollo del componente web Atención al paciente diabético del Sistema Integral para la APS que facilite la gestión de la información en esta, con el fin de garantizar la seguridad en el manejo de los datos generados durante las consultas, garantizando con esto un mayor bienestar para el paciente.

Para el desarrollo del sistema se usaron tecnologías libres y multiplataforma. Como lenguaje de programación Java, Eclipse SDK en su versión 3.5.2, PostgreSQL en su versión 8.4, se utilizó UML, sustentado por la herramienta CASE Visual Paradigm en su v6.4 y JBoss Application Server en su versión 4.2.2.

PALABRAS CLAVE: Atención Primaria de Salud, atención integral al paciente diabético, componente web.

Tabla de Contenidos

Introducción.....	6
Capítulo 1. Fundamentación Teórica	10
1.1 Marco Conceptual	10
1.2 Atención Integral al Diabético	10
1.3 Estado del Arte.....	14
1.4 Situación Problémica	19
1.5 Tecnologías informáticas.....	20
1.6 Estilos Arquitectónicos.....	20
1.7 Lenguajes de programación	22
1.8 Metodología de desarrollo	23
1.9 Gestor de Base de Datos	24
2 Servidor de aplicación.....	24
2.1 JBoss 4.2.2	24
2.2 Frameworks	25
3 Herramientas informáticas.....	26
3.1 Eclipse Ganymede 3.5.2.....	26
3.2 Visual Paradigm 6.4.....	26
Capítulo 2. Características del Sistema	28
2. Modelo del Negocio	28
2.2 Diagramas de procesos del negocio.	28
2.3 Descripción textual de los procesos de negocio.....	31
2.4 Propuesta del Sistema.....	33
Capítulo 3. Diseño del Sistema	39
3 Modelos de Diseño.	39
3.1. Patrones de Diseño.	39
3.2. Diagramas de Diseño.....	40
Capítulo 4. Implementación	58
4.1. Propuesta de Integración del componente web Atención al Paciente Diabético.	58

TABLA DE CONTENIDOS

4.2. Diagrama de Despliegue.	58
4.3. Propuesta de Seguridad.	61
4.4. Implementación.....	62
4.5. Tratamiento de Excepciones.	63
4.6. Descripción de elementos e interfaces de comunicación.....	63
4.7. Estándares de codificación y tratamiento de excepciones.	63
Conclusiones.....	70
Recomendaciones	71
Referencias Bibliográficas	72
Bibliografía	75
Glosario de Términos.....	79
Anexos	80

Introducción

La sociedad vive un momento de transición profunda entre una humanidad de corte industrial y otra marcada por el proceso de desarrollo de la información y las telecomunicaciones. El uso de la informática incide de manera directa e indirecta en todas sus esferas; rediseña la realidad, ya que evoluciona la forma en la que se producen y distribuyen los bienes y servicios, las relaciones internacionales, e incluso puede modificar la forma en la que se interrelacionan y divierten los individuos de ella. Para tener una idea de la magnitud de su impacto en la vida de las personas, se puede comparar con dos transformaciones que contribuyeron a hacer de la civilización lo que es hoy: la revolución agrícola y la revolución industrial.

La difusión masiva de las tecnologías de la información y las comunicaciones en los últimos años del siglo XX ha generado la llamada revolución informática. Esta revolución se conoce como sociedad de la información, dando origen a una nueva época. Una sociedad para ser más poderosa y profesional debe aprovechar la informatización en todas sus esferas y estar convencida de que para los países subdesarrollados resulta indispensable el logro de este propósito, puesto que su principal objetivo es lograr la conservación de su pueblo.

Uno de los aspectos fundamentales y necesarios en la sociedad, para brindar un adecuado servicio en la atención médica, es la disponibilidad de la información en el sector de la salud. El procesamiento de estos sistemas informáticos con el manejo de las nuevas tecnologías y las comunicaciones permite la mejora de todos los procesos con la finalidad de facilitar excelentes servicios de salud y contribuir de esta forma a lograr el progreso en la salud individual, colectiva y una elevada calidad de vida a la población.

El manejo efectivo de la información y las tecnologías informáticas proporcionan a los profesionales y técnicos de las ciencias médicas cubana la posibilidad de elevar la calidad de la investigación en la Atención Primaria de Salud (APS); la cual es el nivel básico de atención, que garantiza la globalidad y continuidad a lo largo de toda la vida del paciente, actuando como gestor, coordinador de casos y regulador de flujos.

La prevención es una de las formas más efectivas para mantener altos niveles de salud en la comunidad, ya que su objetivo es evitar el desarrollo de enfermedades o problemas de salud en un sector habitacional en específico de ahí la importancia de realizar el pesquiasaje para poder actuar sobre los factores de riesgo, tanto en el ámbito personal como el familiar, ya que el diagnóstico temprano, preventivo, la

identificación y modificación de estilos de vida inadecuados garantizarán el control y evitarán complicaciones.

El Pesquisaje constituye en la actualidad una tarea de primer orden ya que consiste en detectar a tiempo enfermedades que se puedan desarrollar en la adultez. La práctica de esta se ha naturalizado en la APS como componente primordial para realizar acciones de prevención, permitiendo el seguimiento integral de los individuos y la evaluación de su comportamiento durante un período de tiempo.

El intento de prevenir y curar las enfermedades mediante el control de los factores de riesgo parece ser en el momento actual el arma más poderosa que existe contra distintas afecciones crónicas. Ya que actualmente se realizan acciones preventivas con el fin de combatir disímiles padecimientos como es la Diabetes Mellitus (DM), a través de las pesquisas.

La DM o enfermedad de la orina dulce, se ha convertido en una epidemia mundial (de la cual no está exenta Cuba) debido a que constituye una de las causas fundamentales de morbilidad y mortalidad. Puede llevar al paciente a una insuficiencia renal crónica terminal, ceguera adquirida, a la amputación no traumática y es, además, uno de los primeros cuatro factores de riesgo clásico para aterosclerosis y problemas coronarios. [1] Es una enfermedad crónica que se caracteriza por niveles de glucosa en sangre elevados (glucemia), como consecuencia de un déficit en la secreción de insulina, acompañado, en muchos casos, de una disminución de la respuesta de las células del organismo al efecto de la insulina, de no ser tratada adecuadamente, puede llevar a complicaciones crónicas que afecten la calidad de vida de quienes la padecen.

El número de personas con DM ha ido en aumento en todos los países, desarrollados y no desarrollados. La Federación Internacional de Diabetes (FID) calculó que en el año 2004 existía en el mundo alrededor de 194 millones de personas con DM. Y se calcula que para el año 2025 esta cifra llegue a los 333 millones de personas, en Cuba se calcula un aproximado de 350 000 personas que padecen de DM. [2] Por esta razón, existen en todos los centros de salud del país una consulta integral al paciente diabético, donde radica un equipo multidisciplinario de la salud. Integrado por un médico con diplomado en diabetes, una enfermera, y un podólogo, los cuales atienden a todos los pacientes con diabetes diagnosticada por su médico de familia. Los especialistas que atienden al paciente diabético tienen como tareas fundamentales brindar orientación en cuanto a su dieta, cómo debe afrontar una hiperglucemia, y tratar aquellos que presenten complicaciones de Pie Diabético.

A nivel nacional e internacional existen dificultades con la gestión de la información en la APS, que faciliten el seguimiento y la evaluación del comportamiento del paciente en la comunidad y le posibilite al médico tener una concepción más amplia de los estudios del paciente con un enfoque integral, con el fin de diagnosticar, rehabilitar e indicarle tratamientos más certeros, teniendo en cuenta los antecedentes y el entorno donde el paciente convive.

Los especialistas registran manualmente la información relacionada con la DM, lo que trae consigo que se acumule y deteriore. Esto provoca que se vea afectado el trabajo del personal de salud en la atención a pacientes, la detección de enfermedades y clasificación de la DM, así como en la remisión, ya que en la actualidad el proceso se realiza por parte del paciente, el cual es el que realiza los trámites para reservar la cita en el centro al que fue remitido incurriendo esto en una sobrecarga innecesaria en el paciente. La información del historial del paciente diabético y su evaluación en ocasiones se pierde lo que dificulta la comparación y asociación de problemas de salud su derivación, motivo y causa.

El sistema de salud actual se ve limitado por la ausencia de herramientas de apoyo al trabajo diario del médico, ya que muchas veces se cometen varios errores al no tener una buena codificación de los términos que se utilizan. Al mismo tiempo, al realizar los diagnósticos de salud sin regirse por estos estándares internacionales impiden el establecimiento de un lenguaje común entre las organizaciones mundiales.

Como resultado del análisis del proceso de atención integral al diabético en el área de salud en la atención Primaria de Salud se identifica como **Problema a Resolver**: ¿cómo facilitar la gestión de la información relacionada con la atención integral al paciente diabético en la APS en Cuba?

El **Objeto de Estudio** se enmarca en el proceso de gestión de la información asistencial de pacientes en la Atención Primaria de la Salud en Cuba. El objeto delimita como **Campo de Acción** el proceso de gestión de la información de la Atención integral al paciente diabético en la atención primaria de salud en Cuba.

Para dar solución a la problemática existente se ha determinado como **Objetivo General** desarrollar el componente Web de Atención al Paciente Diabético del Sistema Integral para la Atención Primaria de Salud, que facilite la gestión de la información de la atención al paciente diabético en la Atención Primaria de Salud en Cuba.

Para dar cumplimiento al objetivo general se han definido las siguientes **Tareas de Investigación**:

1. Analizar los procesos de gestión de la información asistencial de pacientes en la Atención Primaria de la Salud en Cuba.
2. Valorar la arquitectura J2EE 5.0, aprobada por el Centro de Informática Médica (CESIM) para el desarrollo de sus aplicaciones.
3. Identificar las tendencias actuales a nivel nacional e internacional relacionadas con la atención integral al paciente Diabético en la Atención Primaria de Salud.
4. Obtener mediante la guía de desarrollo propuesta por el proceso de mejora los artefactos correspondientes a las fases: “Modelado del Negocio”, “Gestión de Requerimientos”, “Análisis y Diseño” e “Implementación”.
5. Implementar las funcionalidades de los procesos Atención integral al paciente diabético y Pesquisaje.

El documento está estructurado en cuatro capítulos:

CAPÍTULO 1. Fundamentación Teórica: contiene los aspectos esenciales para entender el entorno del problema a resolver. Se describen los conceptos fundamentales asociados al dominio del problema, los aspectos generales del Sistema Nacional de Salud, el estado del arte de los sistemas informáticos relacionados con la atención a los pacientes diabéticos.

CAPÍTULO 2. Características del Sistema: presenta los argumentos principales de los procesos vinculados al problema a resolver, como son la Captura de Requerimientos, se especifican los requerimientos funcionales y no funcionales, el modelado del sistema y el Modelado del Negocio.

CAPÍTULO 3. Diseño del Sistema: muestra los elementos básicos del diseño del sistema, mediante la justificación del uso de patrones, los diagramas de clases del diseño y las descripciones de las clases más significativas del diseño.

CAPÍTULO 4. Implementación: describe los componentes fundamentales de todo el proceso de implementación, argumentando la justificación de la integración con otros módulos, los métodos, agentes, estándares de diseño, codificación y excepciones, así como la representación general del diagrama de componente y de despliegue.

Capítulo 1. Fundamentación Teórica

Para obtener una mayor comprensión de la situación planteada anteriormente se abordarán los conceptos necesarios en este capítulo. Se describe además el estado del arte en cuanto a sistemas existentes, tecnologías y herramientas a nivel nacional e internacional para el desarrollo del sistema propuesto. También se describen las técnicas, herramientas, metodologías y tecnologías utilizadas para realizar el proceso de desarrollo.

1.1 Marco Conceptual

Sistema Nacional de Salud: de acuerdo con la complejidad de las acciones preventivas, curativas y de rehabilitación, así como la especialización de los servicios de salud brindados, los diferentes niveles de atención médica se han organizado en: [3]

- ✓ **Atención Primaria de Salud (APS):** Da solución aproximadamente al 80 % de los problemas de salud de la población y que correspondan con las acciones de promoción y protección de la salud. Aunque sus actividades se realizan en cualquier unidad del Sistema Nacional de Salud (SNS), están relacionados fundamentalmente con las que se realizan en clínicas urbanas o rurales, dispensarios y postas médicas.
- ✓ **Atención Secundaria de Salud:** 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, pero también desempeña funciones de rehabilitación, promoción y prevención de la salud. Se llevan a cabo acciones de salud más complejas y especializadas. Comprende la atención médica brindada en los distintos Hospitales.
- ✓ **Atención Terciaria de Salud:** 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 en los mismos y el desarrollo de la investigación. A este nivel pertenecen los institutos y hospitales especializados.

1.2 Atención Integral al Diabético

Atención integral: la atención integral consiste en abordar las necesidades y problemas de salud, en los individuos, familias o comunidades, en una sociedad concreta, que define las condiciones de vida, perfilando un determinado nivel de salud en los grupos sociales que la conforman. Por lo tanto, posee

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

características propias del sexo, de la etnia y género; su propia historia de vida; una particular condición social y económica que caracteriza su entorno, sus condiciones de vida y desde ahí, sus particulares necesidades de salud. [37]

La atención integral busca satisfacer las necesidades y expectativas de salud de la población, con enfoque individual, familiar y comunitario, en la secuencia técnicamente correcta y mediante mejoras en la organización de los procesos y métodos de trabajo, es por esto que en Cuba se creó la atención integral al paciente diabético, facilitándole el manejo de la enfermedad a los pacientes.

El Sistema Nacional de Salud (SNS) en Cuba no ha estado ajeno dentro de este contexto. En los albores de 1970, el Instituto Nacional de Endocrinología (INEN) creó el Centro de Atención al Diabético (CAD), primero de su tipo en América Latina, y cuya función principal es la educación diabetológica. [4] Además de los CAD, también se han creado una serie de consultas especializadas de atención al diabético en la APS, ya que esta juega un papel fundamental en las acciones de promoción de salud que preconiza a estilos de vida saludables, de prevención primaria así como de acciones de pesquisa activa de la enfermedad y de sus complicaciones agudas y crónicas.

La educación diabetológica y el control de la glicemia y los factores de riesgo vascular constituyen elementos esenciales de la estrategia del programa; tiene como objetivos disminuir el impacto de la DM sobre la población, disminuir la morbilidad por DM, reducir la frecuencia y severidad de las complicaciones agudas y crónicas de la DM y mejorar la calidad de la vida de las personas con DM. [5]

El servicio de atención integral al paciente diabético en la APS, tiene como objetivo principal garantizar la atención integral al paciente diabético en el área de salud, disminuir la incidencia de la Diabetes Mellitus (DM) en la comunidad, realizar el pesquisaje activo y sistemático de la DM, ejecutar y dirigir el cuidado y educación continuada al diabético en la atención primaria de salud, así como prevenir, detectar y tratar adecuadamente las complicaciones de la Diabetes Mellitus.

El proceso de pesquisaje activo no es más que una investigación o indagación llevada a cabo por el Equipo Básico de Salud (EBS) hacia los pacientes para descubrir alguna enfermedad o para obtener información de este para un seguimiento de su salud, también para que el médico lleve constancia de los hábitos en el hogar del personal que atiende y así saber las enfermedades que están más propensas a padecer ya que con el estudio previo de estos se puede evitar ciertas enfermedades y hasta detectar a tiempo como es la DM.

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

Para dar cumplimiento a estos objetivos se han creado consultas especializadas de atención integral al diabético en cada área de salud, para complementar y brindar un mayor apoyo a las ya existentes en el consultorio del médico de la familia. En estas consultas se realizan las tareas de: organización de forma permanente de pesquisa activa de DM, educación terapéutica continuada al paciente diabético, lograr el autocontrol metabólico por parte del paciente, garantizar un tratamiento específico, mediante compuestos orales y/o insulina.

La detección de la DM por parte de los especialistas puede estar dada de distintas formas, principalmente por el pesquiasje activo de los médicos de la familia en la población, explicado anteriormente, o por el pesquiasje pasivo que es la presencia de los factores de riesgo en un paciente que acuda a la consulta, haciéndose éste la prueba de tolerancia a la glucosa y tenga un resultado positivo.

La DM no se comporta de igual forma en todos los pacientes, por lo que ha sido necesario clasificarla en cuatro tipos de diabetes mellitus:

- Diabetes tipo 1.
- Diabetes tipo 2.
- Diabetes gestacional.
- Otros tipos de diabetes.

Diabetes Mellitus Tipo 1

Es aquella en la que el déficit de insulina es absoluto. La susceptibilidad de desarrollar la diabetes tipo 1 se transmite de manera genética. En esta entidad, la herencia predispone a padecerla, pero se requiere la interacción de factores ambientales para que se manifieste clínicamente. [6] Hasta el momento los factores ambientales que más se vinculan son los virales, dietéticos, psicosociales y tóxicos. Este tipo de diabetes es más frecuente en niños y jóvenes, pero puede aparecer a cualquier edad. Las personas que padezcan este tipo de diabetes, requerirán tratamiento con insulina durante toda la vida, de lo contrario incurrirían en una complicación de la diabetes y podría ocasionarles la muerte.

Diabetes Mellitus Tipo 2

Es la más frecuente y constituye un 90% de las personas diagnosticadas con esta entidad. Generalmente se presenta en personas mayores de 40 años, aunque también puede observarse en edades más

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

tempranas. [7] La obesidad es una de las causas desencadenantes de esta enfermedad y también la causa de la resistencia a la insulina. Se transmite genéticamente pero de manera diferente a la de tipo 1.

Diabetes Gestacional.

Es aquella que aparece en las mujeres durante la gestación y al terminar esta, desaparece. Toda mujer evaluada con este diagnóstico es reevaluada 6 semanas después del parto para realizarse una prueba de tolerancia a la glucosa y determinar si el problema de salud persiste o no.

Otros tipos de diabetes.

A este grupo pertenecen las hiperglucemias crónicas causadas por enfermedades endocrinas, pancreáticas, inducidas por drogas o medicamentos, infecciones y algunas enfermedades genéticas. Ejemplos de ellas son: Síndrome de Cushing, operaciones del páncreas, las inducidas por ingestión de esteroides, las secundarias a enfermedades virales y la que acompaña en ocasiones al Síndrome de Down.

En la APS solo se atienden las diabetes tipo 1 y tipo 2, y solamente a adultos y personas mayores, ya que en caso que el paciente diagnosticado sea un niño, este pasa a ser atendido en el hospital pediátrico correspondiente; en el caso que el paciente sea una embarazada, se trata directamente en el hospital que le corresponda.

Una vez que la persona es diabética debe evitar las complicaciones, en especial las agudas, como la cetoacidosis diabética. Es importante la educación que se le brinda al paciente y su conducta.

Para evitar complicaciones macro vasculares, se requiere un control estricto del azúcar y de los factores de riesgo asociados, en especial la hipertensión, la obesidad, el consumo de grasas animales, el hábito de fumar, el sedentarismo y las infecciones.

En la diabetes tipo 1 ha de administrarse insulina diariamente con la dosis indicada; y ante una infección adoptar las dosis pero no suspender la insulina, intensificar la frecuencia de determinación de glucosa en sangre y en orina, y acudir rápidamente al médico.

En la diabetes tipo 2, la complicación aguda más importante es el estado hiperosmolar. Es totalmente normal que la glucosa sea menos de 5,6, pero si llega a 30 ó 40mmol/l la persona llega a un estado en que orina mucho, toma mucha agua, se puede deshidratar y es de mal pronóstico. Requiere atención médica urgente con hidratación e insulina en el centro de salud correspondiente. [8]

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

Hasta el momento esta enfermedad no es curable, por lo que cuando se diagnostica, será compañía de toda la vida para quien la padece, aunque esto no quiere decir que se está permanentemente enfermo, aunque esta enfermedad sea crónica, puede ser perfectamente controlada, manteniendo niveles de glucemia normales, como los de las personas que no padecen esta enfermedad; de esta forma, las posibilidades de tener complicaciones serán mucho menores.

1.3 Estado del Arte

Actualmente a nivel mundial existen soluciones informáticas, organizaciones, centros, entre otros en el campo de la medicina, específicamente en la APS, un ejemplo de esto es la Federación Internacional de Diabetes (FID) que es el representante mundial de los más de 285 millones de personas con diabetes en todo el mundo. Representa a más de 200 asociaciones de diabetes en más de 160 países entre ellos Cuba. La misión de la FID es la promoción de la prevención, la atención y la cura de la diabetes en todo el mundo. La Federación Internacional de Diabetes es una ONG en relaciones oficiales con la Organización Mundial de la Salud y está asociada al Departamento de Información Pública de Naciones Unidas. [28]

La Federación Internacional sobre Diabetes (IDF) señala graves riesgos para la salud y ha publicado hoy 5 de mayo del 2011 una declaración consensuada sobre los trastornos de la tolerancia a la glucosa (IGT) y de la glucemia basal alterada (IFG) como principales factores de riesgo de futuras enfermedades de diabetes[29]. Subvenciona con 2 millones de dólares 9 proyectos de investigación mundial en diabetes, los proyectos elegidos tratan importantes problemas en la prevención primaria y secundaria de la diabetes en distantes partes del mundo, los proyectos se centran en temas tales como la intervención en los estilos de vida para la prevención de la diabetes tipo 2, la diabetes gestacional, entre otros.

En México se ha desarrollado un **Sistema de Información de Medicina Familiar (SIMF)**, en el cual el médico tiene la posibilidad de llevar un control detallado de toda la información de sus pacientes, mediante un subsistema de Historia Clínica Electrónica (HCE). El sistema también cuenta con una Agenda Electrónica, la cual se muestra al médico en la página de inicio, que contiene un calendario y una lista de todas las citas programadas para el día en curso o demás días del año.

Para la gestión de toda la información generada en la consulta, el SIMF cuenta con el módulo de Atención Integral, el cual permite llevar el control de los registros de la historia clínica de los pacientes, realizar la consulta, generar notas médicas, recetas y certificados médicos, lo que permite llevar el control de la evolución del paciente y la prevención oportuna de enfermedades. Contenido en este módulo se

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

encuentra el submódulo de Hoja de Control de Diabetes Mellitus, el cual brinda la posibilidad de registrar y llevar el control de la salud, así como, de la evolución de los pacientes con Diabetes Mellitus, registrados en el área de salud.

El SIMF como desventaja presenta que no permite modificar la información una vez entrada al sistema, no está integrado con ningún otro sistema para la Atención Primaria de Salud, está desarrollado con base en las necesidades tan particulares y complejas de una institución única en el mundo, como es el instituto mexicano del seguro social y tuvo insatisfacción de los pacientes por el tiempo de espera para recibir atención, este era muy largo.

Otra aplicación de gran importancia para el sistema primario de salud es el **GaleNUx**, el cual brinda la posibilidad al médico de contar con un consultorio médico on-line, donde puede llevar su agenda, registros o historias clínicas, además de ocuparse de tareas administrativas como facturación y estadísticas.

Este sistema fue desarrollado en Colombia en el año 2009, utilizando las tecnologías de PHP y HTML, utilizando MySql como gestor de bases de datos, y apache como servidor de alojamiento. Está publicado bajo la versión GPL V3 por lo que puede ser usado sin costo alguno, además de que su código puede ser estudiado y modificado por cualquier persona para mejorar sus prestaciones.

El sistema GaleNUx tiene como desventaja que no presenta un sistema integral para la atención al paciente diabético, el software está inspirado en la arquitectura cliente-servidor mediando Internet ya que se necesita acceso redundante con un mínimo de 128K por cada canal, esto trae problemas en Cuba ya que la conexión de este país a internet es limitada y lenta por el bloqueo establecido por los Estados Unidos.

Ninguno de estos dos sistemas se puede desplegar en Cuba, ya que su infraestructura no está de acuerdo con el Sistema Nacional de Salud, haciendo muy difícil la integración de estos sistemas a otros sistemas informáticos desarrollados en Cuba, además de las desventajas expuestas anteriormente.

Bio Center es una clínica para el tratamiento integral del paciente diabético que cuenta con un laboratorio clínico automatizado en Guatemala; es decir, los resultados de las pruebas son cuantificables y producidos por equipos computarizados este el primer laboratorio clínico en el país cuyos resultados son emitidos por equipos de última tecnología según estándares preestablecidos a nivel mundial, por eso no hay aquello de “el técnico se equivocó”. “Los controles de calidad son objetivos y se dan menos falsos diagnósticos”. [30]

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

El Instituto Mexicano del Seguro Social (IMSS), ha mejorado la salud y calidad de vida de los pacientes con diabetes mellitus, hipertensión arterial y obesidad, a través del Módulo de Atención Integral al Paciente Diabético (MADI). A dichos pacientes se les han retardado las complicaciones de la enfermedad, al disminuir los índices de glucosa, presión arterial y peso corporal. Este tipo de atención se implantó en marzo del 2004. Los niños y adolescentes ya están incluidos en el MADI porque estas enfermedades afectan cada vez con mayor frecuencia a esta población. Los pacientes son enviados a MADI por el médico familiar, o pueden acudir por iniciativa propia. La atención que se les brinda es personalizada y consiste en realizar diferentes actividades educativas tanto para el enfermo como para sus familiares. [31]

En la actual década, surge la Declaración de Saint Vincent en Europa y la Declaración de las Américas, las cuales enfatizan el papel de la educación en la disminución del problema de salud Diabetes. En 1994, los principales grupos de trabajo de todo el orbe se reunieron en Ginebra, Suiza bajo el auspicio de la Organización Mundial de la Salud (OMS), para abordar los problemas de la educación al paciente para el año 2000 bajo el lema “UNA CADENA ES TAN FUERTE COMO SU ESLABON MAS DEBIL”. [36]

La informatización de la APS y la Medicina General Integral (MGI) aparecen a partir de 1985 en Cuba, como apoyo al médico en el diagnóstico y la terapéutica clínica, al diagnóstico y al programa de salud, en la comunidad. [9]

A partir del año 1985, surgen sistemas computarizados para registrar información en la APS en Cuba, que podían ser manejados por los mismos médicos, sin tener un vasto conocimiento de computación. Desde ese entonces en adelante, la informática en la APS ha pasado por varios altibajos para llegar al nivel que representa hoy día.

Uno de los sistemas que logró marcar un hito en la evolución de la informática en la APS en Cuba, fue el **Sistema Informático para la Gestión Médica en la APS** (APUS). Este sistema tiene como función principal automatizar toda la información que se genere en un centro de salud de la APS, ofreciendo al equipo de salud, la información necesaria, oportuna y útil para la toma de decisiones en tiempo real, que ayuden a elevar el nivel de vida de la población bajo su atención.

APUS fue desarrollado en el lenguaje Borland Delphi, conectado a un sistema de bases de datos SQL Server. También cuenta con una versión en ambiente web, utilizando para el desarrollo de esto el lenguaje ASP (Active Server Pages), y basada en la tecnología Cliente-Servidor.

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

Es sistema APUS presenta como desventaja que es monoplataforma, está implementado con tecnología propietaria y su uso en los consultorios fue poco factible.

La doctora Tania Ramírez González publicó el 8 de mayo del 2007 un artículo desarrollado en Cuba con el título *Pesquisaje de la Enfermedad Renal Crónica en diabéticos e hipertensos en un área de salud* el cual se realiza con el objetivo general de determinar el comportamiento de la Enfermedad Renal Crónica en una población de diabéticos e hipertensos en un área de salud y el objetivo específico de identificar la presencia de enfermedad renal crónica en diabéticos e hipertensos y su relación con la edad y el sexo, determinar en los pacientes el estadio de enfermedad renal crónica, precisar la posible existencia de asociación entre enfermedad renal crónica en pacientes diabéticos e hipertensos con factores de progresión y determinar el conocimiento o no del riesgo de presentar una enfermedad renal crónica.[32]

En la Facultad de Ciencias Médicas de Granma se llevó a cabo una investigación en el Hospital Provincial Clínico quirúrgico Docente "Celia Sánchez Manduley" la cual lleva por título *Pesquisaje Oftalmológico*, publicado en la Revista Cubana de Oftalmología, enero-junio, 1995, en la misma se realiza un estudio a 418 pacientes atendidos en el consultorio no. 14 del Policlínico II de Manzanillo, en el período comprendido desde agosto de 1991 hasta agosto de 1992, con el objetivo de detectar afecciones oftalmológicas, por medio de una encuesta oftalmológica confeccionada para estos fines.[33]

En 1980, el grupo de educación cubano diseñó un Programa de Educación en Diabetes que se incorporó al resto de los programas internacionales surgidos en la década y que en 1989 recibió el PREMIO AL MEJOR TRABAJO DE EDUCACION EN DIABETES EN LATINOAMERICA. Se publicaron los folletos educativos "Consejos dietéticos para el diabético" y "El diabético insulino dependiente". También se inició la publicación de "Control. La Revista del diabético". La continua evaluación de este programa facilitó que se fuera perfeccionando, hasta el diseño de una Estrategia Cubana de Educación en Diabetes. [34]

En Holguín, desde el 11 de marzo de 1993, se creó el primer el Centro para el Desarrollo de la Atención integral al diabético del país, cuyo objetivo principal ha sido preparar al paciente a convivir con este padecimiento, pues el enfermo se encuentra expuesto a un sinnúmero de complicaciones que pueden agravar su calidad de vida si no adopta medidas de imprescindible cumplimiento para mejorar su salud. [35]

Los sistemas informatizados para la salud que existen actualmente en el país, no satisfacen completamente las necesidades de este sector, ya que han sido desarrollados para áreas o centros muy

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

específicos, lo cual imposibilita usar esa tecnología en cualquier centro de salud y a cualquier nivel. También es esencial la integración de estos sistemas con otros existentes o con futuros desarrollos, lo cual actualmente no es posible.

A nivel nacional aunque es bastante avanzado en cuanto a infraestructura con respecto a los niveles de años anteriores, todavía carece de un sistema unificado, estable, y que goce de la aceptación por parte de los profesionales de la salud. El APUS creado en Cuba trae consigo algunas desventajas como son que es monoplataforma, ya que solo se ejecuta en sistemas operativos Windows, además que está orientado al trabajo en los policlínicos y no en los consultorios. Debido a problemas técnicos, de infraestructura y poca aceptación por parte del personal de salud, el sistema se dejó de utilizar hace unos años.

A nivel internacional los sistemas mencionados anteriormente, a pesar de cubrir todas las necesidades de la APS, no cuentan con un sistema de remisión, algo que restaría grandemente molestias y trámites innecesarios a los pacientes. A pesar de las facilidades y opciones que brindan estos sistemas, no es factible utilizarlos en Cuba, ya que el sistema nacional de salud de México por ejemplo, es diferente estructuralmente al cubano, además de que la organización del sistema de salud de otros países, difieren en varios aspectos al cubano, por lo que adquirir y desplegar alguno de esos sistemas en los centros de salud de Cuba, sería mucho más costoso que desarrollarlos de acuerdo a nuestras características.

Todas estas tecnologías diseñadas además de sus ventajas han traído para los médicos varios retos. El EBS consume mucho tiempo en el aprendizaje de las tecnologías en detrimento del tiempo que le dedica al paciente. A los desarrolladores también les ocasiona conflictos con las barreras de interoperabilidad. Con tal de contrarrestar estos problemas las soluciones informáticas actuales están encaminadas hacia la confección de interfaces de usuarios amigables, más usables e intuitivas que le faciliten al médico dedicar menos tiempo a la tecnología y más al paciente, así como la utilización de estándares que permitan la comunicación entre los diferentes sistemas que se utilizan en la APS.

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

1.4 Situación Problemática

La informatización de la atención primaria de salud, y en particular la ofrecida por el médico especialista en medicina general integral (MGI), en muchas ocasiones se percibe como un problema simple que se resuelve solamente creando redes de computadoras personales (CP) y otras tecnologías de información y comunicación (TIC) entre los consultorios y policlínicos. [18]

La atención integral al paciente diabético en la APS, no está exenta al hablar de informatización de la APS. Actualmente toda la información que se genera, manipula y almacena en la consulta de diabéticos se hace manualmente, lo que provoca el riesgo de pérdidas o deterioro de información bastante valiosa para el médico, como puede ser la historia clínica del paciente (HC); además de que existen procesos como el de remisión a otras especialidades o interconsultas que demoran varios días o hasta un mes, ya que el mismo paciente es quien tiene que gestionar la cita en la consulta remitida.

Los centros de APS en todo el mundo muestran una serie de dificultades, entre ellos se encuentran las concernientes a la gestión de la información generada durante las pesquisas sobre DM y otras actividades de apoyo a la misma. Estos problemas perturban en gran medida la capacidad de respuesta de la APS a la población, teniendo como consecuencia que se vea afectada la calidad asistencial ofrecida y con ello la calidad de vida de los pacientes.

El conflicto del asunto se evidencia más a la hora de realizar el seguimiento de la DM y de actualizar la historia clínica individual, demorando la obtención y recuperación de la información requerida. Esto se debe fundamentalmente a que en los lugares donde se encuentran archivadas estas informaciones no tienen un orden, ni lugar seguro donde no se dañe, por lo que la información del paciente no es convincente.

El seguimiento a los problemas de salud de los pacientes es una operación importante dentro de la APS que se obstaculiza por el hecho anteriormente explicado, por lo que la pesquisa para la prevención de la DM se basa solo en lo que refiere el paciente, no teniendo en cuenta los estudios realizados con anterioridad (si es que existen y no están estropeados) provocando que los diagnósticos no sean lo más acertados posibles y el paciente tenga que ser sujeto a diversos estudios para llegar a una comprensión definitiva del problema que lo afecta.

Con la implementación e implantación de un sistema informatizado en toda la APS, se garantiza un mejor y eficiente manejo de los datos del paciente generados durante las consultas, además de una rápida

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

gestión de las remisiones a las diferentes especialidades, garantizando con esto un mayor bienestar para el paciente, ya que 60% de los pacientes diabéticos son personas de avanzada edad, y algunos pudieran presentar complicaciones las cuales se pueden evitar con la gestión automatizada de las remisiones.

1.5 Tecnologías informáticas

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

1.5.1 Java EE 5

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. [11]

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.

1.6 Estilos Arquitectónicos

Atendiendo a la recomendación IEEE-1471, arquitectura de software se entenderá como “la organización fundamental de un sistema encarnado 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”.

1.6.1 Arquitectura en Capas

La arquitectura basada en capas se enfoca en la distribución de roles y responsabilidades de forma jerárquica proveyendo una forma muy efectiva de separación de responsabilidades. El rol indica el modo y tipo de interacción con otras capas, y la responsabilidad indica la funcionalidad que está siendo desarrollada.[12]

La programación por capas es un estilo de programación en la que el objetivo primordial es la separación de la lógica de negocios de la lógica de diseño, un ejemplo básico de esto es separar la capa de datos de la capa de presentación al usuario. 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 ataca al nivel requerido sin tener que revisar

entre código mezclado. Entre las más usadas se encuentra la arquitectura de 3 capas la cual se compone de: capa de presentación, capa de negocio, capa de datos.

1.6.2 Patrón Modelo-Vista-Controlador (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, vistas, controlador.

1.6.3 Patrones GRASP

Los patrones generales de software para asignación de responsabilidades o por su nombre en inglés General Responsibility Assignment Software Patterns (GRASP) describen los principios fundamentales de diseño de objetos para la asignación de responsabilidades. Se pueden destacar 5 patrones principales que son: [13]

- Experto.
- Creador.
- Alta cohesión.
- Bajo acoplamiento.
- Controlador.

El GRASP de experto en información es el principio básico de asignación de responsabilidades. Indica, por ejemplo, que la responsabilidad de la creación de un objeto o la implementación de un método, debe recaer sobre la clase que conoce toda la información necesaria para crearlo. De este modo se obtendrá un diseño con mayor cohesión y así la información se mantiene encapsulada (disminución del acoplamiento)

El patrón creador ayuda a identificar quién debe ser el responsable de la creación (o instanciación) de nuevos objetos o clases. Una de las consecuencias de usar este patrón es la visibilidad entre la clase creada y la clase creador. Una ventaja es el bajo acoplamiento, lo cual supone facilidad de mantenimiento y reutilización. La creación de instancias es una de las actividades más comunes en un sistema orientado a objetos. En consecuencia es útil contar con un principio general para la asignación de las responsabilidades de creación. Si se asignan bien, el diseño puede soportar un bajo acoplamiento, mayor claridad, encapsulación y reutilización.

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

El patrón controlador es un patrón que sirve como intermediario entre una determinada interfaz y el algoritmo que la implementa, de tal forma que es la que recibe los datos del usuario y la que los envía a las distintas clases según el método llamado.

Este patrón sugiere que la lógica de negocios debe estar separada de la capa de presentación, esto para aumentar la reutilización de código y a la vez tener un mayor control.

Se recomienda dividir los eventos del sistema en el mayor número de controladores para poder aumentar la cohesión y disminuir el acoplamiento.

Los conceptos de cohesión y acoplamiento están íntimamente relacionados. Un mayor grado de cohesión implica uno menor de acoplamiento. Maximizar el nivel de cohesión intramodular en todo el sistema resulta en una minimización del acoplamiento intermodular.

El patrón alta cohesión dice que la información que almacena una clase debe de ser coherente y debe estar (en la medida de lo posible) relacionada con la clase.

El patrón bajo acoplamiento es la idea de tener las clases lo menos ligadas entre sí que se pueda. De tal forma que en caso de producirse una modificación en alguna de ellas, se tenga la mínima repercusión posible en el resto de clases, potenciando la reutilización, y disminuyendo la dependencia entre las clases.

1. **Acoplamiento de Contenido:** Cuando un módulo referencia directamente el contenido de otro módulo. (En lenguajes de alto nivel es muy raro)

2. **Acoplamiento Común:** Cuando dos módulos acceden (y afectan) a un mismo valor global.

3. **Acoplamiento de Control:** Cuando un módulo le envía a otro un elemento de control que determina la lógica de ejecución del mismo.

1.7 Lenguajes de programación

1.7.1 Java3.0

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.

1.7.2 Extensible Hypertext Markup Language (XHTML) 1.1

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. [14]

1.7.3 Lenguaje de Marcas Extensible (XML) 1.0

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. Actualmente en el proyecto se utiliza para la configuración a la base de datos.

1.7.4 Lenguaje de Expresión (EL)

Es el lenguaje de expresión que provee JSF el cual es utilizado en las páginas de las aplicaciones web con el fin de acceder a la entidad java. [15]

1.7.5 Hibernate Query Language (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. [16]

1.7.6 Unified Modeling Language (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. [17]

1.8 Metodología de desarrollo

La metodología de desarrollo cobra gran importancia en proyectos empresariales, pues al no utilizarla adecuadamente se puede provocar en el fracaso del equipo de desarrollo y 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,

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

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 de las Ciencias Informáticas como parte del proceso de mejora que se está llevando a cabo actualmente para alcanzar el nivel dos de CMMI (Capability Maturity Model Integration). 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.

1.9 Gestor de Base de Datos

Los Sistemas de gestión de bases 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.

1.9.1 PostgreSQL8.4

PostgreSQL es un gestor 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.

2 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.

2.1 JBoss 4.2.2

JBoss es un servidor de aplicaciones Java EE de código abierto implementado en Java puro. 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.

2.2 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 extienden o utilizan 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.

2.2.1 Enterprise Java Beans (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. [18]

2.2.2 Java Server Faces (JSF) 1.2

JSF 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.

2.2.3 Java Persistence API (JPA)

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).

2.2.4 Hibernate 3.3

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.

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

2.2.5 SEAM 2.1.1

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). [19]

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.

2.2.6 Facelets 1.1

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. [20]

2.2.7 RichFaces 3.3.1

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. [21]

3 Herramientas informáticas

3.1 Eclipse Ganymede 3.5.2

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. [22]

3.2 Visual Paradigm 6.4

Visual Paradigm para UML es una herramienta CASE profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. El

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

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. [23]

En este capítulo se trataron temas referentes al proceso de informatización del Sistema Nacional de Salud cubano y a nivel internacional, así como conceptos importantes del sector de la salud que facilitan aún más la comprensión de toda la investigación realizada. Se expusieron y estudiaron los sistemas existentes relacionados con el tema de la investigación, además de las tecnologías, metodologías, herramientas y lenguajes que se usarán para el desarrollo del sistema propuesto.

Capítulo 2. Características del Sistema

En el presente capítulo se tratará todo lo referente al modelado del negocio del sistema de Componente Web Atención al Paciente Diabético. Se detallan los procesos involucrados haciendo un profundo análisis de las actividades que están presentes en éstos y serán objeto de automatización. También se especifican los requerimientos funcionales y no funcionales que debe cumplir el sistema para su correcto funcionamiento.

2. Modelo del Negocio

El objetivo principal para la modelación del negocio es detallar los procesos, existentes en el negocio, con la intención de comprender y concretar qué procesos del negocio resistirá el sistema. Se debe identificar los objetos implicados para establecer las características que se solicitan de cada proceso. Con el propósito de alcanzar estos planes, el proceso de modelado permite adquirir una visión de la entidad, con la que sea factible precisar los procesos, roles y responsabilidades.

Para el correcto desarrollo de un sistema informático, es necesario un perfecto conocimiento del negocio de la entidad y de los procesos que se quieren automatizar. Un proceso como una serie de actividades interconectadas en busca de un propósito. En un contexto de negocios, el propósito de los procesos será proveer a sus clientes internos con sus requerimientos de una manera oportuna. [24]

El modelado del negocio es muy útil para entender la lógica del funcionamiento de la organización, su estructura, y la dinámica de sus procesos, lo que permite identificar mejoras potenciales a problemas existentes, así como lograr que los usuarios finales y los desarrolladores tengan una idea común de la organización, y poder definir los requerimientos del sistema a partir del modelo de negocio que se obtenga.

2.2 Diagramas de procesos del negocio.

El proceso de negocio de este sistema web cuenta con dos procesos el primero: realizar consulta al paciente diabético como se muestra en la fig. 1, el mismo consiste en recoger los datos del paciente cuando llega a la consulta, realizarle el examen físico y en caso necesario los exámenes complementario, si la prueba de tolerancia da positiva procede la clasificación según el tipo de diabetes que posea y se le emite el tratamiento o dieta. El segundo proceso es el de pesquijaje como se muestra en la fig. 2 en el cual se realizan las pesquisas activas que son los médicos en la población haciendo terreno y la pesquisa pasiva es cuando el paciente se detecta los síntomas y acude a la consulta.

CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

2.2.1 Proceso Realizar Consulta al Paciente Diabético.

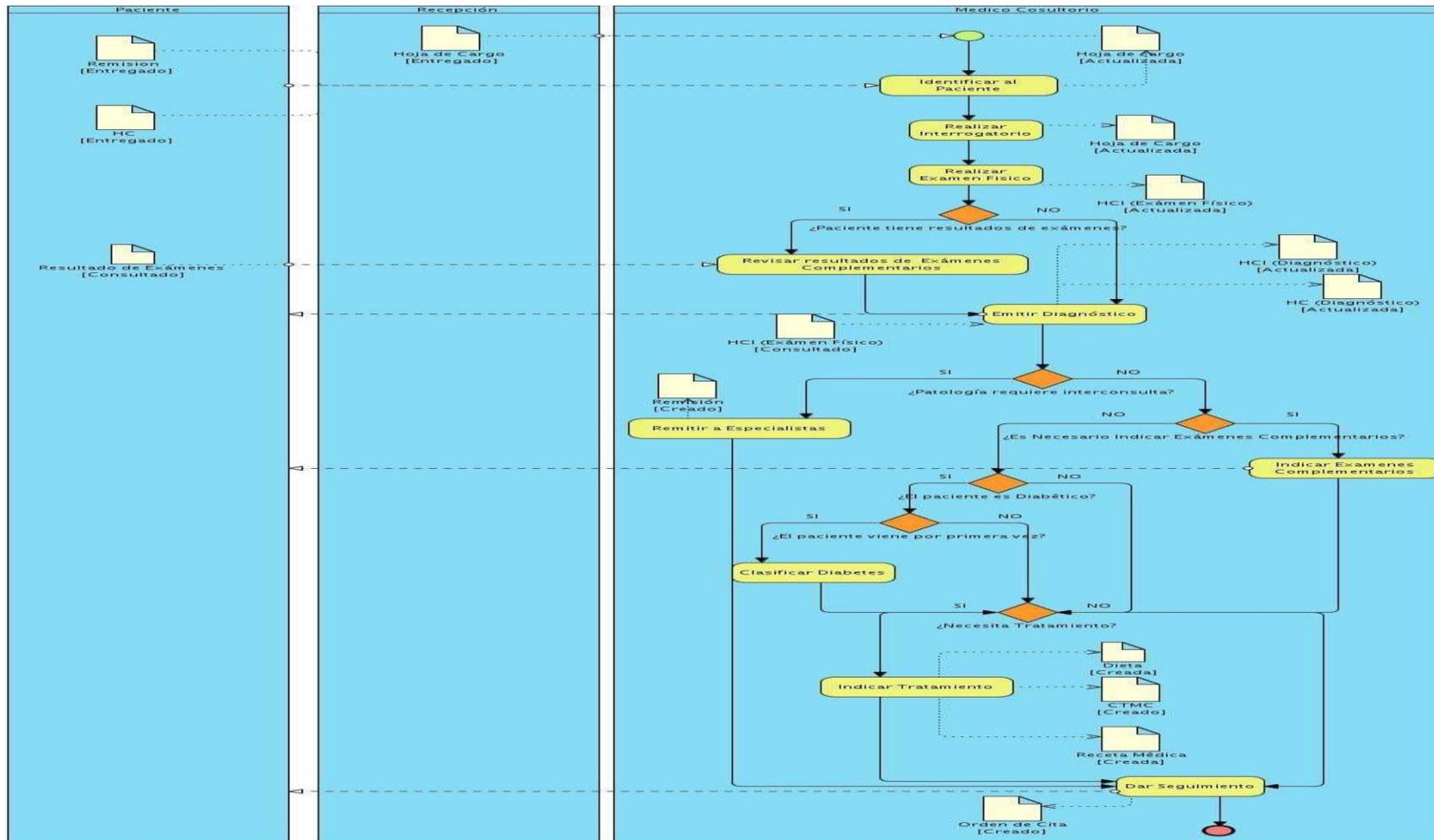


Fig. 1 Realizar consulta al paciente diabético

CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

2.2.2 Proceso Pesquisaje

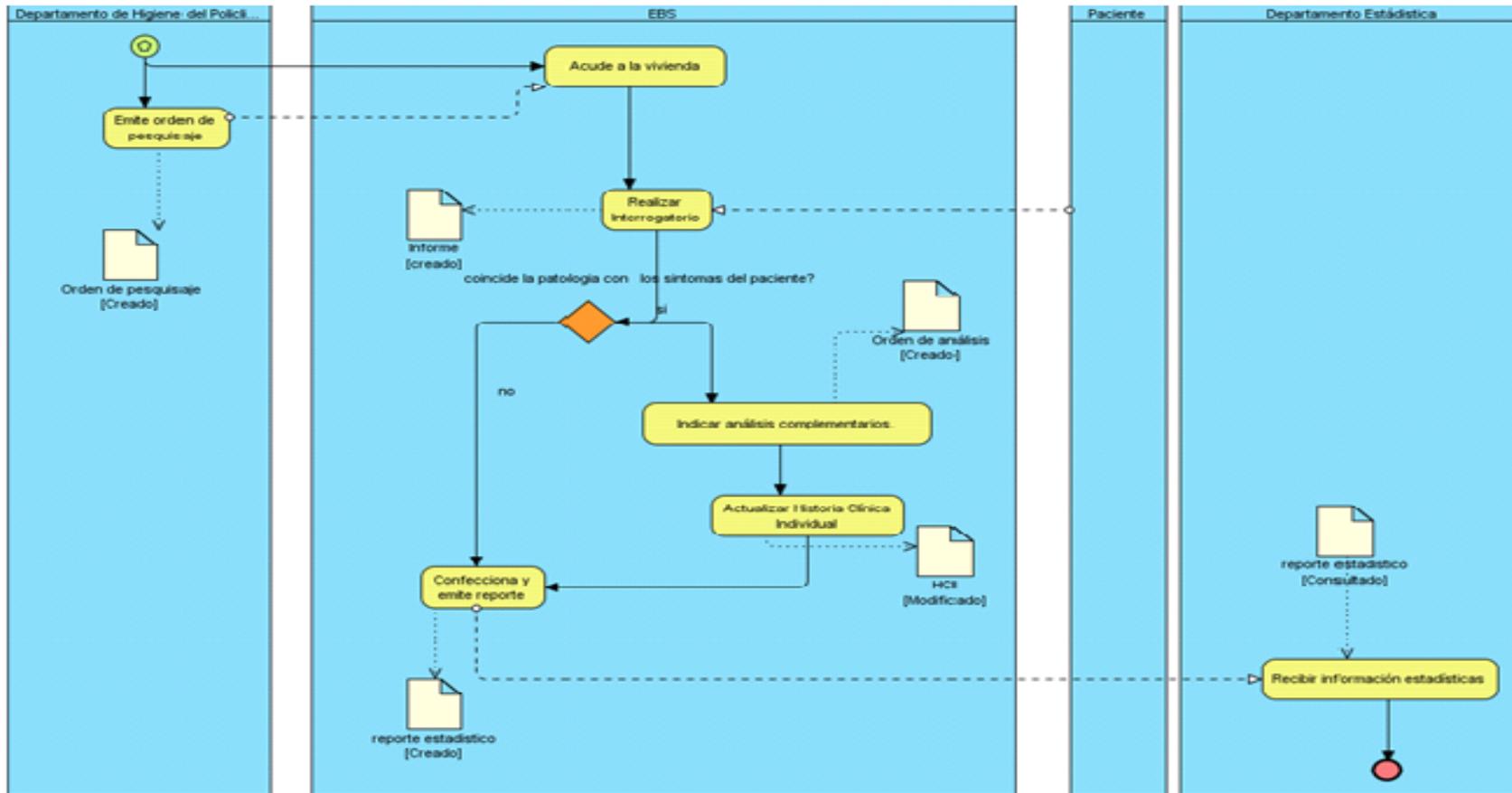


Fig. 2Pesquisaje

CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

2.3 Descripción textual de los procesos de negocio.

El proceso fundamental y más importante del negocio de Atención al Paciente Diabético que se muestra en la tabla1, es el proceso de Realizar Consulta al Paciente Diabético, ya que en él tienen lugar todas las actividades de mayor relevancia para el paciente y el segundo proceso se puede ver en la tabla2, este es el de Pesquisaje, explicado anteriormente.

2.3.1 Proceso de negocio: Realizar Consulta al Paciente Diabético.

Nombre:	P4 Realizar Consulta al Paciente Diabético
Objetivos:	Consultar y tratar al paciente.
Evento(s) que lo generan:	Presencia del paciente con diabetes diagnosticada por el Médico de Familia (Remisión Médica).
Precondiciones:	Debe Haber sido remitido por el Médico de la Familia.
Poscondiciones:	Diagnosticar al paciente
Reglas de Negocio:	
Responsables:	Médico diplomado en Diabetes
Clientes internos:	Paciente
Clientes externos:	CESIM
Entradas:	Hoja de cargo, Remisión, HCl, CTMC, Dieta, Receta Médica
Salidas:	Orden de Cita, Modelo de Remisión
Actividades:	Act.1 Identificar al paciente. Act.2 Realizar Interrogatorio. Act.3 Realizar Examen Físico.

CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

	<p>Act.4 Revisar resultados de exámenes complementarios</p> <p>Act.5 Emitir Diagnóstico</p> <p>Act.6 Remitir a Especialistas</p> <p>Act.7 Indicar exámenes complementarios.</p> <p>Act.8 Clasificar Diabetes</p> <p>Act.9 Indicar Tratamiento.</p> <p>Act.10 Dar Seguimiento.</p>
--	---

Tabla 1 Realizar consulta al paciente diabético

2.3.2 Proceso de negocio: **Pesquisaje.**

Nombre:	P1_Pesquisaje.
Objetivos:	<p>El proceso tiene como objetivo captar los factores de riesgo que propician la enfermedad por la cual se está realizando el pesquisaje.</p> <p>El propósito que persigue es detectar las enfermedades a tiempo y evitar su propagación o en otros casos el empeoramiento de la salud del paciente.</p>
Evento(s) que lo generan:	Una orden emitida desde el Departamento de Higiene del policlínico
Precondiciones:	Que exista un brote de alguna enfermedad que se propague y el Departamento de Higiene del policlínico emita la orden de pesquisaje.
Poscondiciones:	<p>Emitir reportes diarios con estadísticas.</p> <p>Evaluar la calidad de vida del paciente.</p>
Reglas de Negocio:	- Regla 4. Ver referencia 1.
Responsables:	Equipo Básico de Salud (EBS)

CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

Clientes internos:	EBS
Clientes externos:	CESIM
Entradas:	-Orden de pesquijaje
Salidas:	-Orden de análisis -Reporte final

Tabla 2 Pesquijaje

2.4 Propuesta del Sistema.

Establecer un punto de partida es fundamental para realizar cualquier acción y planificar qué se quiere hacer y cómo llevarlo a cabo desde un análisis claro de las necesidades del cliente ya que es vital para el buen desarrollo de la aplicación, cuando un producto es terminado y se entrega sin haber dado este paso, los costos se lanzan, lo que lleva a problemas con los clientes. Manteniendo una comunicación con los beneficiarios del trabajo, no solamente se logra acelerar el proceso de desarrollo y minimizar los costos de producción, sino, que también se evita cambiar toda o gran parte de la estrategia elegida impidiendo así que se retorne al mismo punto de inicio; además se logra hacer que el equipo funcione mejor y aumente su potencial de desarrollo. Decidir qué se va a construir es el punto crítico en el desarrollo de un software, no solamente lo convierte en el contrato con el cliente; sino que define la meta del esfuerzo.

2.4.1 Requerimientos.

La representación completa del comportamiento del sistema que se va a desplegar se ve en la Especificación de Requisitos Software (ERS). Este incluye todas las interacciones que tendrán los usuarios con el software en la descripción de un conjunto de casos de uso. Los últimos también son conocidos como requisitos funcionales. La ERS además de los casos de uso, también contiene requisitos no funcionales, y estos son los que imponen restricciones en el diseño o la implementación.

2.4.1.1 Requisitos Funcionales

Los requisitos funcionales definen el procedimiento interno del software ya que son capacidades o condiciones que el sistema debe cumplir. A partir del estudio realizado en el negocio, se obtuvieron los siguientes requerimientos funcionales que ha de cumplir el sistema, con vista al agrado y al cumplimiento de las necesidades del cliente:

CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

RF 1	Gestionar configuración de cuestionarios para Pesquisajes
RF 1.1	Agregar cuestionarios para Pesquisaje
RF 1.2	Buscar cuestionario para pesquisaje.
RF 1.3	Eliminar cuestionario de pesquisaje
RF 1.4	Modificar cuestionario de pesquisaje
RF 1.5	Listar cuestionarios de pesquisaje
RF 2	Realizar Pesquisa
RF 2.1	Registrar datos del paciente
RF 2.2	Realizar encuesta
RF 2.3	Realizar examen físico
RF 2.4	Realizar exámenes complementarios

2.4.1.2 Requisitos no Funcionales

Los requisitos no funcionales definen propiedades y restricciones del sistema. Habitualmente son afines a requerimientos funcionales, es decir se puede determinar cómo ha de comportarse, una vez que se conozca lo que el sistema debe hacer, qué condiciones debe tener o cuán rápido debe ser.

Requerimientos de Usabilidad

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

- Usuarios normales: 20 días
- Usuarios avanzados: 30 días

Confiabilidad

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 base de datos.

Se garantizarán 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.

CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

Se mantendrá seguridad y 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 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 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.

Se permitirá realizar copias de seguridad de la base de datos hacia otro dispositivo de almacenamiento externo.

Se permitirá realizar réplica de la base de datos entre las diferentes entidades y estas a su vez con el Centro de Datos, esta réplica se podrá hacer de forma manual y automatizada a través de la red.

Se permitirá el chequeo de las operaciones y acceso de los usuarios al sistema, para esto debe existir un registro de trazas que almacene todas las transacciones realizadas en el sistema, indicando para cada caso como mínimo: usuario que realizó la transacción, tipo de operación que se realizó, fecha y hora en que se realizó la operación e información contenida en el registro modificado.

CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

Eficiencia

Se deberá usar siempre que sea posible el patrón Singleton, destruir referencias que ya no estén siendo usadas, optimizar el trabajo con cadenas, entre otras buenas prácticas que ayudan a mejorar el rendimiento.

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

Se potenciará como regla guardar en la memoria caché datos y recursos de alta demanda.

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

Soporte

Se permitirá administración remota, monitoreo del funcionamiento del sistema en los centros hospitalarios y detección de fallas de comunicación.

El subsistema de escritorio brindará la posibilidad de actualizarse y realizar el soporte mediante la tecnología Java Web Start.

Las notificaciones de las deficiencias detectadas en la aplicación desplegada deberán realizarse por escrito.

Una vez notificada por la entidad médica, la deficiencia detectada en la aplicación desplegada, el equipo de desarrollo deberá solucionarla en un período de 7 días.

La capacitación y entrenamiento del profesional de salud para el uso del sistema, se realizará en un período de 6 meses.

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.

CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

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.

La nomenclatura de los bases de datos, tablas, campos de las tablas, esquemas, procedimientos almacenados o funciones, vistas, triggers y tipos de datos se elaborarán siguiendo los estándares definidos en el proyecto.

La codificación se realizará siguiendo el estándar definido en el proyecto.

Las interfaces se realizarán siguiendo el estándar definido en el proyecto.

Requisitos para la documentación de usuarios en línea y ayuda del sistema.

Se posibilitará el uso de ayudas y manuales dinámicos en línea sobre el funcionamiento del sistema.

Interfaz

Las ventanas del sistema contendrán los datos claros y bien estructurados para facilitar 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 el idioma inherente al país o región donde se despliegue el sistema.

Se interactuará con el sistema alas RIS para realizar solicitudes y obtener resultados de estudios radiológicos e imagenológicos a través de una interfaz definida en el proyecto.

Para el intercambio electrónico de datos entre aplicaciones se usará el protocolo HL7 (Health Level Seven).

El sistema usará el formato estándar WSDL (Web Services Description Language) para la descripción de los servicios web.

El sistema implementará mecanismos de encriptación de datos para el intercambio de información con sistemas externos.

CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

El sistema utilizará mecanismos de compactación de los datos que se intercambiarán con sistemas externos con el objetivo de minimizar el tráfico en la red y economizar el ancho de banda.

Los equipos autoanalizadores se podrán programar desde el sistema para realizar análisis y obtener resultados automáticamente a través de una interfaz definida en el proyecto.

Requerimientos de hardware

Las estaciones de trabajo debe tener como mínimo 256 Mb de memoria RAM y un microprocesador de 2.0 Hz.

La solución estará conformada, fundamentalmente, por servidores de alta capacidad de procesamiento y redundancia, que permitan garantizar movilidad, residencia de la información y aplicaciones bajo esquemas seguros y confiables.

Los servidores de base de datos deberán tener: 1 DL380 G5, Procesador Intel® Xeon® 5140 Dual - Core 4GB de memoria y 2x72GB de disco y sistema operativo Linux.

Los servidores de aplicaciones deberán tener: 2 DL380 G5, Procesador Intel® Xeon® 5140 Dual - Core 4GB de memoria y 2x72GB de disco y sistema operativo Linux.

Los servidores de Intercambio deberán tener: 1 DL380 G5, Procesador Intel® Xeon® 5140 Dual - Core 2 GB de memoria y 2x72GB de disco y sistema operativo Linux.

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.

En este capítulo se sentaron las bases para el desarrollo de la Disciplina de Diseño, Implementación y Pruebas del sistema. Se identificaron y describieron cada uno de los procesos del negocio y las actividades que son desarrolladas por este. Se obtuvo un aspecto más amplio del sistema a desarrollar, plasmándose los requisitos funcionales y no funcionales que deberán ser cumplidos.

Capítulo 3. Diseño del Sistema

En este capítulo el flujo de trabajo de Diseño es el objetivo fundamental ya que es el proceso de gestión para la creación de un sistema, el cual engloba un conjunto de actividades y su principal tarea es modelar el sistema, encontrando su forma para que soporte todos sus requisitos, tanto funcionales como no funcionales. También está dedicado a describir el diseño de la solución propuesta con el trabajo fundamental de transformar los requisitos funcionales en un diseño de clases, tomándose en cuenta los puntos relacionados con los requisitos no funcionales. Se construye el Modelo de Datos, los diagramas de clases del diseño y de interacción. Además, se definen los principios de diseño y los distintos patrones a utilizar en el diseño del sistema.

3 Modelos de Diseño.

El proceso de definición del conjunto de elementos y características que conforman su estructura como soporte de información es el diseño de modelos. Este es un esquema del Modelo de Implementación y su código fuente, el cual fundamentalmente se emplea para representar y documentar su diseño. En las actividades relacionadas con la implementación es usado como entrada esencial.

El diseño tiene como propósito formar la base para la implementación, además contienen diagramas, clases, paquetes, subsistemas, cápsulas, protocolos, interfaces, relaciones, colaboraciones, atributos, entre otros que se puedan considerar para el sistema en desarrollo, también permite desarrollar la arquitectura, dándole soporte a todos los requisitos, incluyendo los no funcionales, descompone el trabajo a desarrollar en partes más entendibles que pueda ser realizado por diferentes equipos de desarrollo, teniendo en cuenta aspectos como: lenguaje de programación, sistema y tecnologías a emplear.

3.1. Patrones de Diseño.

Los patrones de diseño son soluciones simples y elegantes a problemas específicos y comunes del diseño orientado a objetos. Son soluciones establecidas a partir de la experiencia. Es una técnica para flexibilizar el código haciéndolo satisfacer ciertos criterios, es una estructura de implementación que consigue un propósito determinado, un lenguaje de alto nivel y una condición con más destreza para describir ciertos aspectos de la organización de un programa. A continuación se describen los patrones utilizados durante el diagrama del diseño de la solución propuesta, el cual fue el patrón Modelo-Vista-Controlador.

El modelo es el Sistema Gestor de Base de Datos (SGBD) y la Lógica de negocio, y el controlador es el responsable de recibir los eventos de entrada desde la vista y es él que engloba toda la lógica del

negocio, está representado por las clases controladoras que son las encargadas de responder a las solicitudes del usuario. El uso del framework JSF permite al usuario interactuar con la aplicación implementando de este modo la vista; además permite definir clases controladoras que recojan la información introducida y que dispongan de métodos que responden a las acciones del usuario y manipulen los datos contenidos en el modelo.

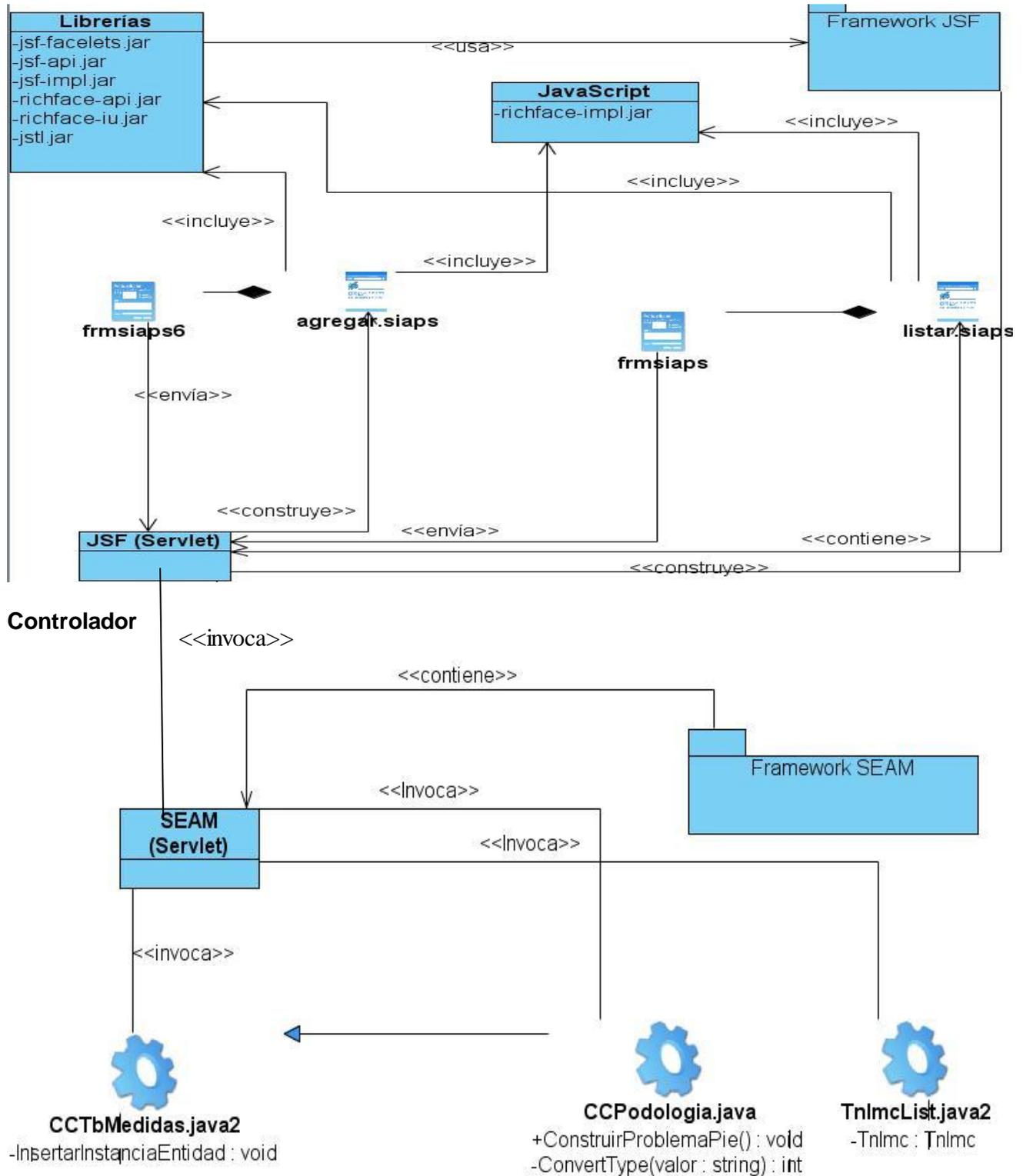
3.2. Diagramas de Diseño.

El diagrama de clases es un ejemplo de diagrama estático que representa la estructura de un sistema manifestando sus clases, atributos y las relaciones entre ellos. Los diagramas de clases se utilizan durante el proceso de análisis y diseño de los sistemas, en el cual se crea el diseño conceptual de la información que se manejará en el sistema, y los componentes que se delegarán del funcionamiento y la relación entre uno y otro. A continuación en la fig. 3 se muestra el diagrama de diseño de Registrar medidas del paciente y en la fig. 4 el diagrama de diseño de Emitir certificado, para más información dirigirse al expediente de proyecto y ver el documento de pautas del diseño del proyecto APS.

Registrar Medidas del Paciente

Vista

CAPÍTULO 3. DISEÑO DEL SISTEMA



Modelo

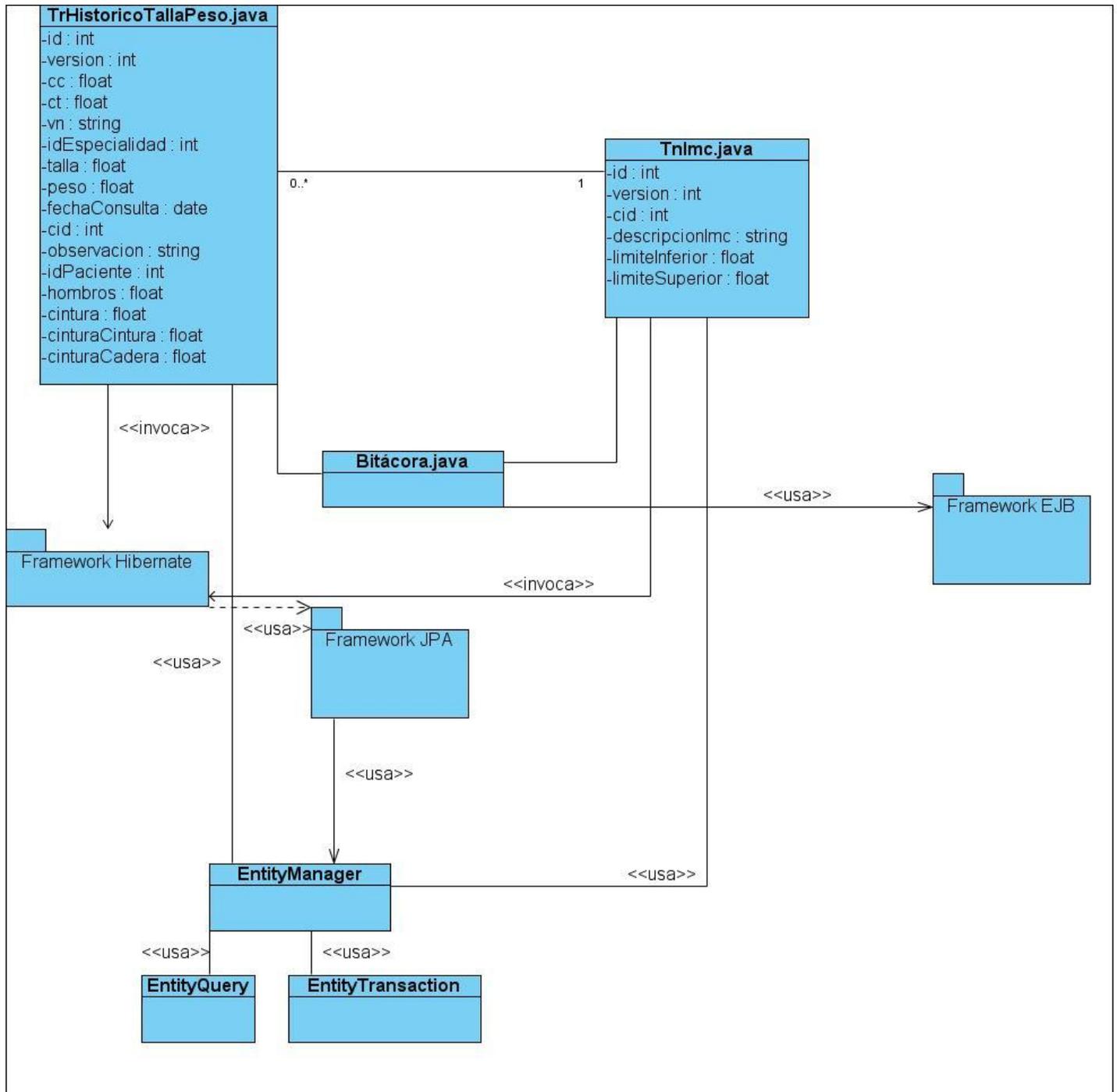
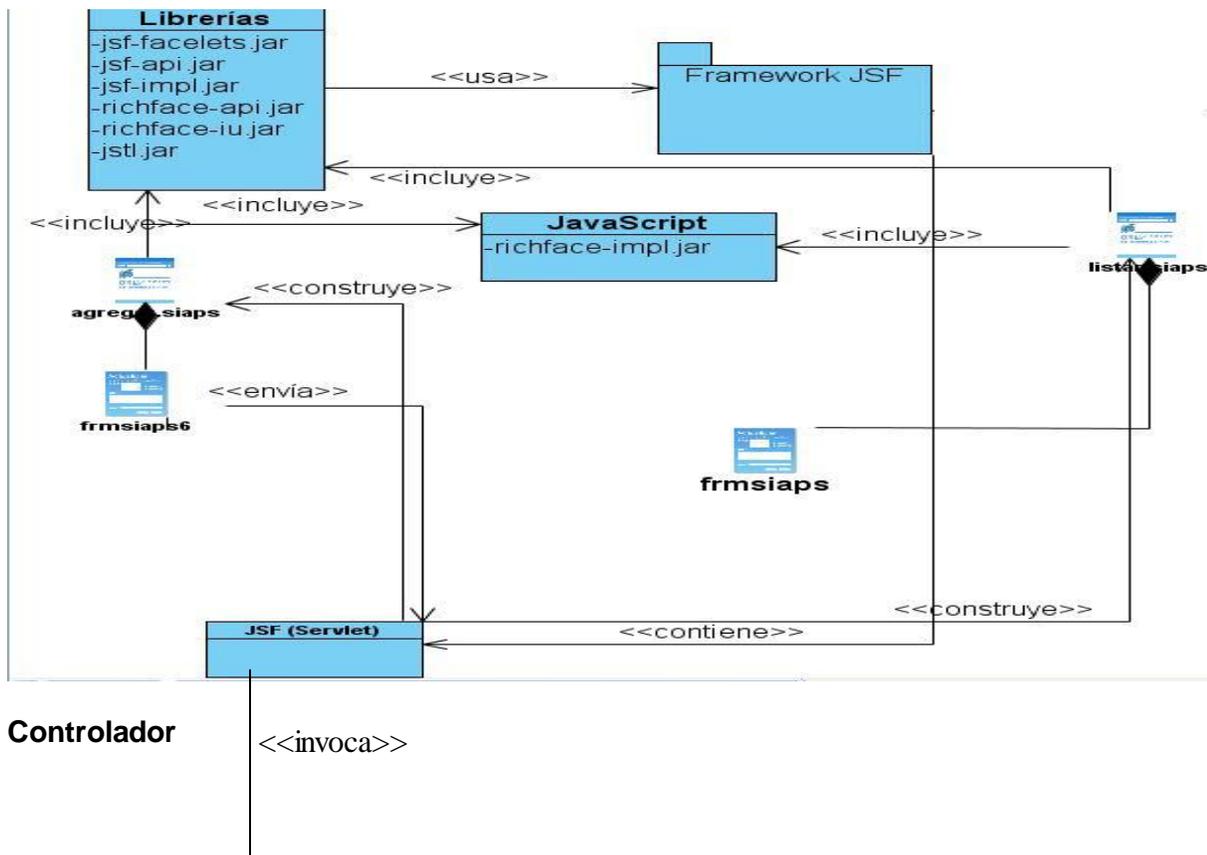


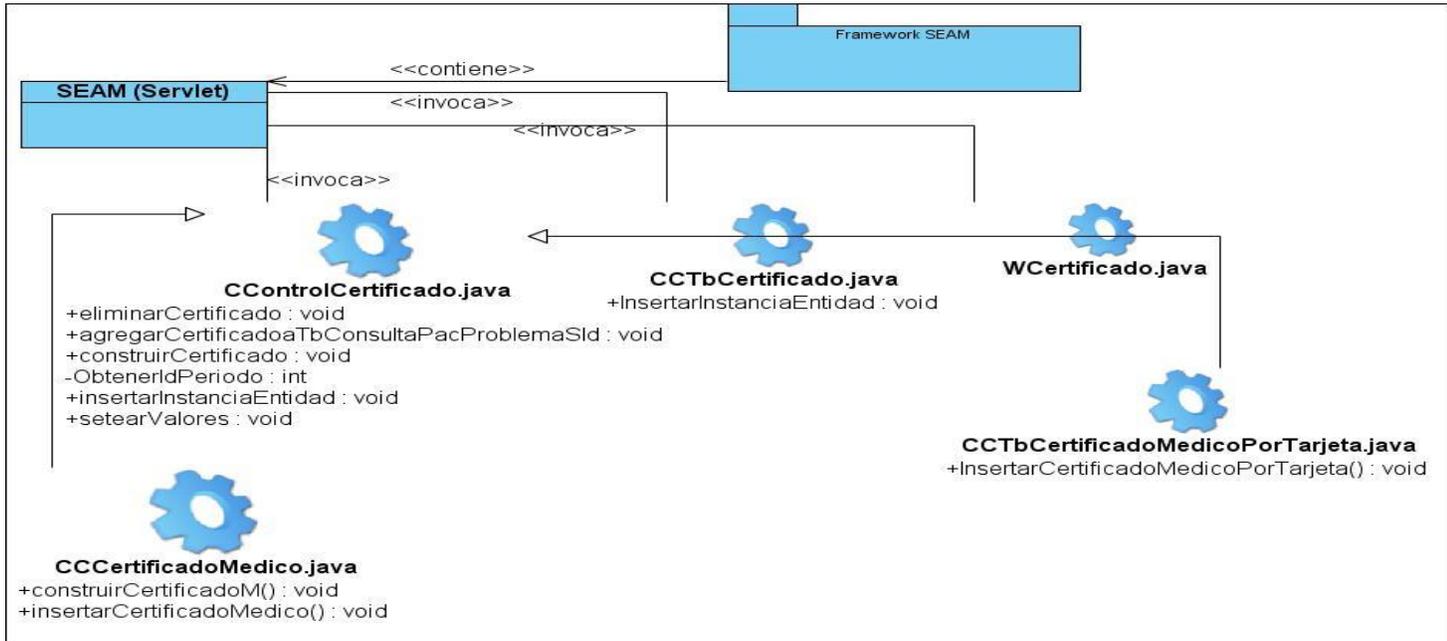
Fig. 3 Registrar Medidas del Paciente

Emitir Certificado

Vista



CAPÍTULO 3. DISEÑO DEL SISTEMA



Modelo

3.2.1 Descripciones Textuales.

Se detallan a continuación algunas de las clases que fueron identificadas para la posterior implementación del sistema (ver tabla 3 y tabla 4), logrando como objetivo una mejor comprensión del trabajo que tendrá el mismo. Estas descripciones de los diagramas le ayudarán a entender los diagramas expuestos anteriormente. Para ver la descripción así como los diagramas restantes, remitirse al expediente de proyecto, el documento APS_SIAPS_0121_MDI_ClinicoQ_v2.0.

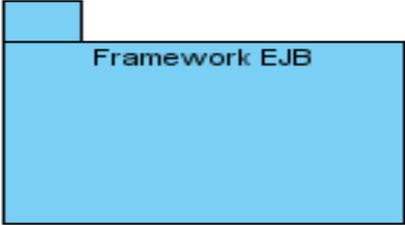
3.2.2 Clases comunes. Descripción.

Capa de Presentación	
Nombre:	Propósito:
<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px; background-color: #ADD8E6; text-align: center; font-weight: bold;">JavaScript</div> <div style="border: 1px solid black; height: 40px; width: 100%;"></div>	<p>Proveer el desarrollo de interfaces de usuario mejoradas y páginas web dinámicas.</p>
Descripción:	
<p>Permite realizar las validaciones del lado del cliente y construir páginas más dinámicas integradas a un navegador web</p>	
Nombre:	Propósito:
<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px; background-color: #ADD8E6; text-align: center; font-weight: bold;">Librerías</div> <div style="border: 1px solid black; padding: 5px; background-color: #ADD8E6;"> -jsf-facelets.jar -jsf-api.jar -jsf-impl.jar -richface-api.jar -richface-impl.jar -richface-iu.jar -jstl.jar </div>	<p>Permite la creación de interfaces.</p>
Descripción:	
<p>Contiene un conjunto de librerías que permiten construir una interfaz de usuario con componentes reutilizables y extensibles. Reducen significativamente la carga de construir y mantener aplicaciones web con componentes de interfaz del lado del servidor.</p>	

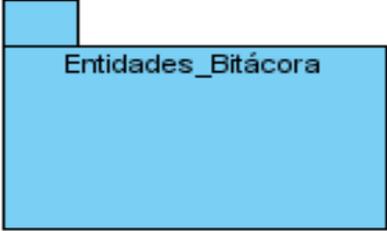
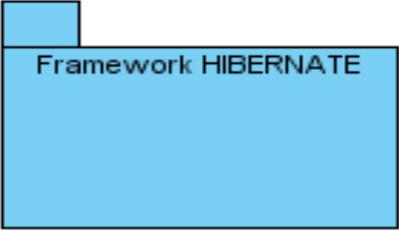
CAPÍTULO 3. DISEÑO DEL SISTEMA

Nombre:	Propósito:
	Interceptar las peticiones a las páginas JSF.
Descripción:	
Es el controlador de JSF que intercepta las peticiones de las páginas clientes, asociándoles a estas páginas, clases java que recogen la información introducida y que disponen de métodos que responden a las acciones del usuario. Además, prepara el contexto JSF antes de enrutar a las páginas correspondientes e interviene en la construcción de la respuesta para generar la vista, luego de ser invocada una petición.	
Nombre:	Propósito:
	Simplificar el desarrollo de las interfaces de usuario en aplicaciones java basadas en el patrón Modelo-Vista-Controlador.
Descripción:	
La tecnología Java Server Faces es un framework de los componentes de la interfaz de usuario y es válido para todas aquellas aplicaciones web basadas en la tecnología JAVA, está basado en el patrón MVC. Forma parte de la especificación JEE 5.	
Capa de Negocio	
Nombre:	Propósito:
	Proveer las funcionalidades para realizar las auditorias del sistema.
Descripción:	
Permite realizar las funciones para almacenar datos como la fecha, hora, usuario, contraseña, entre otros, del usuario que inicia y finaliza la sesión. Además, datos como el modulo y funcionalidad accedidos	

CAPÍTULO 3. DISEÑO DEL SISTEMA

así como las que acciones ejecutadas.	
Nombre:	Propósito:
	Encapsular la lógica de negocio que cumplimenta el propósito de la aplicación.
Observaciones:	
El Framework EJB está incluido en las capas de Negocio y Persistencia.	
Descripción:	
Es una plataforma para construir aplicaciones de negocio portables, escalables, y reutilizables utilizando el lenguaje de programación Java. El objetivo de Enterprise JavaBeans (EJB) 3.0 es simplificar el desarrollo de aplicaciones Java y estandarizar el API de persistencia para la plataforma Java. Forma parte de la especificación JEE 5.	
Capa de Datos	
Nombre:	Propósito:
	Gestionar las entidades proveyendo servicios de persistencia.
Descripción:	
Permite realizar las operaciones CRUD (Crear, Leer, Actualizar y Eliminar) que impliquen entidades.	
Nombre:	Propósito:
	Agregar consultas que pueden aplicarse a las entidades del modelo.
Descripción:	
Permite encontrar objetos persistentes manejando cierto criterio de búsqueda. Permite realizar peticiones a la base de datos y controla cómo se ejecuta dicha petición. Se utiliza para enlazar los parámetros de la petición, limitar el número de resultados devueltos por la petición y para ejecutar dicha petición.	
Nombre:	Propósito:

CAPÍTULO 3. DISEÑO DEL SISTEMA

 <p>A UML package diagram showing a package named 'EntityTransaction' with a small blue square icon in the top-left corner.</p>	Agrupar las operaciones sobre datos persistentes en una unidad transaccional.
Descripción:	
Permite realizar operaciones sobre datos persistentes de manera que agrupados formen una unidad de trabajo transaccional, en el que todo el grupo sincroniza su estado de persistencia en la base de datos o todos fallan en el intento, en caso de fallo, la base de datos quedará con su estado original. Maneja el concepto de todos o ninguno para mantener la integridad de los datos.	
Nombre:	Propósito:
 <p>A UML package diagram showing a package named 'Entidades_Bitácora' with a small blue square icon in the top-left corner.</p>	Agrupar las entidades que contienen la información de las auditorías del sistema.
Descripción:	
Contiene el conjunto de entidades que poseen la información de los usuarios y sus trazas en cuanto a sesión utilizada, módulos accedidos, funcionalidades permitidas, acciones realizadas y atributos modificados. Son utilizadas por la clase Bitácora.java para realizar las auditorías del sistema.	
Nombre:	Propósito:
 <p>A UML package diagram showing a package named 'Framework HIBERNATE' with a small blue square icon in the top-left corner.</p>	Proveer el mapeo objeto/relacional con la base de datos.
Descripción:	
Conjunto de clases agrupadas en componentes que constituyen una herramienta de Mapeo objeto/relacional ó ORM de código abierto (Object Relational Mapping) y un generador de sentencias	

CAPÍTULO 3. DISEÑO DEL SISTEMA

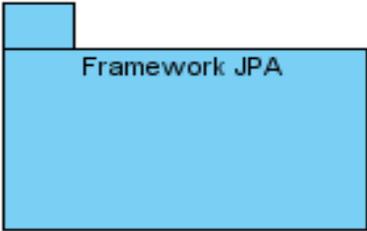
<p>SQL. Permite diseñar objetos persistentes que podrán incluir polimorfismo, relaciones, colecciones, y un gran número de tipos de datos. De una manera muy rápida y optimizada permite generar Bases de Datos en cualquiera de los entornos soportados: Oracle, PostgreSQL, DB2, MySQL, entre otras.</p>	
Nombre:	Propósito:
	<p>Unificar la manera en que funcionan las utilidades que proveen un mapeo objeto-relacional.</p>
Descripción:	
<p>Conjunto de clase agrupadas en componentes que constituyen la API de persistencia desarrollada para la plataforma Java EE e incluida en el estándar EJB 3.0 como parte de JSR 220, aunque su uso no se limita a los componentes software EJB. Permite 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.</p>	
Nombre:	Propósito:
	<p>Proveer la interacción de la capa de presentación con la de negocio.</p>
	Observaciones:
	<p>No se encuentra en ninguna capa en específico, sino que se encuentra ubicado entre la capa de Presentación y la de Negocio.</p>
Descripción:	
<p>Es el controlador de SEAM que capta las peticiones derivadas de la interacción del usuario después de interactuar con el Servlet de JSF. Enrutar las peticiones hacia los Beans que posibilitarán darle respuesta a la petición solicitada. Interviene en la integración de las capas de presentación y negocio.</p>	

Tabla3 Clases Comunes

3.2.3 Emitir Certificado. Descripción.

Capa de Presentación	
Nombre:	Propósito:
 <p style="margin-top: 10px;">. Clase Agregar.siaps</p>	<p>Proveer la interacción con el usuario.</p>
Descripción:	
<p>La clase Agregar.siaps es una página web que se ejecuta del lado del cliente sobre un navegador. Permitirá insertar todos los datos necesarios para proceder con el certificado del paciente y que todo quede correctamente registrado. 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.</p>	
Nombre:	Propósito:
 <p style="margin-top: 10px;">frmsiaps</p>	<p>Enviar los datos a las páginas servidoras.</p>
Descripción:	
<p>La clase frmsiaps contiene una colección de elementos de entrada que están contenidos en la página cliente para insertar todos los datos necesarios para proceder con el certificado y que todo quede</p>	

CAPÍTULO 3. DISEÑO DEL SISTEMA

correctamente registrado. Sus atributos son los elementos de entrada del formulario (inputboxes, textareas, checkboxes, entre otros). No tienen operaciones, el método para el paso de los parámetros es \$_POST.

Nombre:

Propósito:



Clase Listar.siaps

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. Permitirá buscar cualquier registro hecho en alguna fecha o por alguna enfermera en específico. 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.

Nombre:

Propósito:



frmsiaps

Enviar los datos a las páginas servidoras.

Descripción:

La clase frmsiaps contiene una colección de elementos de entrada que están contenidos en la página cliente para listar todos los datos necesarios de los certificados de cada paciente. Sus atributos son los elementos de entrada del formulario (inputboxes, textareas, checkboxes, entre otros). No tienen operaciones, el método para el paso de los parámetros es \$_POST.

CAPÍTULO 3. DISEÑO DEL SISTEMA

Capa de Negocio	
Nombre:	Propósito:
 CControlCertificado.java Clase CControlCertificado.java	Proveer una respuesta a las peticiones realizadas en la vista.
Descripción:	
La clase ControlCertificado.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 contienen. Se encarga de gestionar la acción de insertar los datos sobre la página cliente correspondiente. Hace uso del Framework EJB que encapsula la lógica de negocio, integrándose con la vista a través del Framework SEAM.	
Nombre:	Propósito:
 CCTbCertificado.java Clase TbCertificado.java	Proveer una respuesta a las peticiones realizadas en la vista.
Descripción:	
La clase TbCertificado.java es una clase que se genera de forma automática para realizar la ingeniería inversa y 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 contienen. Se encarga de gestionar la acción de	

CAPÍTULO 3. DISEÑO DEL SISTEMA

listar los datos solicitados sobre la página cliente correspondiente. Hace uno del Framework EJB que encapsula la lógica de negocio, integrándose con la vista a través del Framework SEAM.

Nombre:

Propósito:



WCertificado.java
Clase WCertificado.java

Proveer una respuesta a las peticiones realizadas en la vista.

Descripción:

La clase WCertificado.java es una clase que se ejecuta del lado del servidor. Hereda todas las funcionalidades de la clase autogenerada TbRegistroVacunaList.java y permite guardar los cambios originales de la misma evitando que los mismos se pierdan al realizar la ingeniería inversa. Hace uno del Framework EJB que encapsula la lógica de negocio, integrándose con la vista a través del Framework SEAM.

Nombre:

Propósito:



CCCertificadoMedico.java
Clase CertificadoMedico.java

Proveer una respuesta a las peticiones realizadas en la vista.

Descripción:

La clase CertificadoMedico.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 contienen. Se encarga de gestionar la acción de modificar los datos de cualquier registro hecho con anterioridad que se solicite sobre la página cliente correspondiente. Hace uno del Framework EJB que encapsula la lógica de negocio, integrándose con la vista a través del Framework SEAM.

CAPÍTULO 3. DISEÑO DEL SISTEMA

Nombre:	Propósito:
 CCTbCertificadoMedicoPorTarjeta.java Clase TbCertificadoMedicoPorTarjeta.java	Proveer una respuesta a las peticiones realizadas en la vista.
Descripción:	
La clase TbCertificadoMedicoPorTarjeta.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 contienen. Se encarga de gestionar la acción de mostrar los datos de la Tarjeta de Vacunación de cualquier paciente escogido con anterioridad que es solicitado desde la página cliente correspondiente. Hace uno del Framework EJB que encapsula la lógica de negocio, integrándose con la vista a través del Framework SEAM.	
Capa de Datos	
Nombre:	Propósito:
<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;">TbCertificadoMedico_ClinicoQ -id : int -version : int -tbCertificado : TbCertificado_ClinicoQ -observaciones : string -casoDe : bool -seguirTrabajando : bool -diasIncapacidad : int -cid : int</div> <p style="text-align: center;">. Clase TbCertificadoMedico_ClinicoQ</p>	Proveer el mapeo con la base de datos.
Descripción:	
La clase TbCertificadoMedico_ClinicoQ es una clase que se ejecuta del lado del servidor. En ella se almacenan los modelos de certificados. Representa una tabla en el modelo de datos relacional y cada	

CAPÍTULO 3. DISEÑO DEL SISTEMA

instancia de esta entidad corresponde a un registro en esa tabla. Es persistida por las clases servidoras para darle una respuesta a las páginas clientes. Hace uso del Framework Hibernate y JPA.

Nombre:

Propósito:

TnClasificCertificadoMedic_ClinicoQ
-id : int -version : int -descripcion : string -cid : int -tbCertificadoMedicPorTarjetas : Set<TbCertificadoMedicPorTarjeta_ClinicoQ>

Clase TnClasificCertificadoMedic_ClinicoQ

Proveer el mapeo con la base de datos.

Descripción:

La clase TnClasificCertificadoMedic_ClinicoQ es una clase que se ejecuta del lado del servidor. En ella se almacenan la clasificación de los certificados. Representa una tabla en el modelo de datos relacional y cada instancia de esta entidad corresponde a un registro en esa tabla. Es persistida por las clases servidoras para darle una respuesta a las páginas clientes. Hace uso del Framework Hibernate y JPA.

Nombre:

Propósito:

TbCertificado_ClinicoQ
-id : int -version : int -tipoCertificado : bool -cid : int -tbCertificadoMedicPorTarjeta : TbCertificadoMedicPorTarjeta_ClinicoQ -tbCertificadoMedico : TbCertificadoMedico_ClinicoQ -trConsultaPacProblemas : Set<TrConsultaPacProblema_ClinicoQ>

Clase TbCertificado_ClinicoQ

Proveer el mapeo con la base de datos.

Descripción:

La clase TbCertificado_ClinicoQ es una clase que se ejecuta del lado del servidor. En ella se almacenan

CAPÍTULO 3. DISEÑO DEL SISTEMA

los certificados médicos y los que son por tarjetas. Representa una tabla en el modelo de datos relacional y cada instancia de esta entidad corresponde a un registro en esa tabla. Es persistida por las clases servidoras para darle una respuesta a las páginas clientes. Hace uso del Framework Hibernate y JPA.

Nombre:**Propósito:****TbCertificadoMedicPorTarjeta_ClinicoQ**

```
-id : int  
-version : int  
-tbCertificado : TbCertificado_ClinicoQ  
-tnClasificCertificadoMedic : TnClasificCertificadoMedic_ClinicoQ  
-nroTarjeta : int  
-cid : int  
-id_periodo : int  
-valido_por : int
```

Clase TbCertificadoMedicPorTarjeta_ClinicoQ

Proveer el mapeo con la base de datos.

Descripción:

La clase TbCertificadoMedicPorTarjeta_ClinicoQ es una clase que se ejecuta del lado del servidor. En ella se almacenan los datos certificados médicos por tarjetas. Representa una tabla en el modelo de datos relacional y cada instancia de esta entidad corresponde a un registro en esa tabla. Es persistida por las clases servidoras para darle una respuesta a las páginas clientes. Hace uso del Framework Hibernate y JPA.

Tabla 4 Emitir Certificado

En este capítulo se representó la propuesta de solución al problema planteado. Los patrones de diseños utilizados fueron detallados, definidos y explicados. Se especificó la estructura del diseño, la cual complementa los requerimientos funcionales y no funcionales. Todo esto estableció un movimiento fundamental para el progreso del flujo de trabajo de implementación.

Capítulo 4. Implementación

Este capítulo constituye la continuidad del Modelo de Diseño. Establece la necesidad de integración del componente web atención integral al paciente diabético con otros sistemas del módulo Clínico Quirúrgico y presenta el Diagrama de Despliegue de la solución propuesta. Se definieron los elementos clave para la seguridad, se describen los métodos más importantes, los estándares de codificación y el tratamiento de errores en la aplicación.

4.1. Propuesta de Integración del componente web Atención al Paciente Diabético.

En la actualidad ninguna aplicación de la salud se concibe como un sistema aislado y el Módulo Clínico Quirúrgico y Medicina Familiar Web no son la excepción, por lo que se hace necesario pensar en su integración con otros sistemas en funcionamiento o en vías de desarrollo.

El componente de atención al paciente diabético perteneciente al módulo Clínico Quirúrgico 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, así como la integración con el alas SIAPS como plataforma única para la gestión, procesamiento y transmisión de la información clínica en el SNS. La comunicación se realizará entre las bases de datos.

4.2. Diagrama de Despliegue.

Los Diagramas de Despliegue muestran las relaciones físicas de los distintos nodos que componen un sistema y el reparto de los componentes sobre dichos nodos. La vista de despliegue representa la disposición de las instancias de componentes de ejecución en instancias de nodos conectados por enlaces de comunicación. [26]

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. De manera general el diagrama de despliegue del Módulo de Clínico Quirúrgico y Medicina Familiar quedaría de la siguiente manera en la fig. 5:

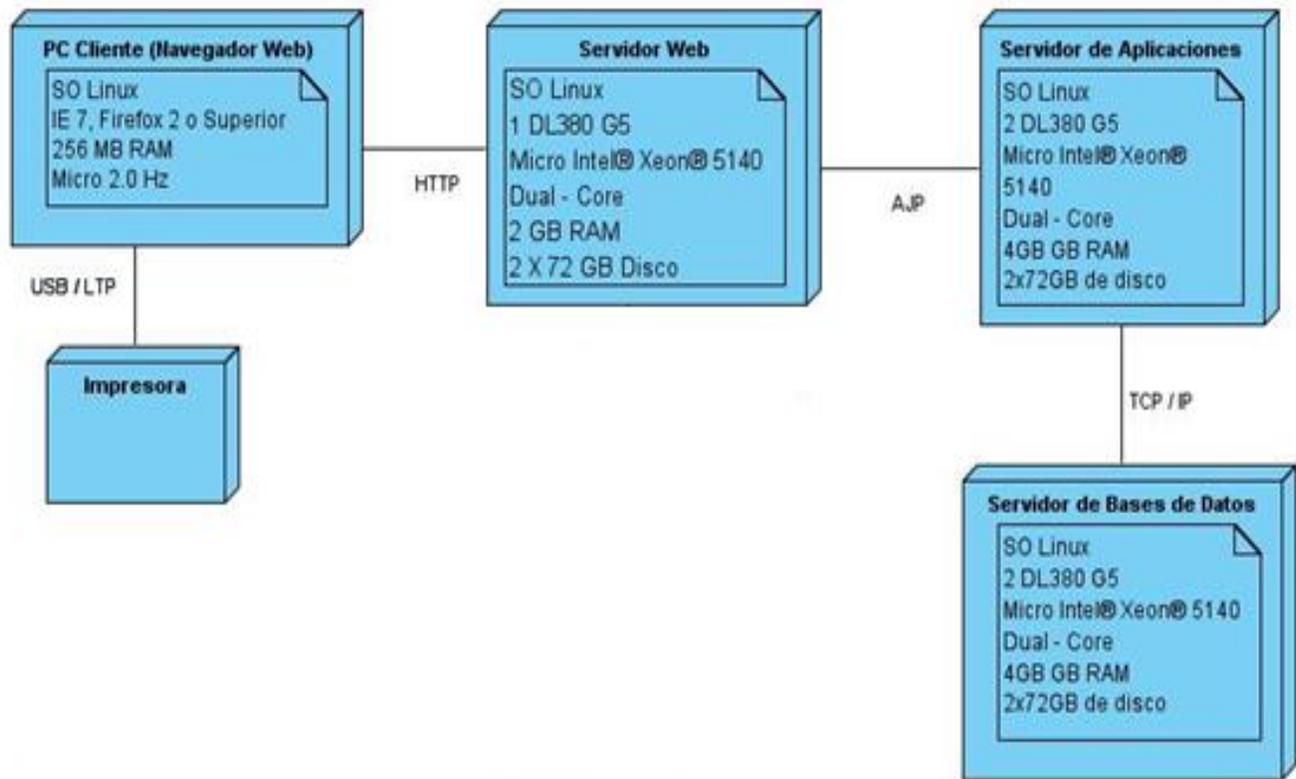


Fig. 5 Diagrama de Despliegue

4.2.1 Descripción de los elementos

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 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, 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.

4.2.2 Procesadores:

Servidor(es) de Aplicaciones:

Sistema Operativo Linux

2 DL380 G5,

Procesador Intel® Xeon® 5140 Dual - Core

4GB de memoria

2x72GB de disco

Servidor de Base de Datos:

Sistema Operativo Linux

1 DL380 G5

Procesador Intel® Xeon® 5140 Dual – Core

4GB de memoria

2x72GB de disco

Servidor Web:

Sistema Operativo Linux

1 DL380 G5

Procesador Intel® Xeon® 5140 Dual - Core

2 GB de memoria

2x72GB de disco

Estaciones Clientes:

Sistema Operativo Linux.

IE 7, Firefox 2 o versiones superiores.

256 Mb de memoria RAM

Microprocesador de 2.0 Hz

4.3. Propuesta de Seguridad.

La seguridad informática se dedica principalmente a proteger la **confidencialidad**, la **integridad** y **disponibilidad** de la información, conformando de esta forma los 3 objetivos principales de la seguridad informática:

Confidencialidad: se refiere a que la información solo puede ser conocida por individuos autorizados

Integridad: se refiere a la seguridad de que una información no ha sido alterada, borrada, reordenada o copiada durante el proceso de transmisión o en su propio equipo de origen.

Disponibilidad: se refiere a la seguridad que tiene la información de que pueda ser recuperada en el momento que se necesite.

Debido al grado tan elevado de importancia que tienen los procesos que se llevan a cabo en los departamentos de clínico quirúrgico y medicina familiar, nace la necesidad de que estos sean realizados con las característica requerida y que el acceso a la información de los pacientes atendidos solo sea permisible para las personas que se encuentren acreditadas, por lo que se tiene en cuenta una serie de requisitos, así como los tres objetivos principales de seguridad de modo que no se pueda afectar el funcionamiento del sistema, ya que el sistemas informático propuesto enfoca el trabajo de la seguridad informática en estos tres objetivos, determinando en todo momento qué necesita ser protegido, de qué necesita protegerse y cómo protegerlo mientras exista.

Entre los requisitos de seguridad y los principales objetivos mencionados anteriormente para lograr un sistema seguro se llevará a cabo un control entre los usuarios y sus contraseñas. Las contraseñas solo podrán ser cambiadas por el usuario o por el administrador del sistema. El sistema también permitirá llevar una traza de todas las operaciones realizadas por cada usuario mediante un registro de actividades por usuario en todo momento. Con el objetivo de obtener la fidelidad de los datos, todo el intercambio entre el sistema y otros sistemas que soliciten información desde cualquier entidad, se realizará de forma cifrada eliminando posibilidades de acceso o modificación de la misma.

4.4. Implementación.

Para encabezar el desarrollo del flujo de trabajo de Implementación se comienza con el resultado del diseño definiéndose cómo se van a organizar las clases y objetos en términos de componentes, cuáles nodo se utilizarán, la ubicación en ellos de los componentes y la estructura de capas de la aplicación, para la cual se utilizaron los patrones GRASP como son:

-El patrón experto que se observa en las clases controladoras las cuales implementan los métodos que tienen funciones específicas dentro del sistema.

-El patrón creador se pone de manifiesto en todas las clases a la hora de crear objetos como instancias de otras clases

-El patrón de alta cohesión se ve en los parámetros que se le ponen a las clases, por ejemplo, si la clase es persona los atributos deben ser características descriptivas de la persona y no de otro sustantivo.

-El patrón controlador ya que hace de intermediario entre la interfaz y el algoritmo que la implementa recibiendo los datos del usuario y enviándolos al método correspondiente.

Además de implementar las clases y subsistemas hallados durante el diseño. Las clases se implementan como componentes de ficheros que contienen código fuente. Uno de los principales propósitos de este flujo de trabajo consiste en desarrollar la arquitectura y el sistema como un todo.

4.4.1 Descripción de algunos métodos

Seguidamente se muestran algunas descripciones de algunos de los métodos del módulo, en la Capa de Negocio.

ConstruirMedidasPaciente ()

Construye un objeto con las medidas del paciente insertadas anteriormente desde la interfaz de usuario, para posteriormente insertarlo en la base de datos.

ConstruirCertificado ()

Construye los objetos de certificado médico y certificado médico por tarjeta para luego construir el objeto Certificado e insertarlo en la base de datos.

4.5. Tratamiento de Excepciones.

Las excepciones son escenarios que pueden presentarse y que requieren de un tratamiento especial. Dichas situaciones pueden ocurrir dentro del programa durante su ejecución, pudiendo interrumpir el correcto funcionamiento del mismo. Si se consigue dominar su programación, la calidad de las aplicaciones que se desarrollen aumentará considerablemente.

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.

4.6. Descripción de elementos e interfaces de comunicación

Las 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.

4.6.1 Características físicas de la conexión

Enlace Público: Se utilizará el enlace público solo desde las estaciones clientes para nutrirse de la información que brinda el sistema SIAPS.

Enlace Privado Se utilizó el enlace privado desde el cortafuego al servidor Web y desde este último hacia el servidor de aplicaciones mediante crossover.

4.7. Estándares de codificación y tratamiento de excepciones.

Con la intención de que exista homogeneidad entre las aplicaciones que se encuentran integradas al Sistema Integral para la Atención Primaria de Salud alas SIAPS se han definido una serie de estándares tanto para el diseño como para la codificación, además como se le dará el tratamiento de errores al módulo para que los códigos fuentes de las aplicaciones y los mensajes que se emitan mantengan una uniformidad.

4.7.1 Estándares de codificación.

CAPÍTULO 4. IMPLEMENTACIÓN

Un estándar de codificación no es más que “un conjunto de reglas de notación y nomenclatura, específicas de cada lenguaje de programación, que se usan y se siguen durante la fase de implementación (codificación) de una aplicación y reducen perceptiblemente el riesgo de que los desarrolladores introduzcan errores que no son detectados por los compiladores, reduciendo el tiempo y coste de las actividades de depuración y pruebas necesarias para la detección y corrección de los mismos.” [27]

Las convenciones de código o estándares de codificación son importantes para los programadores por un gran número de razones:

- ✓ El 80% del coste del código de un programa va a su mantenimiento.
- ✓ Casi ningún software lo mantiene toda su vida el autor original.
- ✓ Las convenciones de código mejoran la lectura del software, permitiendo entender código nuevo mucho más rápidamente y más a fondo.
- ✓ Si distribuyes tu código fuente como un producto, necesitas asegurarte de que está bien hecho y presentado como cualquier otro producto.

Idioma: Se debe utilizar como idioma el español, las palabras no se acentuarán y estarán unidas.

Algunas de las convenciones más importantes de este estándar son las siguientes:

Declaraciones de Clases e Interfaces.

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

CAPÍTULO 4. IMPLEMENTACIÓN

		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.

Rompiendo Líneas.

Cuando una expresión no entre en una línea, romperla de acuerdo con estos principios:

- Romper después de una coma.
- Romper antes de un operador.
- Preferir roturas de alto nivel (más a la derecha que el "padre") que de bajo nivel (más a la izquierda que el "padre").
- Alinear la nueva línea con el comienzo de la expresión al mismo nivel de la línea anterior.
- Si las reglas anteriores llevan a código confuso o a código que se aglutina en el margen derecho, indentar justo 8 espacios en su lugar.
- Ejemplos de cómo romper la llamada a un método:

```
unMetodo(expresionLarga1, expresionLarga2, expresionLarga3,
          expresionLarga4, expresionLarga5);

var = unMetodo1(expresionLarga1,
                unMetodo2(expresionLarga2,
                          expresionLarga3));
```

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 otros 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:

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

Declaraciones de Clases e Interfaces.

Al codificar clases e interfaces de Java, se siguen las siguientes reglas de formato:

- Ningún espacio en blanco entre el nombre de un método y el paréntesis "(" que abre su lista de parámetros.

- La llave de apertura "{" aparece al final de la misma línea de la sentencia declaración.
- La llave de cierre "}" empieza una nueva línea indentada para ajustarse a su sentencia de apertura correspondiente, excepto cuando no existen sentencias entre ambas, que debe aparecer inmediatamente después de la de apertura "{".

```
class Ejemplo extends Object {
    int ivar1;
    int ivar2;

    Ejemplo(int i, int j) {
        ivar1 = i;
        ivar2 = j;
    }

    int metodoVacio() {}

    ...
}
```

- Los métodos se separan con una línea en blanco.

Referencias a Variables y Métodos de Clase.

Evitar usar un objeto para acceder a una variable o método de clase (static). Usar el nombre de la clase en su lugar. Por ejemplo:

```
metodoDeClase();           //OK
UnaClase.metodoDeClase();  //OK
unObjeto.metodoDeClase();  //EVITAR!
```

Comentarios, líneas y espacios en blanco.

Las líneas en blanco mejoran la facilidad de lectura separando secciones de código que están lógicamente relacionadas.

Se deben usar siempre dos líneas en blanco en las siguientes circunstancias:

- ✓ Entre las secciones de un fichero fuente
- ✓ Entre las definiciones de clases e interfaces.

Se debe usar siempre una línea en blanco en las siguientes circunstancias:

- ✓ Entre métodos
- ✓ Entre las variables locales de un método y su primera sentencia
- ✓ Antes de un comentario de bloque o de un comentario de una línea.
- ✓ Entre las distintas secciones lógicas de un método para facilitar la lectura.

Se deben usar espacios en blanco en las siguientes circunstancias:

Una palabra clave del lenguaje seguida por un paréntesis debe separarse por un espacio.

Indentación y longitud de la línea.

Se deben emplear cuatro espacios como unidad de indentación. La construcción exacta de la indentación (espacios en blanco contra tabuladores) no se especifica. Los tabuladores deben ser exactamente cada 8 espacios.

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

Variables, constantes, clases y métodos.

Todas las instancias y variables de clase empezarán con minúscula y los métodos con mayúscula. Las palabras internas que lo forman (si son compuestas) empiezan con su primera letra en mayúsculas. Los nombres de variables no deben empezar con los caracteres subguión "_" o signo del dólar "\$", aunque ambos están permitidos por el lenguaje.

Los nombres de las variables deben ser cortos pero con significado. Los nombres de variables de un solo carácter se deben evitar, excepto para variables índices temporales.

Los nombres de las clases deben ser sustantivos, cuando son compuestos tendrán la primera letra de cada palabra que lo forma en mayúsculas. Mantener los nombres de las clases, simples y descriptivos. Usar palabras completas, evitar acrónimos y abreviaturas.

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

CAPÍTULO 4. IMPLEMENTACIÓN

En este capítulo se mostró la integración de la solución propuesta con otros componentes, se relaciona con todos los módulos del sistema alas SIAPS. Se presentó el modelo de implementación con el diagrama de despliegue, así como los estándares de codificación definidos por el proyecto para la implementación.

Conclusiones

La realización del presente trabajo ha posibilitado cumplir con el objetivo propuesto, por lo que se pueden plantear las siguientes conclusiones:

- ✓ El análisis de los procesos de negocio llevados a cabo en el departamento de atención integral al paciente diabético en la APS demostraron la necesidad de desarrollar un sistema informático para la gestión de la información. A través de estos procesos se identificaron las funcionalidades previas a tener en cuenta para el desarrollo del sistema propuesto y se informatizaron los procesos de negocio de pesquiseo y atención integral al paciente diabético.
- ✓ Los sistemas informáticos analizados automatizan algunos de los procesos que se desarrollan en el departamento de atención integral al paciente diabético en la APS. Estos no responden totalmente a las funcionalidades identificadas ya que aún carecen de un sistema unificado, estable, y que responda a los intereses de los profesionales de la salud.
- ✓ Con la aplicación del sistema se logró facilitar la gestión de la información suministrándole al personal de la consulta integral al paciente diabético los registros informatizados que gestionan todo el proceso de esta, así como actualizar en tiempo real las pesquises realizadas.

Recomendaciones

En aras de mejorar el funcionamiento del sistema realizado se proponen las siguientes recomendaciones:

- ✓ La implementación de un sistema que recoja las pesquisas activas en la población, para incidir en la diabetes desde edades tempranas.
- ✓ Realizar la integración con sistemas expertos para apoyar el trabajo del médico en la toma de decisiones, de las consultas de atención integral al paciente diabético.
- ✓ Desplegar el sistema web en los policlínicos y médicos de familia, para facilitar la atención al paciente diabético en la APS.

Referencias Bibliográficas

1. Centro de Atención al diabético de Holguín: Contribuye a elevar calidad de vida - Periódico AHORA. [8 noviembre 2010]. Disponible en la Web: <<http://www.ahora.cu/secciones/salud/5885-centro-de-atencion-al-diabetico-de-holguin-contribuye-a-elevar-calidad-de-vida.html>>.
2. **Herrera, Omaidá Torres. 2009.** *Controlar su Diabetes*. s.l. : Editorial Científico-Técnico, 2009.
—, 2009. *Controlar su Diabetes*. s.l. : Editorial Científico-Técnica, 2009. 2.
3. WHO | World Health Organization. [8 noviembre 2010]. Disponible en la Web: <http://www.who.int/topics/health_systems/es.>.
4. Diabetes: causas y azares. [10 noviembre 2010]. en la Web: <http://www.trabajadores.cu/materiales_especiales/suplementos/salud/enfermedades-cronicas-no-trasmisibles-1/diabetes-causas-y-azares.>.
5. Idem a la referencia 4
6. Idem a la referencia 2
7. Idem a la referencia 2
8. Guía Breve de XHTML. [15 noviembre 2010]. Disponible en la Web: <<http://www.w3c.es/Divulgacion/Guiasbreves/XHTML>>.
9. Arquitectura basada en capas. - Blog de Juan Peláez en Geeks.ms. [15 noviembre 2010]. Disponible en la Web: <<http://geeks.ms/blogs/jkpelaez/archive/2009/05/29/arquitectura-basada-en-capas.aspx>>.
10. Idem a la referencia 9
11. Idem a la referencia 9
12. Sun Microsystems. [6 diciembre 2010]. Disponible en la Web: <http://developers.sun.com/docs/jscreator/help/jsp-jsfel/jsf_expression_language_intro.html>.
13. virtual.usalesiana.edu.bo. [10 marzo 2011]. Disponible en la Web: <<http://www.virtual.usalesiana.edu.bo/web/practica/archiv/despliegue.doc>>.
14. Chapter 14. HQL: The Hibernate Query Language. [6 diciembre 2010]. Disponible en la Web: <<http://docs.jboss.org/hibernate/core/3.3/reference/en/html/queryhql.html>>.

REFERENCIAS BIBLIOGRÁFICAS

15. Rumbaugh James, Jacobson Ivar, Booch Grady. *El Lenguaje Unificado de Modelado*. Manual de Referencia. Pearson Education. 2007. p18.
16. Seam Framework - JBoss Seam. [12 enero 2011]. Disponible en la Web: <<http://www.seamframework.org/Home>>.
17. Idem a la referencia 15
18. Idem a la referencia 15
19. Liu, Xinyu. Developing applications with Facelets, JSF and JS. [12 enero 2011]. Disponible en la Web: <<http://today.java.net/pub/a/today/2006/08/29/developing-with-facelets-jsf-jsp.html>>.
20. Hernandis, José Alberto. Introducción a RichFaces. [20 enero 2011]. Disponible en la Web: <<http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=richFacesJsfIntro>>.
21. Mi primera hora con eclipse.
22. Hernandis, José Alberto. *Versión Cero*. [20 enero de 2011.] Disponible en la web: <<http://www.versioncero.com/noticia/210/visual-paradigm-for-uml>>.
23. EJB in Action
24. Idem a la referencia 21
25. Index of
/materias/Material_Desarrollo_Software/exposiciones2005B/polimorfismoFabricacionPura_G01. [15 febrero 2011]. Disponible en la Web: <http://eisc.univalle.edu.co/materias/Material_Desarrollo_Software/exposiciones2005B/polimorfismoFabricacionPura_G01/>.
26. Idem a la referencia 13
27. Marrero Nieblas, Ariuska y Suarez Rivero, Yoel Odelso. Modulo Banco de sangre del sistema de información hospitalaria alas HIS, 2009.
28. La-federacion-internacional-de-diabetes-subvenciona-con-dos-millones-de-dolares-nueve-proyectos-de-investigacion-mundiales/. [Online] 2010. [Cited: 5 5, 2011.] <http://articulos.sld.cu/diabetes/2010/02/10/la-federacion-internacional-de-diabetes-subvenciona-con-dos-millones-de-dolares-nueve-proyectos-de-investigacion-mundiales/>.

REFERENCIAS BIBLIOGRÁFICAS

29. Idem a la referencia 28

30. Atención integral para el paciente diabético. [Online] 2 14, 2011. [Cited: 5 5, 2011.] <http://www.elperiodico.com.gt/es/20110214/salud/190758>.

31. Atención al paciente diabético en México. [Online] 9 18, 2008. [Cited: 5 5, 2011.] <http://www.infodiabeticos.com/atencion-al-paciente-diabetico-en-mexico>.

32. **Gonzalez Ramirez, Dra. Tania.** Pesquisaje de la Enfermedad Renal Cronica en diabeticos e hipertensos en un area de salud. [Online] [Cited: 5 5, 2011.] http://bvs.sld.cu/revistas/oft/vol8_1_95/oft08195.htm2.

33. La Educación, el punto más débil de la atención integral del paciente diabético. [Online] 6 1997. [Cited: 5 5, 2011.] http://bvs.sld.cu/uats/rtv_files/rtv0697.htm.

34. Atención integral al paciente diabético en Holguín. [Online] 11 14, 2010. [Cited: 5 5, 2011.] <http://www.ahora.cu/secciones/salud/4861-atencion-integral-al-paciente-diabetico-en-holguin-.html>.

35. Idem a la referencia 34

36. **Monge Sánchez, Aurora. 2004.** *LA CONCEPTUALIZACIÓN DE LA SALUD EN EL MODELO DE ATENCIÓN DEL SISTEMA NACIONAL DE SALUD.* Costa Rica : s.n., 2004.

Bibliografía

- ✓ Barrios Leyva, Aliuska. Programa de Atención Primaria de Salud en la Revolución cubana: un poco de historia. [15 febrero 2011]. Disponible en la Web: <http://www.tiempo21.cu/hipertextos/programa_atencion_primaria_salud_cuba_100708.htm>.
- ✓ [23 febrero 2011]. Disponible en la Web: <<http://www.ahora.cu/secciones/salud/4861-atencion-integral-al-paciente-diabetico-en-holguin-.html>>.
- ✓ Liu, Xinyu. Developing applications with Facelets, JSF and JS. [12 enero 2011]. Disponible en la Web: <<http://today.java.net/pub/a/today/2006/08/29/developing-with-facelets-jsf-jsp.html>>.
- ✓ Arquitectura basada en capas. - Blog de Juan Peláez en Geeks.ms. [15 noviembre 2010]. Disponible en la Web: <<http://geeks.ms/blogs/jkpelaez/archive/2009/05/29/arquitectura-basada-en-capas.aspx>>.
- ✓ Category:OWASP PHP.../es - OWASP. [2 marzo 2011]. Disponible en la Web: <http://www.owasp.org/index.php/Category:OWASP_PHP.../es>.
- ✓ Centro de Atención al diabético de Holguín: Contribuye a elevar calidad de vida - Periódico AHORA. [5 diciembre 2010]. Disponible en la Web: <<http://www.ahora.cu/secciones/salud/5885-centro-de-atencion-al-diabetico-de-holguin-contribuye-a-elevar-calidad-de-vida.html>>.
- ✓ Chapter 14. HQL: The Hibernate Query Language. [5 diciembre 2010]. Disponible en la Web: <<http://docs.jboss.org/hibernate/core/3.3/reference/en/html/queryhql.html>>.
- ✓ Guía Breve de XHTML. [12 diciembre 2010]. Disponible en la Web: <<http://www.w3c.es/Divulgacion/Guiasbreves/XHTML>>.
- ✓ Index of /materias/Material_Desarrollo_Software/exposiciones2005B/polimorfismoFabricacionPura_G01. [9 enero 2011]. Disponible en la Web: <http://eisc.univalle.edu.co/materias/Material_Desarrollo_Software/exposiciones2005B/polimorfismoFabricacionPura_G01/>.
- ✓ Introducción a los Sistemas Distribuidos | Asociación de Usuarios de GNU/Linux de Castilla y León (AUGCyL). [2 marzo 2011]. Disponible en la Web: <<http://www.augcyl.org/?q=glol-intro-sistemas-distribuidos>>.

- ✓ La importancia de los procesos de negocio en las IT | GestioPolis. [2 marzo 2011]. Disponible en la Web: <<http://www.gestiopolis.com/operaciones/reingenieria-de-procesos-de-negocios.htm>>.
- ✓ OMS | Sistemas de salud. [4 marzo 2011]. Disponible en la Web: <http://www.who.int/topics/health_systems/es/>.
- ✓ Patrón "Modelo-Vista-Controlador". [8 enero 2011]. Disponible en la Web: <<http://www.proactiva-calidad.com/java/patrones/mvc.html>>.
- ✓ Programa de Atención Primaria de Salud — 50 Aniversario de la Revolución Cubana. [10 marzo 2011]. Disponible en la Web: <<http://revolucioncubana.cip.cu/logros/modelo-social-socialista/salud/programa-de-atencion-primaria-de-salud>>.
- ✓ Programa de Higiene y Epidemiología — 50 Aniversario de la Revolución Cubana. [16 marzo 2011]. Disponible en la Web: <<http://revolucioncubana.cip.cu/logros/modelo-social-socialista/salud/programa-de-higiene-y-epidemiologia>>.
- ✓ Programa Integral de Atención al Adulto Mayor — 50 Aniversario de la Revolución Cubana. [18 marzo 2011]. Disponible en la Web: <<http://revolucioncubana.cip.cu/logros/modelo-social-socialista/salud/programa-integral-de-atencion-al-adulto-mayor>>.
- ✓ Seam Framework - JBoss Seam. [8 enero 2011]. Disponible en la Web: <<http://www.seamframework.org/Home>>.
- ✓ Sun Microsystems. [16 enero 2011]. Disponible en la Web: <http://developers.sun.com/docs/jscreator/help/jsp-jsfel/jsf_expression_language_intro.html>.
- ✓ Tratamiento de excepciones. [20 marzo 2011]. Disponible en la Web: <<http://elvex.ugr.es/decsai/builder/intro/6.html>>.
- ✓ VII International Congress on Health Informatics — Informática en salud 2009. [22 marzo 2011]. Disponible en la Web: <<http://www.informatica2009.sld.cu/>>.
- ✓ virtual.usalesiana.edu.bo. [20 enero 2011]. Disponible en la Web: <<http://www.virtual.usalesiana.edu.bo/web/practica/archiv/despliegue.doc>>.
- ✓ WHO | World Health Organization. [23 enero 2011]. Disponible en la Web: <http://www.who.int/topics/health_systems/es.>.

- ✓ Atención al paciente diabético en México. [Online] 9 18, 2008. [Cited: 5 5, 2011.] <http://www.infodiabeticos.com/atencion-al-paciente-diabetico-en-mexico>.
- ✓ Atención integral al paciente diabético en Holguín. [Online] 11 14, 2010. [Cited: 5 5, 2011.] <http://www.ahora.cu/secciones/salud/4861-atencion-integral-al-paciente-diabetico-en-holguin-.html>.
- ✓ Atención integral para el paciente diabético. [Online] 2 14, 2011. [Cited: 5 5, 2011.] <http://www.elperiodico.com.gt/es/20110214/salud/190758>.
- ✓ [cjimenez.pdf](#). [3 febrero 2011]. Disponible en la Web: <http://www.inf.udec.cl/~revista/ediciones/edicion9/cjimenez.pdf>.
- ✓ Contreras, José L. La Importancia De Las Tic en Aps.
- ✓ Editorial Ciencias Médicas. Revista Cubana de Medicina General Integral - La informatización de la atención primaria de salud. [23 marzo 2011]. Disponible en la Web: http://scielo.sld.cu/scielo.php?pid=S0864-21252006000400012&script=sci_arttext.
- ✓ Editorial Ciencias Médicas. Revista Cubana de Salud Pública - Informática en la salud pública cubana. [2 abril 2011]. Disponible en la Web: http://scielo.sld.cu/scielo.php?pid=S0864-34662006000300015&script=sci_arttext.
- ✓ 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.
- ✓ García de Jalón Javier, Rodríguez Iñigo Mingo José Ignacio, Alfonso Brazález Aitor Imaz, Larzabal Alberto, Calleja Jesús, García Jon. Aprende Java como si estuviera en primero. Escuela Superior de Ingenieros Industriales. Universidad de Navarra. España. Enero 2000.
- ✓ Hernandis, José Alberto. Introducción a RichFaces. [20 enero 2011]. Disponible en la Web: <http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=richFacesJsflIntro>.
- ✓ <http://www.imss.gob.mx/NR/rdonlyres/43F50D4B-5CBE-4307-B95D-E12B96B8E885/0/mejora.pdf>
- ✓ La Educación, el punto más débil de la atención integral del paciente diabético. [Online] 6 1997. [Cited: 5 5, 2011.] http://bvs.sld.cu/uats/rtv_files/rtv0697.htm.

- ✓ La-federacion-internacional-de-diabetes-subvenciona-con-dos-millones-de-dolares-nueve-proyectos-de-investigacion-mundiales/. [Online] 2010. [Cited: 5 5, 2011.] <http://articulos.sld.cu/diabetes/2010/02/10/la-federacion-internacional-de-diabetes-subvenciona-con-dos-millones-de-dolares-nueve-proyectos-de-investigacion-mundiales/>.
- ✓ Marrero Nieblas, Ariuska y Suarez Rivero, Yoel Odelso. Modulo Banco de sangre del sistema de información hospitalaria alas HIS, 2009.
- ✓ [Online] [Cited: 5 5, 2011.] http://bvs.sld.cu/revistas/oft/vol8_1_95/oft08195.htm2.
- ✓ Gonzalez Ramirez, Tania Dra. Pesquisaje de la Enfermedad Renal Cronica en diabeticos e hipertensos en un area de salud. rtv0697.pdf. [8 febrero 2011]. Disponible en la Web: <<http://www.sld.cu/galerias/pdf/sitios/vigilancia/rtv0697.pdf>>.

Glosario de Términos

Atención integral: la atención integral consiste en abordar las necesidades y problemas de salud, en los individuos, familias o comunidades, en una sociedad concreta, que define las condiciones de vida, perfilando un determinado nivel de salud en los grupos sociales que la conforman

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

Diabetes: es una enfermedad crónica que se caracteriza por niveles de glucosa en sangre elevados (glucemia), como consecuencia de un déficit en la secreción de insulina, acompañado, en muchos casos, de una disminución de la respuesta de las células del organismo al efecto de la insulina

Equipos Básicos de Salud: Binomio conformado por el médico y enfermera de la familia, que atiende una población geográficamente determinada, que puede estar ubicado en la comunidad, centros laborales o educacionales.

Informatizar: Proceso de aplicar sistemas o equipos informáticos al tratamiento de la información.

Pesquisaje: es el procedimiento clínico o de laboratorio que se realiza en una población o grupo poblacional, para detectar las personas con un riesgo incrementado de padecer una enfermedad o problema de salud específico que justifique acciones de diagnóstico y/o seguimiento.

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

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

