



UNIVERSIDAD DE SANCTI SPÍRITUS

“JOSÉ MARTÍ PÉREZ”

Facultad de Ingeniería

Ingeniería Informática

Trabajo diploma para optar por el título de
Ingeniería Informática

Título: *Sistema Experto para la clasificación temprana de Dengue Severo en el Instituto Pedro Kourí (IPK) de Ciudad de la Habana.*

Autor: *Marysleivy Martín Roque.*

Tutor: *Dra.C Lydia Rosa Ríos Rodríguez.*

Sancti Spíritus

Junio del 2010

Agradecimientos

Como toda investigación, esta es parte de la dedicación y el esfuerzo, y no se hubiera podido concluir, sin el apoyo y la cooperación de muchas personas, por lo que intento ofrecerles a todas ellas mi más sincera gratitud.

A mi mamá por todo el amor, por el apoyo y las fuerzas que siempre me ha dado.

A mi papá y hermano que aunque lejos siempre me apoyaron.

A mi novio, inspirador de mi mejor desempeño.

A mis abuelos, tíos y tías por su apoyo diario.

A Orlando A. Brunet Arias, por su amistad, dedicación y apoyo incondicional.

A todos mis compañeros de grupo por su apoyo durante estos años de la carrera.

A mi tutora que con su experiencia y conocimientos me ayudó a que fuera posible la culminación de este trabajo.

A todos mis profesores por ayudarme a crecer como persona y como profesional.

En general a todos los que participaron directamente en la investigación, o los que me estimularon con su ánimo, deseo significar con respeto y cariño mi agradecimiento sincero.

Dedicatoria

A todos mis seres queridos, porque sin ustedes no hubiese llegado hasta aquí.

Resumen

La Inteligencia Artificial (IA) es una rama de la ciencia de la computación cuyo principal objetivo es llevar a la computadora las amplias capacidades del pensamiento humano. Dentro del campo de la IA juegan un papel importante los Sistemas Expertos.

En esta investigación se diseña e implementa un sistema experto capaz de multiplicar la experiencia lograda por los médicos del IPK en la clasificación temprana de pacientes con Dengue Severo. Para ello se utilizó el lenguaje de programación lógica Prolog, el Borland Delphi para la interfaz de usuario y las reglas de producción como forma de representar el conocimiento. La metodología empleada es la propuesta por el Dr.C Mateo Lezcano en su libro: Prolog y los Sistemas Expertos. También se realiza un estudio de la factibilidad de desarrollar el sistema.

El sistema ayuda a los médicos no especialista en Dengue a clasificar tempranamente los pacientes como posible caso crítico, posible caso de alarma o posible caso de alerta.

Abstract

Artificial Intelligence (AI) is a branch of computer science whose main objective is to bring the computer the extensive capabilities of human thought. Within the field of AI play an important role Expert Systems.

In this research is designed and implemented an expert system able to multiply the experience gained by doctors of the IPK in the early classification of patients with Severe Dengue. For this was used the programming language Prolog, Borland Delphi for the user interface and production rules as a way of representing knowledge. The methodology used is that proposed by the Dr.C Mateo Lezcano in his book: Prolog and Expert Systems. Also a study of the feasibility of developing the system.

The system helps the doctors non specialists in Dengue to classify patients like early possible critical case, possible case of alarm or possible case of alert.

ÍNDICE

INTRODUCCIÓN	1
Capítulo 1: La inteligencia Artificial y los Sistemas Expertos	6
1.1 Conocimiento como componente del sistema.	7
1.2 Sistemas Basados en Conocimiento.....	8
1.3 Formas de Representación del Conocimiento.	9
1.3.1 Los Frames.	10
1.3.2 Los Script.	11
1.3.2 Las Reglas de Producción.	11
1.4 Relación entre las Formas de Representación del Conocimiento.....	13
1.4.1 Relación entre frames y reglas de producción.	13
1.4.2 Relación entre frames y redes semánticas.	14
1.4.3 Relación entre frames y los script.	14
1.4.4 Relación entre redes semánticas y programación lógica.	14
1.5 Tipos de Sistemas Basados en el Conocimiento.	14
1.5.1 Redes Neuronales.	16
1.5.2 Redes Semánticas.	16
1.5.3 Algoritmos Genéticos.	17
1.5.4 Sistemas Basados en Frames.	17
1.5 .5 Sistemas Basados en Reglas de Producción.....	18
1.6 Sistemas Expertos.	19
1.6.1 Arquitectura básica de los sistemas expertos.	20
1.6.2 El equipo de desarrollo de los Sistemas Expertos	21
1.6.3 Aspectos esenciales para el diseño de un Sistema Experto.	22
1.7 Los Sistemas Expertos y sus aplicaciones.....	22
1.7.1 Los Sistemas Expertos para la Salud.	24
1.8 El Dengue.	24
1.8.1 Clasificación temprana de Dengue Severo.	25

1.9 Metodología para los Sistemas Expertos.....	26
1.9.1 Método clásico para el diseño de un Sistema Experto.....	26
1.10 El Proceso Unificado de desarrollo de software.....	27
1.11 Tecnologías y Herramientas	28
1.11.1 Prolog.....	28
1.11.2 SWI-Prolog.....	29
1.11.3 AMZI! Prolog.	29
1.11.4 Delphi.....	29
1.12 Método para la estimación del esfuerzo llamado COCOMO II.....	30
1.13 Conclusiones.....	30
Capítulo 2: Análisis, diseño e implementación de la propuesta de solución. ...	32
2.1 Identificación	32
2.2 Establecer conceptos.....	38
2.3 Formalización.....	38
2.3.1 Clasificación del conocimiento	39
2.4 Implementación.....	41
2.4.1 Formalización de las reglas.....	42
2.4.1.1 ¿Cómo trabajar con AMZI! Prolog?.....	46
2.4.2 Diseño de los módulos del sistema.....	47
2.4.3 Diálogos del sistema.....	47
2.4.4 Diagrama de casos de uso del sistema.....	52
2.4.5 Descripción de los casos de usos.....	53
2.4.6 Implementación de la interfaz del usuario.....	55
2.5 Diseño de la interfaz gráfica.....	58
2.6 Conclusiones.....	59
Capítulo 3: Estudio de factibilidad.	60
3.1 Introducción.....	60
3.2 Planificación.....	60
3.3 Costos.....	65
3.4 Beneficios tangibles e intangibles.....	68
3.5 Análisis de costos y beneficios.....	69
3.6 Conclusiones.....	69

CONCLUSIONES	70
RECOMENDACIONES	71
REFERENCIAS BIBLIOGRÁFICAS.....	72
ANEXOS	75

ÍNDICE DE TABLAS

TABLA 1.1 TIPOS DE SISTEMAS BASADOS EN EL CONOCIMIENTO.	15
TABLA 3.1 ENTRADAS EXTERNAS.....	60
TABLA 3.2 SALIDAS EXTERNAS.....	63
TABLA 3.3 PETICIONES.	63
TABLA 3.4 FICHEROS LÓGICOS INTERNOS.	64
TABLA 3.5 PUNTOS DE FUNCIÓN SIN AJUSTAR	64
TABLA 3.6 INSTRUCCIONES FUENTES.....	65
TABLA 3.7 DEFINICIÓN DE LOS MULTIPLICADORES DE ESFUERZO (MEJ).....	65
TABLA 3.8 DEFINICIÓN DE LOS VALORES DE LOS FACTORES DE ESCALA (SFI).	66
TABLA 3.9 CÁLCULO DEL ESFUERZO, TIEMPO DE DESARROLLO, CANTIDAD DE HOMBRES Y COSTO.	68

ÍNDICE DE FIGURAS

FIGURA. 2.1- DISEÑO DEL MÓDULO.....	47
FIGURA. 2.2- DIAGRAMA DE CASO DE USO DEL SISTEMA.....	52

*La Inteligencia Artificial de hoy en día
se centra en buscar nuevas maneras de conectar
las personas con las computadoras,
las personas al conocimiento,
las personas al mundo físico
y las personas con las personas.*

Patrick Winston

Introducción

La Inteligencia Artificial (IA), si bien como ciencia apareció durante el siglo pasado, ha tenido una larga historia que podemos descubrir desde los orígenes del pensamiento humano. Actualmente, posee múltiples aplicaciones en la casa, la industria y el comercio. Ha dejado de ser una ciencia de los laboratorios y ha entrado a formar parte de la vida cotidiana.

La Inteligencia Artificial es una rama de la ciencia de la computación cuyo principal objetivo es llevar a la computadora las amplias capacidades del pensamiento humano y, para ello, estas se convierten en “entes inteligentes” con la creación de *software* que les permite imitar algunas de las funciones del cerebro humano en aplicaciones particulares. El fin no es reemplazar al hombre, sino proveerlo de una herramienta poderosa para asistirlo en su trabajo. (García Valdivia, Bello Pérez, Gálvez Lío, Lezcano Brito, & Reynoso Lobato, 2000)

En 1969 publicaciones pioneras sobre IA en Cuba y América Latina fueron realizadas por un grupo de profesores universitarios de los Departamentos de Computación y de Lógica Matemática de la Universidad de La Habana, interesados en el estudio de la programación no numérica, en particular de problemas simbólicos y de la posible automatización de sus métodos de solución. También en la Universidad Central “Martha Abreu ” de la Villas en 1985, en el departamento docente de Ciencia de la Computación y en la facultad de Ingeniería Eléctrica se organizó un grupo de investigación, con profesores que junto con estudiantes elaboraron *software* para diseñar e implementar sistema inteligentes de ayuda al diseño y sistema inteligentes para la enseñanza . (García Valdivia, Bello Pérez, Gálvez Lío, Lezcano Brito, & Reynoso Lobato, 2000)

Este amplio campo trata problemas, que en un principio parecían imposibles, intratables y difíciles de formular utilizando ordenadores (Winton, 2005). Las primeras dificultades que se

trató de solucionar con la IA fueron puzzles, juegos de ajedrez y traducción de textos a otro idioma. Los investigadores de esta ciencia se concentran en el control automático, las bases de datos inteligentes y la ingeniería del software (diseños de entornos de programación inteligente). Otros investigadores trabajan en el desafío del reconocimiento de modelos donde se espera un rápido progreso en este campo que comprende la comprensión y la suma del habla, el proceso de imágenes y la visión artificial. (Coello Coello, La Computación Evolutiva en el Contexto de la Inteligencia Artificial, 2005)

Sin embargo, uno de los principales objetivos de los investigadores actuales en Inteligencia Artificial es la reproducción automática del razonamiento humano, donde juegan un papel importante los Sistemas Expertos.

Los Sistemas Expertos son una expresión de los Sistemas Basados en el Conocimiento (SBC), los cuales permiten la creación de máquinas que razonan como el hombre, restringiéndose a un espacio de conocimientos limitados. En teoría pueden razonar siguiendo los pasos que seguiría un experto humano (médico, analista, empresario, etc.) para resolver un problema concreto. (Bello, Gálvez, García, & Lezcano, 1995).

Los inicios de los SE datan de mediados de los años sesenta. Durante esta década los investigadores Alan Newell y Herbert Simon desarrollaron un programa llamado GPS (General Problem Solver; solucionador general de problemas). Podía trabajar con criptoaritmética, con las torres de Hanoi y con otros problemas similares. Lo que no podía hacer el GPS era resolver problemas del mundo real, tales como un diagnóstico médico. (Reyes Olivans, 2005).

Para desarrollar Sistema Expertos del tipo "*clasificación*" y "*selección*", Lezcano Brito recomienda como lenguaje de programación lógica, Prolog. (Lezcano Brito, 1995).

Prolog es un lenguaje de programación hecho para representar y utilizar el conocimiento que se tiene sobre un determinado dominio. Un programa escrito en Prolog consiste en un conjunto de reglas y hechos que expresan relaciones entre los objetos del dominio. (Stair & Reynolds, 2005)

Los Sistemas Expertos se utilizan en una gran diversidad de campos, por ejemplo en las telecomunicaciones, la industria, la aeronáutica, la electrónica, el transporte, la educación y

otros. También se han utilizado en el campo de la medicina, donde han alcanzado notable relevancia en áreas como la neurofisiología, la urología y las enfermedades infecciosas.

Entre las enfermedades infecciosas en Cuba, el Dengue ocupa un lugar importante, esta es una enfermedad viral transmitida por artrópodos de mayor importancia médica en los humanos. Es además la fiebre hemorrágica viral de mayor extensión geográfica mundial. Se reporta en las zonas tropicales y subtropicales del planeta, donde viven alrededor de 2 500 millones de personas. Su distribución geográfica se relaciona al hábitat de su principal vector transmisor, el mosquito *Aedes aegypti* (Pacheco Arencibia, 2009).

Esta enfermedad es considerada como un serio problema de salud para la humanidad y de un costo elevado, ya que además de la atención médica hay que adicionarle las medidas de control y las horas laborales perdidas por los pacientes.

En los últimos 30 años Cuba ha presentado varias epidemias de Dengue que han sido eliminadas, sin presentar transmisión del virus en los períodos interepidémicos. Esta es una situación excepcional en el mundo, la cual ha permitido realizar numerosos estudios y observaciones que han contribuido al conocimiento de diferentes aspectos de la enfermedad (M, y otros, 2006). El Instituto Pedro Kourí (IPK) de Ciudad de la Habana ha sido uno de los centros de salud encargado de la atención de los casos presentados en las diferentes epidemias que ha afrontado Cuba. Dicho Instituto tiene como misión contribuir a mejorar la calidad de vida de la población cubana y del resto de la comunidad internacional, brindando servicios especializados y de alta tecnología en el diagnóstico, la atención médica, la docencia y en el desarrollo de proyectos de investigación; dirigidos a la prevención, control y eliminación de las enfermedades transmisibles, principalmente en los aspectos relacionados al Dengue.

Mediante la caracterización clínica del Dengue Severo en adultos es posible identificar peculiaridades de esa forma clínica y elementos asociados a severidad que contribuyan a la prevención de este, sin embargo existen pocos especialistas en el país, capacitados en el tema para enfrentar, en un momento dado, una gran epidemia. Todo lo antes expuesto nos lleva a la necesidad de contar con recursos informáticos capaces de multiplicar la experticia lograda por los médicos del IPK en la clasificación temprana de los pacientes con Dengue Severo que contribuya a facilitar la prevención este, siendo esta la **situación problemática**

de la presente investigación.

Esta situación problemática ha dado lugar al siguiente **Problema Científico**:

¿Cómo contribuir a la clasificación temprana de Dengue Severo mediante un Sistema Experto en el IPK de Ciudad de la Habana?

Como **antecedente** a este trabajo se encontró un SE que trata el tema de la enfermedad conocida como Dengue; este fue realizado por el Dr. Alexander Sánchez Arbeláez especialista en medicina general. Dicho sistema utiliza un Sistema Basado en Casos; este indica si un paciente tiene Dengue o no, partiendo de los diferentes síntomas que estos presentan; (Sánchez Arboláez, 2006) el sistema no realiza una clasificación de pacientes con Dengue Severo. Este sistema no cuenta con una interfaz visual.

El **Objeto de Estudio** de la presente investigación se define como:

La clasificación temprana de Dengue Severo.

Y el **Campo de Acción** como:

La clasificación de Dengue Severo a través de un Sistema Experto.

Teniendo en cuenta lo planteado anteriormente, y como solución al problema se define como **objetivo general** de este trabajo: Desarrollar un Sistema Experto para la clasificación temprana de Dengue Severo en el IPK de Ciudad de la Habana.

Del objetivo general se plantean las siguientes **Preguntas Científicas**:

1. ¿Cuáles son los fundamentos teóricos y metodológicos que permitan desarrollar un recurso informático para la clasificación temprana de Dengue Severo?
2. ¿Cómo diseñar un Sistema Experto, basado en la metodología seleccionada para la clasificación temprana de Dengue Severo en el IPK de Ciudad de la Habana?
3. ¿Será factible el desarrollo de un Sistema Experto para la clasificación temprana de Dengue Severo en el IPK de Ciudad de La Habana?

4. ¿Cómo implementar un Sistema Experto para la clasificación temprana de Dengue Severo en el IPK de Ciudad de la Habana?

Para responder a las preguntas científicas se plantean las **Tareas de Investigación:**

1. Determinación de los fundamentos teóricos y metodológicos que permitan desarrollar un recurso informático para la clasificación temprana de Dengue Severo.
2. Diseño de un Sistema Experto utilizando la metodología seleccionada para contribuir a la clasificación temprana Dengue Severo en el IPK de Ciudad de la Habana.
3. Realización del estudio de los costos-beneficios de un Sistema Experto para la clasificación temprana de Dengue Severo en el IPK de Ciudad de la Habana.
4. Implementación de un Sistema Experto para contribuir a la clasificación temprana de Dengue Severo en el IPK de Ciudad de la Habana.

El presente trabajo se estructura de la siguiente forma: introducción, tres capítulos, conclusiones, recomendaciones y bibliografía.

En la **Introducción** se expresa los aspectos más generales de la investigación del diseño teórico y metodológico.

En el **Capítulo 1** se trata el tema de la Inteligencia Artificial y la relación que tienen los Sistemas Expertos con esta rama de la ciencia de la computación. También se abordan elementos de la enfermedad conocida como Dengue.

En el **Capítulo 2** se describen aspectos del dominio y los conceptos descriptivos de sus propiedades, las relaciones que se establecen entre ellos, se organiza el conocimiento y posteriormente se pasa a su formalización e implementación.

En el **Capítulo 3** se hace el estudio de la factibilidad. Se describe el estudio de factibilidad realizado a la aplicación, así como un análisis de los costos-beneficios que esta trae consigo.

*Para tener acceso a la producción moderna
y dominar las tecnologías avanzadas es imprescindible
instruir a los hombres y mujeres que las van a manejar,
formarlos para el mayor conocimiento de sus especialidades.*

Fidel Castro.

Capítulo 1: La inteligencia Artificial y los Sistemas Expertos

Desde la antigüedad, el hombre vio en la naturaleza un modelo, que por su perfección y acabado, soñó copiar para reproducir lo alcanzado por ella. Estos intentos no siempre tuvieron resultados satisfactorios ni encontraron un camino expedito, la leyenda de *Icaro* es un buen ejemplo de un intento fracasado: no bastaba en inspirarse en el vuelo de los pájaros para poder volar. Solo cuando dirigieron su atención a las expresiones matemáticas propuestas por la termodinámica pudieron diseñar y echar a volar el aeroplano. Las matemáticas se erigieron, entonces, en un sólido pilar para la modelación de múltiples procesos.

Pero el hombre no solo aspiró a volar, también sintió un profundo interés por crear máquinas inteligentes: autómatas capaces de realizar cualquier tarea. Los primeros pasos estuvieron ligados a la Cibernética, la cual fue pionera en tratar de copiar la naturaleza, así como de descubrir las analogías entre esta y las máquinas.

Sin embargo, el modelo cibernético fue acusado de ser demasiado general y no estar sustentado sobre leyes propias. No obstante, aportó conceptos importantes a la teoría de la información, la biología, la neurología... y dio lugar al surgimiento de una nueva ciencia: la Inteligencia Artificial.

La Inteligencia Artificial nació bajo la expectativa de desarrollar sistemas inteligentes, capaces de ejecutar actividades consideradas propias e inherentes a los seres humanos. Agrupa a todas las ramas que se interesan por la emulación de la inteligencia del hombre y de los procesos que le acompañan y caracterizan: el sentido común, razonamiento, aprendizaje y el habla, entre otras. (Ramos Ríos, 2009)

Son diversos los campos de aplicación de las técnicas de la IA; entre ellos se destacan: los juegos, el procesamiento de lenguaje natural, la robótica, la educación, la programación automática, el reconocimiento de modelos y los sistemas expertos. (Winton, 2005).

En 1965 se empezaron a utilizar técnicas para la resolución de problemas que se caracterizaban por la búsqueda heurística como modelo para la resolución de problemas, y con ellas comenzó la investigación y desarrollo de los sistemas expertos; donde aparece el primer sistema experto, este se llamó DENDRAL (desarrollado por Feigenbaum), el cual era capaz de calcular o descubrir hechos relativos a la estructura molecular a partir de unos datos químicos sin elaborar. (García Valdivia, Bello Pérez, Gálvez Lío, Lezcano Brito, & Reynoso Lobato, 2000)

Entre los años 80 a 85 se produce la revolución de los Sistemas Expertos. En estos cinco años se crearon diversos SE como el DELTA, de General Electric Company, para la reparación de locomotoras diesel y eléctricas. El "Aldo en Disco" para la reparación de calderas hidrostáticas giratorias usadas para la eliminación de bacterias. (Elaine & Kevin, 1994).

A partir de los 90 y con el desarrollo de la informática, se produce un amplio desarrollo en el campo de la IA y los sistemas expertos, pudiéndose afirmar que estos se han convertido en una herramienta habitual en determinadas empresas en la actualidad. La evolución histórica de los métodos utilizados en el desarrollo de los sistemas expertos también se ha producido a medida que se ha ido desarrollando la IA y los diferentes métodos que se han empleado para su resolución. El desarrollo de lenguajes como LISP y PROLOG condicionaron esa evolución, así como investigaciones en diversos campos relacionados. (Castillo, Gutiérrez, & Hadi, 1999)

1.1 Conocimiento como componente del sistema.

En el diseño de un Sistema Experto se le concede significativa importancia a la forma de representación del conocimiento, ya que ésta define la estructura interna de todo tipo de información. Es precisamente debido al énfasis que se hace en la representación y dominio del conocimiento, que los sistemas que utilizan técnicas de inteligencia artificial para lograr conocimiento experto se conocen como **sistemas basados en conocimiento o sistemas de conocimiento**. En este sentido, es importante resaltar que para iniciar la construcción de un sistema experto, es necesario, en primer lugar, tener una clara definición de lo que es el conocimiento en sí como componente del sistema (Lezcano Brito, 1995)

El concepto de conocimiento es una categoría cualitativamente nueva surgida en la teoría de estructura de datos y es una generalización de datos en términos de su contenido de información interna. El conocimiento consiste de **descripciones, relaciones y procedimientos**. Las **descripciones** en una base de conocimiento identifican y diferencian objetos y clases. Son sentencias en algún lenguaje formal, cuyas componentes elementales contienen características primitivas o conceptos. Un sistema de descripción generalmente, incluye reglas o procedimientos para aplicar e interpretar estas descripciones. Las **relaciones** son un tipo particular de descripción. Expresan dependencias y asociaciones entre entidades en la base de conocimiento y por otra parte, especifican las operaciones que se realizan cuando se pretende razonar o solucionar un problema. (Ramírez, 1986)

En la práctica, el conocimiento no aparece de una forma perfectamente definido, que se ajuste nítidamente a estas categorías abstractas y puede tomar muchas formas, tales como, asociaciones empíricas, conceptos, restricciones y regulaciones, entre otras.

Existen dos tipos de conocimiento, independientemente de la forma en que éste pueda presentarse: **público y privado**. El conocimiento **público** incluye las definiciones, hechos y teorías que se definen en libros y otros materiales. El conocimiento **privado** es aquel que poseen los expertos y que generalmente no se encuentra en la literatura. Es el llamado conocimiento heurístico. (Lezcano Brito, 1995)

En la mayoría de los casos, durante la construcción de un sistema experto, se encuentran problemas que no pueden ser expresados a través de algoritmos. En estas condiciones es imprescindible acudir al conocimiento heurístico. La efectividad de las soluciones depende, en estos casos, del uso conveniente del conocimiento para definir los aspectos relevantes y desechar los no relevantes. (García Valdivia, Bello Pérez, Gálvez Lío, Lezcano Brito, & Reynoso Lobato, 2000)

1.2 Sistemas Basados en Conocimiento.

En la década del 70 se reconoció que los métodos de solución de problemas generales eran insuficientes para resolver los problemas orientados a aplicaciones. Se determinó que era necesario conocimiento específico sobre el problema, limitado a los dominios de aplicación de interés, en lugar de conocimiento general aplicable a muchos dominios. Este reconocimiento condujo al desarrollo de Sistemas Basados en el Conocimiento (SBC);

el conocimiento representado en estos es el de los expertos en el dominio (Gálvez Lio, 1998).

El conocimiento debe representar generalidades, en otras palabras, no es necesario representar de forma separada cada situación individual. Este conocimiento debe ser comprendido por las personas que lo propician y debe poseer facilidad de modificarse para corregir errores y reflejar los cambios. La potencia de un SBC radica en el conocimiento que posee este. (Elaine & Kevin, 1994).

García Valdivia, Bello Pérez, Gálvez Lío, Lezcano Brito y Reynoso Lobato (2000) en términos generales, definen que un SBC puede ser:

“Un sistema computarizado que usa conocimiento sobre un dominio para arribar a una solución de un problema de ese dominio. Esta solución es esencialmente la misma que la obtenida por una persona experimentada en el dominio del problema cuando se enfrenta al mismo problema”. (p. 77)

La arquitectura de un SBC de alguna manera refleja la estructura cognitiva y los procesos humanos. La primera parte es la memoria de largo plazo, en la que guarda los hechos (Base de Hechos) y los conocimientos (Base de Conocimientos) acerca del dominio en el que tiene experiencia. La segunda parte es el sistema que realiza la función de razonamiento para resolver problemas (Motor de Inferencia). Finalmente, la tercera parte la conforman las unidades de entrada y salida que permiten la comunicación entre el sistema y su entorno (Winton, 2005).

Generalmente los términos SBC y Sistemas Expertos se usan indistintamente, aunque algunas autores limitan el uso del término SE a aplicaciones donde el conocimiento al nivel de experto es requerido (Gálvez Lio, 1998).

1.3 Formas de Representación del Conocimiento.

Existe diferentes Formas de Representación de Conocimiento (FRC), todos estos esquemas de representación tienen propiedades particulares, pero de forma general ellos deben describir:

- los problemas a que están dirigidos en forma natural, reflejando, tan claramente como sea posible, la estructura de objetos, hechos y relaciones entre ellos,
- describir el conocimiento procesal (procedural) y declarativo en un modo unificado y a partir de perspectivas diferentes, aceptar conocimiento empírico, teórico y heurístico;
- posibilitar una fácil adquisición del conocimiento;
- posibilitar la estratificación y agrupamiento del conocimiento, de acuerdo a su significado y función;
- tener poder expresivo;
- ser comprensible y poseer accesibilidad. (Lezcano Brito, 1995).

La distinción entre cada uno de los esquemas de representación es importante, ya que en dependencia de la forma en que se representa el conocimiento, se emplearán los mecanismos de inferencia (Allende & Rodrigo, 2003).

1.3.1 Los Frames.

Un *frame* es una estructura de datos compleja que contiene un agregado de información acerca de un objeto. La información almacenada en el *frame* se distribuye en diferentes campos llamados *slot* (aspectos), cada *slot* contiene la información sobre un atributo del objeto que se modele. A cada slot se le puede asociar varios tipos de información llamados facetas del *slot*. (García Valdivia, Bello Pérez, Gálvez Lío, Lezcano Brito, & Reynoso Lobato, 2000)

Los *frame* pueden usarse para representar conceptos amplios, clases de objetos, entidades individuales o componentes de objetos y se unen en una jerarquía hereditaria que se encarga de la transmisión de propiedades comunes entre los marcos, sin necesidad de la especificación múltiple de estas propiedades (Elaine & Kevin, 1994).

El lenguaje de los marcos, proporciona al constructor de la base de conocimiento un medio fácil y eficaz para describir los tipos de objetos del dominio que el sistema debe modelar. En este tipo de sistema, el conocimiento se representa a través de las relaciones mutuas que

tienen lugar entre los marcos, de modo que cada relación define una red en el sistema de marcos. Siendo así, podría considerarse la base de conocimiento como un agrupamiento de conceptos abstractos y específicos (representados en marcos), interrelacionados en una compleja red que sería la composición de todas las redes particulares. (Lezcano Brito, 1995)

1.3.2 Los Script.

Un *script* es una FRC similar a un *frames*, pero en lugar de describir un objeto, el *script* describe una secuencia de eventos. Los *scripts* son estructuras de datos complejas designadas para almacenar el conocimiento sobre una secuencia estereotipada de acciones (Elaine & Kevin, 1994).

Este conocimiento le dice a los sujetos que intervienen en la secuencia de eventos lo que puede suceder en una situación, que evento sigue y qué papel debe jugar cada quien en la actividad social que se describe. Para almacenar este conocimiento el *script* usa una serie de *slots* que contienen información acerca de los sujetos, objetos y acciones que están involucradas en el evento (Lezcano Brito, 1995).

Como los *scripts* permiten describir la secuencia de eventos que definen una situación estereotipada, ellos son útiles en la predicción de lo que sucederá en cierta situación aún sin observar algunos eventos; en dar respuestas a las expectativas creadas durante el procesamiento de textos (resolución de pronombres, eliminar la ambigüedad del sentido de las palabras, inferir actividades); en el procesamiento del lenguaje natural; y en representar, procesar y generar historietas (Winton, 2005).

1.3.2 Las Reglas de Producción.

Las reglas de producción son un subconjunto del cálculo de predicados con una componente añadida que indica cómo usar la información almacenada en las reglas, esta componente no es más que la estrategia de solución de conflictos que el usuario tiene que definir para el control de la inferencia. En cuanto a la capacidad de expresión, los sistemas de producción son equivalentes a los sistemas construidos usando un lenguaje de programación lógica. (García Valdivia, Bello Pérez, Gálvez Lío, Lezcano Brito, & Reynoso Lobato, 2000)

Cada regla se divide en dos partes: *antecedente* y *consecuente*. La porción *antecedente* es la *condición* o *premisa* que la regla debe cumplir para que sea verdadera. El *consecuente* es la *conclusión* a la que se arriba si la regla se cumple (Gálvez Lio, 1998).

En general las reglas expresan condiciones, pero ellas pueden tener diferentes interpretaciones, por ejemplo:

Si condición P **entonces** conclusión C

Si situación S **entonces** acción A

Si condición C1 y condición C2 **entonces** condición C no es válida

Cuando el consecuente define una acción, el efecto de satisfacer el antecedente es catalogar la acción para la ejecución. Cuando el consecuente define una conclusión, el efecto es inferir la conclusión. Las reglas definen, a menudo, relaciones lógicas que se pueden definir como conocimiento "categórico", debido a que ellas siempre van a ser absolutamente verdaderas. Sin embargo, en algunos dominios tales como el diagnóstico médico, prevalece el conocimiento probabilístico o inseguro, lo cual se debe a que algunas regularidades empíricas son usualmente válidas sólo para un determinado grado de certidumbre. En tales casos, las reglas se modifican, añadiendo cualificaciones probabilísticas a sus interpretaciones lógicas de la siguiente forma:

Si condición P entonces conclusión C con certidumbre F

donde F es un número que expresa el grado de confianza del autor de las reglas en las conclusiones, dando el antecedente como verdadero. A este número generalmente se le llama factor de certidumbre (Lezcano Brito, 1995).

La más popular y efectiva forma de representación para describir el conocimiento en un sistema experto ha sido la estructura basada en *reglas de producción*. Ésta constituye una particularización evolucionada del modelo computacional introducido en 1972 por Newell para la solución universal de problemas, las cuales son en efecto un subconjunto del cálculo de predicados con un componente descriptivo adicional, que indica cómo se va a utilizar la información almacenada en las reglas durante el proceso de razonamiento. (Coello Coello, La Computación Evolutiva en el Contexto de la Inteligencia Artificial., 2005)

La aplicación de las reglas de producción se debe en gran medida a su poder expresivo para representar un rango útil de reglas de inferencia y a su amplia capacidad de explicación. Se usan para expresar conocimiento deductivo, conocimiento orientado a objetivos y relaciones causales, pero en general son apropiadas cuando el conocimiento a representar aparece naturalmente en forma de reglas, cuando el control del programa es muy complejo o cuando se espera que el programa sea modificado reiteradamente. (Castillo, Gutiérrez, & Hadi, 1999)

1.4 Relación entre las Formas de Representación del Conocimiento.

No se puede decir que realmente una FRC es mejor que otra; se diferencian entre ellos en su expresividad y tratabilidad, por ejemplo los frames son más expresivos pero menos tratables computacionalmente que las redes semánticas. La experiencia ha demostrado que ninguna FRC es capaz de satisfacer todos los requerimientos necesarios para un dominio de aplicación, de ahí que frecuentemente se valore la combinación de dos o más formalismos para una aplicación. Este carácter complementario no existe solamente entre los modelos declarativos, sino que varios autores han valorado también la necesidad de usar en numerosos dominios tanto el conocimiento declarativo como el procedimental; como ejemplo está la combinación de los lenguajes MODULA y PROLOG en un ambiente de programación. (Elaine & Kevin, 1994).

1.4.1 Relación entre frames y reglas de producción.

Las principales ineficiencias de las reglas de producción están en áreas donde los frames son efectivos y viceversa, por eso la integración de ambos ha reportado resultados exitosos. Estos sistemas han demostrado cómo los frames pueden servir como una potente base para los sistemas de producción, tanto mediante el empleo de la riqueza estructural de los lenguajes basados en frames para describir los objetos de las reglas de producción, como para particionar, indexar y organizar un conjunto de reglas. Entre los sistemas que usan la integración de frames y reglas de producción para formar sistemas complejos que ofrecen descripción estructural y control de razonamiento están LOOPS, el KEE, el CENTAUR, el DESINGN, el MicroScope y el KNOBS (García Valdivia, Bello Pérez, Gálvez Lío, Lezcano Brito, & Reynoso Lobato, 2000)

1.4.2 Relación entre frames y redes semánticas.

Las redes semánticas ponen énfasis en unidades pequeñas y primitivas de información, mientras que los frames ponen énfasis en unidades de conocimientos más grandes y complicados. Las redes semánticas pueden ser transformadas en frames y viceversa, pero los dos estilos de representación son convenientes para un rango diferente de dominio. Un ejemplo que integra varias FRC es el ORBIT, el cual es un sistema de programación interactivo implementado en lenguaje LISP para aplicaciones de inteligencia artificial y que se basa en frames, la programación orientada a objeto y las redes semánticas. Ha sido utilizado entre otras aplicaciones para diseñar circuitos VLSI y la creación de un sistema experto para geología. (Elaine & Kevin, 1994).

1.4.3 Relación entre frames y los script.

Estructuralmente, un script es similar al frame, pero sus propósitos son diferentes. El script en lugar de describir un objeto describe una secuencia de eventos; pero ambos lo hacen en forma estereotipada. A diferencia del frame, el script generalmente se presenta en un contexto particular. (García Valdivia, Bello Pérez, Gálvez Lío, Lezcano Brito, & Reynoso Lobato, 2000)

1.4.4 Relación entre redes semánticas y programación lógica.

Los enlaces entre los nodos de las redes semánticas pueden ser representados por predicados binarios. De este modo, a cada tipo de relación corresponde un nombre de predicado y el polimorfismo de la programación lógica facilita el almacenamiento de múltiples enlaces de la red que tienen la misma etiqueta. Inversamente, los predicados binarios del cálculo de predicados se pueden representar gráficamente usando redes semánticas. Los términos del formalismo se pueden representar por los nodos del grafo, mientras que los predicados se corresponden con los arcos. (Elaine & Kevin, 1994).

1.5 Tipos de Sistemas Basados en el Conocimiento.

En la siguiente tabla se muestran varios tipos de Sistemas Basados en el Conocimiento (SBC) (García Valdivia, Bello Pérez, Gálvez Lío, Lezcano Brito, & Reynoso Lobato, 2000):

Tabla 1.1 Tipos de sistemas basados en el conocimiento.				
Nombre	Forma de Representación del Conocimiento (FRC)	Explicación	Método de solución del Problema (MSP)	Fuentes de Conocimiento
Sistemas basados en reglas (SBR)	Reglas de producción	Reglas activas	Usualmente búsqueda primero en profundidad con dirección backward o forward	Expertos, publicaciones, ejemplos
Sistemas basados en frames (SBF)	Frames		Herencia y procedimientos adjuntos	Expertos, publicaciones, ejemplos
Sistemas basados en casos (SBCasos)	Casos	Casos semejantes	Razonamiento basado en casos (búsqueda por semejanza y adaptación de las soluciones)	Ejemplos
Sistemas basados en probabilidades (SBP)	Probabilidades o frecuencias	Valores de probabilidades condicionales	Teorema de Bayes y otras técnicas de inferencia estadística	Ejemplos
Redes expertas	Pesos y alguna otra FRC	Según el modelo simbólico	Cálculo de niveles de activación de las neuronas	Ejemplos
Sistemas basados en modelos	Modelo del artefacto		Razonamiento basado en modelos	Esquemas estructurales y funcionales del artefacto

1.5.1 Redes Neuronales.

El primer modelo matemático de un nodo en una red neuronal fue presentado por McCulloch y Pitts en 1943. Fue un dispositivo binario que usaba entradas binarias, salida binaria, y un umbral fijo de activación. Las Redes Neuronales Artificiales (RNA) consisten en una simulación de las propiedades observadas en los sistemas neuronales biológicos a través de modelos matemáticos recreados mediante mecanismos artificiales (como un circuito integrado, un ordenador o un conjunto de válvulas). El objetivo es conseguir que las máquinas den respuestas similares a las que es capaz de dar el cerebro que se caracterizan por su generalización y su robustez. (Kasabov, 1998).

Las características de las RNA las hacen bastante apropiadas para aplicaciones en las que no se dispone a priori de un modelo identificable que pueda ser programado, pero se dispone de un conjunto básico de ejemplos de entrada (previamente clasificados o no). Así mismo, son altamente robustas tanto al ruido como a la disfunción de elementos concretos y son fácilmente paralelizables. Esto incluye problemas de clasificación y reconocimiento de patrones de voz, imágenes, señales y otros. Así mismo se han utilizado para encontrar patrones de fraude económico, hacer predicciones en el mercado financiero, hacer predicciones de tiempo atmosférico y otras aplicaciones. (Allende & Rodrigo, 2003)

1.5.2 Redes Semánticas.

Las redes semánticas, se consideran en nuestros días como una acumulación de conocimientos, son útiles para representar estructuras físicas o dependencias causales, combinando en un modelo integrado y a través del mismo formalismo, los datos y el conocimiento acerca de estos datos, lo que permite la recuperación de informaciones complejas.

Las redes semánticas, constituyen un medio flexible de representación del conocimiento, de cualquier tipo de objeto, atributo, concepto y los enlaces (Lezcano Brito, 1995).

Esta forma FRC pone énfasis en la conectividad del conocimiento análogamente a la construcción de un diccionario. Cada palabra (concepto) es definida en términos de otra, creando una estructura enlazada que interrelaciona virtualmente cualquier cosa. (García Valdivia, Bello Pérez, Gálvez Lío, Lezcano Brito, & Reynoso Lobato, 2000)

Las redes semánticas han sido propuestas como un mecanismo para simular algunas de las características asociativas de la memoria humana. Muchos investigadores en inteligencia artificial consideran que las redes semánticas son el formalismo más apropiado para representar el conocimiento humano (Winton, 2005).

1.5.3 Algoritmos Genéticos.

En los años 1970, de la mano de John Henry Holland, surgió una de las líneas más prometedoras de la inteligencia artificial, la de los algoritmos genéticos (AG). Son llamados así porque se inspiran en la evolución biológica y su base genético-molecular. (Coello Coello, Introducción a los Algoritmos Genéticos, 1997).

Básicamente estos algoritmos plantean el problema de la optimización como un proceso iterativo de búsqueda de la mejor solución dentro del espacio de posibles respuestas al problema. Comienza con un conjunto aleatorio de soluciones tentativas, que constituye la *población* inicial. Combinando las mejores respuestas de ese conjunto se crea un nuevo conjunto que reemplaza al anterior, y que es la generación siguiente; así repetidamente. Con cada nueva iteración la población se va refinando más, consiguiendo mejores respuestas al problema, convergiendo finalmente la mayoría de los individuos en un entorno de la solución óptima. Para aplicar un AG a un problema se necesita identificar el espacio de posibles soluciones, de respuestas al problema, y una codificación de los puntos de ese espacio en un determinado formato. El AG se puede ver como un algoritmo de búsqueda dentro del espacio de las posibles soluciones al problema planteado (Cañas Plaza, 1997).

1.5.4 Sistemas Basados en Frames.

Los sistemas basados en frames constituyen esencialmente sistemas contestadores a preguntas; ellos pueden ser considerados SBC rudimentarios. La verdadera capacidad de los sistemas basados en frames se obtiene al integrarlo con otros modelos. Por ejemplo, los frames son la FRC preferido para los sistemas basados en modelos y los sistemas basados en casos. De este modo se aprovechan sus ventajas y se minimizan sus desventajas (Gálvez Lio, 1998).

Muchos sistemas combinan frames y reglas de producción, ellos usan los frames para representar el conocimiento estructurado y las reglas para razonar. Los frames también pueden ser usados para implementar la representación de los Sistemas Basados en Reglas

de Producción (SBR), esto se hace implementando cada regla como frames, donde la premisa, acción y otros elementos de la regla se representan como slots. De esta forma las reglas de producción son agrupadas en clases. (García Valdivia, Bello Pérez, Gálvez Lío, Lezcano Brito, & Reynoso Lobato, 2000)

1.5 .5 Sistemas Basados en Reglas de Producción.

Los Sistemas Basados en Reglas de Producción (SBR) son SBC en los que la forma de representación del conocimiento usado son las reglas de producción y como método de inferencia utiliza la regla de Modus Ponens. (Lezcano Brito, 1995).

Los SBR son llamados frecuentemente sistemas de producción una idea de 1943 expresada por E. L. Post.

El proceso de solución de problemas en un SBR es crear una cadena de inferencias que constituye un camino entre la definición del problema y su solución. Esta cadena de inferencias puede construirse por dos vías (direcciones de búsqueda):

1. Comenzar con todos los datos conocidos y progresar hacia la conclusión.
2. Seleccionar una conclusión posible y tratar de probar su validez buscando evidencias que la soporten (García Valdivia, Bello Pérez, Gálvez Lío, Lezcano Brito, & Reynoso Lobato, 2000).

Otra de las facilidades que el SBR ofrece es la manipulación de incertidumbre. La necesidad de considerar la incertidumbre se origina en dos hechos:

1. Imprecisiones en los datos.
2. Falta de seguridad en las reglas.

Como los datos que describen el problema a resolver y las reglas pueden ser no totalmente ciertos durante el proceso de razonamiento es necesario que se calcule, para cada nuevo valor inferido, su certidumbre; al terminar el proceso de inferencia la respuesta debe tener asociado su grado de certidumbre (Lezcano Brito, 1995).

El Sistema Basado en Reglas debe ser capaz de ofrecer explicaciones al usuario cuando este se la pida. (Gálvez Lio, 1998).

La función básica de un SBR, es producir resultados. La salida primaria puede ser la solución a un problema, la respuesta a una pregunta o un análisis de algún dato, aunque pueden producirse salidas secundarias en dependencia de los propósitos del sistema. (García Valdivia, Bello Pérez, Gálvez Lío, Lezcano Brito, & Reynoso Lobato, 2000)

Los SBR son los más conocidos de los SBC, sistemas basados en el conocimiento clásicos como Dendral, MYCIN y R1 también conocido como XCON fueron SBR (Gálvez Lio, 1998).

1.6 Sistemas Expertos.

Un Sistema Experto debe ser capaz de procesar y memorizar información, aprender y razonar en situaciones deterministas e inciertas, comunicar con los hombres y/u otros sistemas expertos, tomar decisiones apropiadas, y explicar por qué se han tomado tales decisiones. Se puede pensar también en un sistema experto como un *consultor* que puede suministrar ayuda a (o en algunos casos sustituir completamente) los expertos humanos con un grado razonable de fiabilidad. (Lezcano Brito, 1995).

En la literatura existente se pueden encontrar muchas definiciones de Sistemas Expertos, por ejemplo la que da Stevens (1984):

“Los sistemas expertos son máquinas que piensan y razonan como un experto lo haría en una cierta especialidad o campo. Por ejemplo, un sistema experto en diagnóstico médico requeriría como datos los síntomas del paciente, los resultados de análisis clínicos y otros hechos relevantes, y, utilizando éstos, buscaría en una base de datos la información necesaria para poder identificar la correspondiente enfermedad. [. . .] Un Sistema Experto de verdad, no sólo realiza las funciones tradicionales de manejar grandes cantidades de datos, sino que también manipula esos datos de forma tal que el resultado sea inteligente y tenga significado para responder a preguntas incluso no completamente especificadas.” (p. 40)

Aunque la anterior es todavía una definición razonable de un sistema experto, han surgido desde entonces otras definiciones, debido al rápido desarrollo de la tecnología. El sentido de estas definiciones puede resumirse como sigue :(Castillo, Gutiérrez, & Hadi, 1999)

“Un sistema experto puede definirse como un sistema informático (hardware y software) que simula a los expertos humanos en un área de especialización dada” (p. 8)

Durante la última década se han desarrollado muy rápidamente numerosas aplicaciones de sistemas expertos a muchos campos, pues existen varias razones para utilizarlos ya que con la ayuda de estos, personal con poca experiencia puede resolver problemas que requieren un conocimiento de experto, también el conocimiento de varios expertos humanos puede combinarse, los sistemas expertos pueden responder a preguntas y resolver problemas mucho más rápidamente que un experto humano. (Castillo, Gutiérrez, & Hadi, 1999)

Una característica decisiva de los Sistemas Expertos es la separación entre conocimiento (reglas, hechos) por un lado y su procesamiento por el otro. A ello se añade una interface de usuario y un componente explicativo (Gálvez Lio, 1998).

1.6.1 Arquitectura básica de los sistemas expertos.

Los Sistemas Expertos están compuestos por la siguiente arquitectura (Bello, Gálvez, García, & Lezcano, 1995):

La base de conocimientos. Es la parte del sistema experto que contiene el conocimiento sobre el dominio, hay que obtener el conocimiento del experto y codificarlo en la base de conocimientos. Una forma clásica de representar el conocimiento en un sistema experto son las reglas de producción.

La base de hechos (Memoria de trabajo). Contiene los hechos sobre un problema que se han descubierto durante una consulta. Durante una consulta con el sistema experto, el usuario introduce la información del problema actual en la base de hechos. El sistema empareja esta información con el conocimiento disponible en la base de conocimientos para deducir nuevos hechos.

El motor de inferencia. El sistema experto modela el proceso de razonamiento humano con un módulo conocido como el motor de inferencia. Dicho motor de inferencia trabaja con la información contenida en la base de conocimientos y la base de hechos para deducir nuevos hechos. Contrasta los hechos particulares de la base de hechos con el conocimiento contenido en la base de conocimientos para obtener conclusiones acerca del problema.

El subsistema de explicación. Una característica de los sistemas expertos es su habilidad para explicar su razonamiento. Usando el módulo del subsistema de explicación, un sistema experto puede proporcionar una explicación al usuario de por qué está haciendo una

pregunta y cómo ha llegado a una conclusión. Este módulo proporciona beneficios tanto al diseñador del sistema como al usuario. El diseñador puede usarlo para detectar errores y el usuario se beneficia de la transparencia del sistema.

La interfaz de usuario. La interacción entre un sistema experto y un usuario se realiza en lenguaje natural. También es altamente interactiva y sigue el patrón de la conversación entre seres humanos. Para conducir este proceso de manera aceptable para el usuario es especialmente importante el diseño del interfaz de usuario. Un requerimiento básico del interfaz es la habilidad de hacer preguntas. Para obtener información fiable del usuario hay que poner especial cuidado en el diseño de las cuestiones. Esto puede requerir diseñar el interfaz usando menús o gráficos.

1.6.2 El equipo de desarrollo de los Sistemas Expertos

Las personas que componen un grupo o un equipo, como en todos los ámbitos deben cumplir unas características y cada uno de ellos dentro del equipo desarrolla un papel distinto.

A continuación de detalla cada componente del equipo dentro del desarrollo y cuál es la función de cada uno: (Coello Coello, La Computación Evolutiva en el Contexto de la Inteligencia Artificial, 2005)

El experto: La función de este es la de poner sus conocimientos especializados a disposición del Sistema Experto.

El ingeniero del conocimiento: Este plantea las preguntas al experto, estructura sus conocimientos y los implementa en la base de conocimientos.

El usuario: Este aporta sus deseos y sus ideas, determinado especialmente el escenario en el que debe aplicarse el Sistema Experto.

En el desarrollo del Sistema Experto, el ingeniero del conocimiento y el experto trabajan muy unidos. El primer paso consiste en elaborar los problemas que deben ser resueltos por el sistema. Precisamente en la primera fase de un proyecto es de vital importancia determinar correctamente el ámbito estrechamente delimitado de trabajo. Aquí se incluye ya el usuario

posterior, o un representante del grupo de usuarios. Para la aceptación, y consecuencia para el éxito, es de vital y de suma importancia tener en cuenta los deseos y las ideas del usuario.

1.6.3 Aspectos esenciales para el diseño de un Sistema Experto.

Para el diseño de un sistema experto se deben tener en cuenta aspectos esenciales, por ejemplo: (Lezcano Brito, 1995)

La base de conocimiento debe ser consecuente, es decir, se debe construir a partir de información precisa, obtenida de la práctica, experiencia y opiniones del experto y que haya sido probada en el medio adecuado con resultados exitosos. Los procedimientos de la base de conocimiento deben responder a una metodología consistente y correcta, partiendo de la información obtenida en base a la regla anterior.

Se debe proporcionar una estructura adecuada para la socialización con el hombre. Los mensajes y preguntas se deben presentar en forma semejante a las que se usan en el medio real, puesto que las preguntas que carecen de claridad pueden dar lugar a ambigüedades y disgustos por parte del usuario, provocando además ciertas limitaciones al sistema.

Se debe proporcionar un tiempo de respuesta adecuado, de forma que se aumente la productividad del sistema. Esto implica que se eviten, por todos los medios, iteraciones y lazos innecesarios, para mantener de esta forma el tiempo de espera al mínimo.

Un sistema experto es tan bueno como lo sea su base de conocimiento, por tanto en su construcción se deben emplear todos los esfuerzos y recursos necesarios, así como todo el tiempo que se disponga. Se deben realizar tantas iteraciones como sean precisas hasta que el sistema alcance su estabilidad.

1.7 Los Sistemas Expertos y sus aplicaciones.

Los Sistemas Expertos (SE) tienen muchas aplicaciones (Castillo, Gutiérrez, & Hadi, 1999). Por ejemplo no hace mucho, para hacer una transacción bancaria, tal como depositar o sacar dinero de una cuenta, se tenía que visitar el banco en horas de oficina. Hoy en día, esas y otras muchas transacciones pueden realizarse en cualquier momento del día o de la noche usando los cajeros automáticos. Otra aplicación es en el control de tráfico, anteriormente, el flujo de tráfico en las calles de una ciudad se controlaba mediante guardias

de tráfico que controlaban el mismo en las intersecciones. Hoy se utilizan SE que operan automáticamente los semáforos y regulan el flujo del tráfico en las calles de una ciudad y en los ferrocarriles.

En la educación también existen SE, ejemplo de estos es un módulo tutor inteligente como aplicación pedagógica de la ley distributiva, este en un sistema experto que ayuda al proceso de enseñanza y aprendizaje en la aplicación de la ley distributiva a la solución del producto de un número natural por un polinomio aritmético. (Gutiérrez de Piñerez Reyes, 2008)

Dentro del sector textil, la evolución permanente en el diseño es un mercado cada vez más exigente, para ayudar a los nuevos diseños de productos se desarrolló un sistema experto para la toma de decisiones en la etapa inicial del diseño del producto, para enlazar planificaciones, funciones técnicas y comerciales actualmente separadas. Esto se hace partiendo del conocimiento del diseño de un nuevo producto, del conocimiento del proceso actual de fabricación, del conocimiento del funcionamiento actual de la maquinaria y del conocimiento de la flexibilidad de la maquinaria. (Goicoechea Castaño, 2005).

Sin embargo en el campo de la medicina donde estos sistemas inteligentes se han aplicado con notable relevancia, pues pueden ser utilizados para contestar a las siguientes preguntas:

1. Supuesto que un paciente presenta un conjunto de síntomas, ¿cómo se decide qué enfermedad es la que más probablemente tiene el paciente?
2. ¿Cuáles son las relaciones entre un conjunto (normalmente no observable) de enfermedades y un conjunto (observable) de síntomas?
 - En otras palabras, ¿qué modelos pueden utilizarse para describir las relaciones entre los síntomas y las enfermedades?
3. Dado que el conjunto de síntomas conocidos no es suficiente para diagnosticar la enfermedad con cierto grado de certeza, ¿qué información adicional debe ser obtenida (por ejemplo, ¿qué síntomas adicionales deben ser identificados? o ¿qué pruebas médicas deben realizarse?).
4. ¿Cuál es el valor de cada una de estas piezas de información? En otras palabras, ¿cuál es la contribución de cada uno de los síntomas adicionales o pruebas a la toma de decisión?

1.7.1 Los Sistemas Expertos para la Salud.

Para los SE, el campo de la salud es un área importante de desarrollo. Por ejemplo, se registran sistemas para ayudar en el diagnóstico de abdomen agudo, anomalías craneofaciales (Señor García, 2007) y para la realización de diagnóstico parálisis facial con electromiografía: **PARFAC**. (Ruiz, 2006)

PARFAC ayuda a los médicos en la determinación de diagnósticos en el campo de la neurofisiología. Es consultado por el personal de salud, quien primero revisa los datos generales del paciente junto a un determinado diagnóstico. Una vez conocida esta información por el sistema, el Sistema Experto plantea unas hipótesis, que de inmediato serán verificadas para comprobar la exactitud de las premisas de la regla; luego busca las correspondientes de los enunciados en la base de conocimientos. (Ruiz, 2006). Por otra parte científicos onubenses desarrollan un sistema experto para la detección precoz de la retinopatía diabética. El sistema trabaja sobre las imágenes tomadas con retinógrafos digitales y es capaz de detectar el disco óptico y los capilares del ojo. Funcionará como un asistente mecánico que, en base al análisis e interpretación de retinografías tomadas en los centros de salud, suministrará al médico de forma automática indicios claros de si un paciente sufre o no la enfermedad para que valore si debe ser tratado por el especialista (Pérez Martínez, 2009).

Un sistema experto para ayudar en el diagnóstico urodinámico, permitiendo al urólogo decidir el posible diagnóstico de un paciente realizando el menor número de ensayos; se desarrolló en la Universidad de Alicante. El conocimiento de los especialistas, con experiencia en la casuística del problema, ofrece una fuente de conocimientos con la que obtener una completa e interesante base de datos a la que se le pueden aplicar técnicas de inteligencia artificial, lo que permitirá ayudar en la toma de decisiones.

A partir de los nuevos datos obtenidos, el “sistema experto” y el urólogo tendrán más información para concluir el diagnóstico (Universidad de Alicante, 2007).

1.8 El Dengue.

El dengue es una enfermedad infecciosa producida por un virus de genoma ARN, al cual se le reconocen cuatro serotipos (DEN-1, DEN-2, DEN-3 y DEN-4) que son transmitidos por el

Aedes aegypti como principal vector. Se manifiesta clínicamente en dos formas principales: la fiebre del dengue (FD) también llamada dengue clásico y la forma hemorrágica: fiebre hemorrágica del dengue (FHD) a veces con síndrome de choque por dengue (FHD/SCD) (Pacheco Arencibia, 2009).

La primera epidemia de dengue hemorrágico en las Américas se presentó en Cuba (Guzmán Tirado, 1984) en 1981 y fue por DEN-2. En mayo de ese año se comenzaron a notificar algunos enfermos con síndrome febril compatible con el diagnóstico de dengue en el municipio Boyeros de la Ciudad de La Habana. En ese mismo lugar, se comprobó, retrospectivamente, la existencia de pacientes similares durante los meses anteriores. La enfermedad fue confirmada simultáneamente en La Habana, Cienfuegos y Camagüey (PAHO-OMS, 1986). Después se vieron afectadas las restantes provincias.

Por el hecho de que se presenta en forma de epidemias, el dengue tiene gran repercusión económica y social por la afectación laboral, ausentismo escolar y, en general, grandes molestias a la población. A partir de la década de los años 50 y durante más de 25 años la FHD/SCD constituyó un grave problema para los países del Sudeste Asiático y Oceanía, pero durante la década de los años 80, comenzando por Cuba en 1981, esta enfermedad se manifestó en diversos países de América y actualmente se ha extendido a casi toda la región (Suárez García, 2009).

1.8.1 Clasificación temprana de Dengue Severo.

De las enfermedades virales transmitidas por artrópodos, el Dengue es la de mayor importancia actualmente y constituye una prioridad de salud pública en los países tropicales y subtropicales.

Desde el punto de vista patogénico todavía quedan algunas incógnitas con relación al Dengue, la mayoría de los pacientes desarrollan formas benignas de la enfermedad, pero otros llegan hasta el choque. Por lo pronto es muy importante la búsqueda de elementos clínicos o de laboratorio que ayuden a identificar de forma precoz las posibles complicaciones. Pudiendo ser esta la clave para disminuir la mortalidad (Suárez García, 2009).

La actual definición de caso de Dengue Severo establecida por la Organización Mundial de la Salud (OMS) incluye cuatro criterios: fiebre, algún tipo de sangrado, trombocitopenia y evidencia objetiva de salida de plasma.

El Dengue Severo es clasificado a su vez en cuatro grados. En el grado I el sangramiento solo se evidencia mediante la prueba del torniquete, mientras que en el grado II el sangramiento es espontáneo. Los grados III y IV corresponden al Síntoma de Choque (PAHO, 1994).

1.9 Metodología para los Sistemas Expertos.

La creación de las bases de conocimiento es un complejo y largo período de adquisición de información, llevado a cabo por el ingeniero del conocimiento; quién, partiendo de los juicios y opiniones de diferentes expertos, debe codificar la información obtenida utilizando un lenguaje formal de representación del conocimiento. La culminación exitosa de este trabajo requiere un consenso en cuanto al criterio de cada uno de los expertos, eliminando cualquier discrepancia que pudiera haber surgido en el proceso de refinamiento con respecto a un tema determinado (lo cual ocurre a menudo). Si esto se logra, la base de conocimiento resultante contará con una biblioteca de información mayor que cualquier caudal de información individual.

Al diseñar una base de conocimiento, se está construyendo algo como una teoría sobre un determinado dominio o que al menos aspira a llegar a serlo, en este sentido deben prevalecer en su diseño los aspectos conceptuales. La información debe estar organizada en forma armónica y debe ser, ante todo, lo más exacta posible. Sin embargo, el conocimiento de los expertos es a menudo incompleto e inconsistente, por lo que el proceso puede tornarse bastante engorroso y en algunos casos, el resultado pudiera no ser lo suficientemente exacto. (Lezcano Brito, 1995).

1.9.1 Método clásico para el diseño de un Sistema Experto.

Los métodos de construcción de sistemas expertos parten, en primer lugar, de la identificación de los elementos del dominio y los conceptos descriptivos de sus propiedades, definiendo además las relaciones que se establecen entre ellos, en base a las cuales se organiza el conocimiento y posteriormente se pasa a su formalización. Desde el punto de vista cognoscitivo, la formalización del conocimiento va de la contemplación viva al

pensamiento abstracto y de ahí nuevamente a la realidad. En este camino se parte del plano real (observación de la realidad), se pasa por el plano conceptual (creación del modelo para partes de esa realidad) y se llega al plano formal (elaboración de la teoría). (Lezcano Brito, 1995)

Existe un método clásico general, que permite la separación de este proceso en varias etapas. El primer paso a seguir cuando se habla de la construcción de un sistema experto es el estudio del dominio, con vista a determinar si es realmente propicia la construcción del sistema experto o no. Después de hecho este análisis, si se determina que es posible la construcción del sistema experto, se pasa a las diferentes etapas como son: la identificación del problema, se establecen los conceptos, se formalizan los conceptos claves y subproblemas y se pasa a la implementación. (Gutiérrez de Piñerez Reyes, 2008)

1.10 El Proceso Unificado de desarrollo de software.

En primer lugar, el Proceso Unificado es un proceso de desarrollo de software. Un proceso de desarrollo de software es el conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema software. Sin embargo, el Proceso Unificado es más que un simple proceso; es un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas de software, para diferentes áreas de aplicación, diferentes tipos de organizaciones, diferentes niveles de aptitud y diferentes tamaños de proyectos. El Proceso Unificado está basado en componentes, lo cual quiere decir que el sistema software en construcción está formado por componentes software interconectados a través de interfaces bien definidas.

El Proceso Unificado utiliza el Lenguaje Unificado de Modelado (Unified Modeling Language, UML) para preparar todos los esquemas de un sistema software. De hecho, UML es una parte esencial del Proceso Unificado. No obstante, los verdaderos aspectos definitorios del Proceso Unificado se resumen en dos partes claves; la definición de los casos de usos y la definición de la arquitectura. (Jacobson, Booch, & Rumbauch, 2000)

1.11 Tecnologías y Herramientas

1.11.1 Prolog.

Prolog es un lenguaje de programación para manipular objetos y las relaciones entre ellos y se clasifica como un lenguaje de programación lógica debido a que se basa en la prueba de teoremas, a partir de una base de datos interna formada por reglas escritas en la forma de cláusulas de Horn, donde se aplica el principio de resolución y de unificación. Se dice que Prolog es adecuado para buscar soluciones de problemas que no sabemos resolver (Lezcano Brito, 1995).

Por tal razón son muchas las compañías de software que han creado sus propias versiones. La diferencia entre una versión y otra es mínima ya que la sintaxis y la semántica, generalmente, es la misma, la variación que más resalta es el cambio de plataforma para el cual fueron desarrollados.

Según Ramos Ríos, los más usados y conocidos son: (Ramos Ríos, 2010)

ADA PD Prolog es un intérprete antiguo (1986) y lento para MS-DOS de Automata Design Associates.

Arity/Prolog32 es un entorno completo de programación en Prolog para Windows de 32 bits que incluye un verdadero compilador, editor, depurador, intérprete y ayuda.

B-Prolog es un sistema CLP (Constraint Logic Programming) que ejecuta programas Prolog. Hay versiones para Windows y varios Unix, incluyendo Linux. Se distribuye también el código fuente.

Ciao es un entorno de programación multi-paradigma de acuerdo con el estándar ISO, permitiendo tanto restricciones como extensiones al lenguaje. Se distribuye el código fuente y binario para Windows.

CU-Prolog es un lenguaje CLP experimental adecuado para el procesamiento de lenguajes naturales. Hay versiones Unix, MS-DOS (djcup) y Macintosh (MacCup).

Existen otras versiones de Prolog como DGKS, ESL Prolog-2, GNU Prolog, JIProlog, JProlog, Kernel Prolog, K-Prolog, LPA PROLOG Professional, MACPROLOG, Micro Prolog,

NU-Prolog, Open Prolog, PROLOG1, PROLOG QUINTUS, Qu-Prolog (de la Universidad de Queensland), Reform Prolog, Sicstus Prolog, Strawberry Prolog, Turbo Prolog, Visual Prolog, XSB, YAP, SWI-Prolog, Amzi! Prolog y otros.

1.11.2 SWI-Prolog.

SWI-Prolog trabaja en un entorno de Software Libre, licenciado bajo la Lesser Gnu Public License y posee un conjunto de herramientas de gráficos XPCE, su desarrollo se inició en 1987 y ha sido impulsada por las necesidades de aplicaciones en el mundo real. En estos días SWI-Prolog se utiliza en la investigación y la educación, así como para aplicaciones comerciales. (Ramos Ríos, 2010)

1.11.3 AMZI! Prolog.

AMZI! Prolog es una de las implementaciones de Prolog más fácil y amigable que se acopla muy bien con Windows. Una de las facilidades que esta versión ofrece es que es ampliamente compatible con muchos lenguajes de programación, pues ofrece interfaces de comunicación con Java, Visual Basic, C++, MS Office, Delphi y otros.

Dentro de las características más importantes de AMZI-Prolog están: compilación rápida; el tamaño del programa, la longitud de los átomos, término-arity o valores son ilimitados. Posee un rico conjunto de predicados incorporados que hacen posible el desarrollo de aplicaciones robustas. Amzi! Prolog posee una interfaz simple para generar aplicaciones de consola. Pero por supuesto que esto no es suficiente cuando es necesario utilizar Prolog para aplicaciones profesionales que requieren una interfaz más amigable y accesible para todo tipo de usuarios; por lo que ofrece una interfaz para el lenguaje Object Pascal. (Tito, Aliendre, & Juaniquina, 2006)

1.11.4 Delphi.

Delphi es un entorno de desarrollo de software diseñado para la programación de propósito general con énfasis en la programación visual. En Delphi se utiliza como lenguaje de programación una versión moderna de Pascal llamada Object Pascal. En sus diferentes variantes, permite producir archivos ejecutables para Windows, GNU/Linux y la plataforma .NET. Un uso habitual de Delphi (aunque no el único) es el desarrollo de aplicaciones visuales.

Como desarrollo visual, la programación en Borland Delphi consiste en diseñar los formularios que componen al programa colocando todos sus controles (botones, etiquetas, campos de texto, etc.) en las posiciones deseadas, normalmente usando un ratón. Luego se asocia código a los eventos de dichos controles y también se pueden crear módulos de datos, que regularmente contienen los componentes de acceso a datos y las reglas de negocio de una aplicación (Bucknall, 2006).

1.12 Método para la estimación del esfuerzo llamado COCOMO II.

El SEI (del inglés, Software Engineering Institute) propone desde hace algunos años un método para la estimación del esfuerzo llamado COCOMO II. Este método está basado en ecuaciones matemáticas que permiten calcular el esfuerzo a partir de ciertas métricas de tamaño estimado, como el Análisis de Puntos de Función y las líneas de código fuente (en inglés SLOC, Source Line Of Code). Estas ecuaciones se encuentran ponderadas por ciertos factores de costo (cost drivers) que influyen en el esfuerzo requerido para el desarrollo del software.

La estimación por COCOMO II (con Puntos de Función sin ajustar como entrada), resulta muy útil para estimar un proyecto en forma global. Utilizando la herramienta del SEI, se puede refinar la estimación a medida que se va adquiriendo más información sobre el proyecto. Esta herramienta no está calibrada para proyectos menores a 2000 líneas de código, con lo cual no es aplicable a proyectos muy pequeños.

El método de estimación COCOMO II está basado dos modelos: uno aplicable al comienzo de los proyectos (Diseño preliminar, en inglés Early Design) y otro aplicable luego del establecimiento de la arquitectura del sistema (Post arquitectura, en inglés Post Architecture). (Peralta, 2005)

1.13 Conclusiones.

En este capítulo se recogen aspectos de la Inteligencia Artificial y los Sistemas Expertos, se describe la clasificación temprana de Dengue Severo a través de un sistema experto, se relaciona las razones para la selección de la metodología a seguir en la documentación de Sistemas Experto, precisa la conveniencia de la elección del lenguaje de programación Prolog, para desarrollar este tipo de sistema, así como las reglas de producción como forma de representar el conocimiento.

Se hace referencia a la utilidad de SWI-Prolog y AMZI! Prolog para la confección de la base de conocimiento y de Borland Delphi para modelar el análisis y diseño.

*El hombre crece
con el trabajo
que sale de sus manos.*

José Martí

Capítulo 2: Análisis, diseño e implementación de la propuesta de solución.

En el desarrollo de esta investigación se utilizó la metodología descrita por el Dr. Mateo Lezcano Brito en su libro Prolog y los Sistemas Expertos, donde primeramente se identifican los elementos del dominio y los conceptos descriptivos de sus propiedades, se define las relaciones que se establecen entre ellos, en base a las cuales se organiza el conocimiento y posteriormente se pasa a su formalización e implementación.

De forma complementaria se utilizó el Lenguaje Unificado del Modelo con el objetivo de establecer los casos de uso del sistema.

Un estudio del dominio con vista a determinar si es realmente propicia la construcción del SE o no, basados en los siguientes criterios:

- inexistencia de una solución algorítmica para el problema,
- complejidad de las tareas a resolver,
- interés práctico y beneficios que se observaran,
- posibilidad de pérdida del conocimiento humano,

arrojó que es posible la construcción del Sistema Experto.

Seguidamente se describen las etapas para el diseño del Sistema Experto.

2.1 Identificación

En esta etapa se determinó, básicamente, cuál es el problema que se quiere resolver y sus características, así como quienes van a participar en el levantamiento de la base de conocimiento y el papel de cada cual en ese proceso. Este es más bien un período de

familiarización, donde tiene lugar la relación inicial entre los experto y el ingeniero del conocimiento. Es un encuentro donde los experto hacen que el ingeniero de conocimiento se lleve una idea general del dominio que se quiere modelar, haciendo una caracterización informal del problema y mostrando algunas descripciones de problemas típicos y los posibles pasos para su solución.

Los participantes en el levantamiento de la base de fueron:

- Ernesto Arencibia Obregón encargado del diagnóstico y la vigilancia de enfermedades infecciosas.
- Beatriz Vega Riverón y Lizet Sánchez Valdés quienes aportaron las técnicas de clasificación y procesamiento de las bases de datos. Sus resultados sirvieron para establecer las reglas del sistema.
- Dr. Daniel González Rubio, Dr. Osvaldo Castro, Dr. Orlando Javier Suárez García y Dr. Betsy Pacheco Arencibia participaron como expertos de Dengue, ellos facilitaron la información clínica útil al sistema y asesoraron en este sentido.

De forma conjunta, se determinaron los elementos del dominio con sus elementos descriptivos, los que se relacionan seguidamente:

Elementos del dominio:

- Signos de alarmas.

Conceptos descriptivos: (Martínez Viera, Blanco Sánchez, & González Ávila, 1989)

- Fiebre: elevación sostenida de la temperatura corporal.
- Cefalea: dolor de la cabeza.
- Astenia marcada: decaimiento.
- Vómitos reiterados: expulsión frecuente por la boca del contenido de las vías digestivas.

- Anorexia: pérdida del apetito.
- Rash: sarpullido o erupción cutánea.
- Antecedentes patológicos.

Conceptos descriptivos :(Tkachuk & Hirschmann, 2007)

- Sicklemia: anemia hemolítica crónica de carácter hereditario producida por una sustitución de aminoácido en la cadena polipeptídica de la hemoglobina.
- Asma Bronquial: enfermedad respiratoria en la que el espasmo y la constricción de los bronquios y la inflamación de su mucosa limita el paso del aire, con la consiguiente dificultad respiratoria. Este estrechamiento de las vías respiratorias es típicamente temporal y reversible, pero en ataques severos de asma puede provocar incluso la muerte.
- Diabetes Mellitus: enfermedad producida por una alteración del metabolismo de los carbohidratos en la que aparece una cantidad excesiva de azúcar en la sangre y a veces en la orina.
- Hipertensión Arterial (HTA): tensión arterial alta.
- Alergia: enfermedad debida a una reacción exagerada del sistema inmunológico (hipersensibilidad) frente a determinadas sustancias que son inocuas para la mayoría de las personas.
- Género: Evaluado según características fenotípicas (M, y otros, 2006).
- Hallazgos de laboratorio.

Conceptos descriptivos :(Tkachuk & Hirschmann, 2007)

- Plaquetas: también denominada trombocito, fragmento citoplasmático de un megacariocito (la célula de mayor tamaño presente en la médula ósea), que se encuentra en la sangre periférica, donde interviene en el proceso de coagulación de la sangre.

- Hemoglobina: pigmento especial que predomina en la sangre cuya función es el transporte de oxígeno.
 - Leucopenia: Número de leucocitos en la sangre inferior al normal. Valor del leucograma menor de 5×10^9 .
 - Trombocitopenia: descenso patológico del número de plaquetas circulantes. Valor del conteo de plaquetas menor o igual que 150×10^9 .
 - Citolisis hepática: ALT, AST o ambas con valor mayor de 40 UI/L
 - Hemoconcentración: Aumento de 20% del hematocrito por encima de un valor basal
 - Proteinuria: Presencia de proteínas en el sedimento urinario.
- Hallazgos de ultrasonido.

Conceptos descriptivos: (Tkachuk & Hirschmann, 2007)

- Derrame pleural: presencia de líquido en la cavidad pleural motivado por múltiples enfermedades que de acuerdo a la características de ese líquido puede ser exudado o traxudado.
 - Derrame pericárdico: presencia de líquido en la cavidad del pericárdico.
 - EPV: presencia de epato y o esplenomegaria.
- Manifestaciones clínicas.

Conceptos descriptivos: (Tkachuk & Hirschmann, 2007)

- Manifestación hemorrágica: lesiones en la piel, sangramiento por la mucosa u órganos internos.
- Diarreas: alteración del ritmo intestinal que se acompaña de aumento del número de deposiciones o de alteración de las características normales de una deposición (líquidas o semilíquidas). Se trata en general de un proceso

transitorio cuya causa más frecuente son las infecciones víricas o bacterianas. También puede ser producida por la toma de sustancias tóxicas que lesionan el intestino, por situaciones de tensión emocional o por estados nerviosos alterados.

- Prurito: sensación procedente de la piel que produce el deseo de rascado. También se llaman así las alteraciones cutáneas caracterizadas por esa sensación. El picor, o escozor, se produce por la irritación de las terminales nerviosas sensitivas del dolor y el tacto localizadas en la piel, y puede acompañarse de inflamación o erupciones cutáneas.
- Mareos: sensación que afecta a muchas personas en diferentes grados al inclinarse, al realizar giros o movimientos de vaivén, o al bostezar.

El problema inicial se puede dividir en posibles subproblemas, estos son:

- Clasificar los posibles casos críticos de Dengue Severo.
- Clasificar los posibles casos de alarmas de Dengue Severo.
- Clasificar los posibles casos de alertas de Dengue Severo.

Los objetivos relevantes e irrelevantes de la investigación:

- Relevantes:
 - Lograr clasificar los posibles casos críticos.
- Irrelevantes:
 - Lograr clasificar los posibles casos de alarmas.
 - Lograr clasificar los posibles casos de alertas.

Los recursos con que se cuenta para la obtención de información:

- Expertos en Dengue del IPK de Ciudad de la Habana.

- Libro: Dengue y Dengue Hemorrágico por el Dr. Eric Martínez Torres Doctor en Ciencias Médicas Profesor Titular de Pediatría, Miembro de la Academia de Ciencias de Cuba, Miembro de la Asociación Panamericana de Infectología (API) y de la Sociedad Latinoamericana de Infectología Pediátrica (SLIPE), Director Nacional de Investigación y Desarrollo del Ministerio de Salud Pública de Cuba.
- Tesis: Factores de riesgo asociados a la Fiebre hemorrágica del Dengue. Instituto “Pedro Kourí”, 2006. Por la Dra. Betsy Pacheco Arencibia MD, el cual abarca el tema de los factores de riesgo asociados a la fiebre hemorrágica del Dengue (FHD) en los pacientes hospitalizados en el IPK durante la epidemia presentada en los meses de mayo a noviembre del 2006, en la provincia Ciudad de La Habana.
- Tesis: Caracterización y valor pronóstico de los signos de alarma del Dengue. Instituto “Pedro Kourí”. 2006. Por el Dr. Orlando Javier Suárez García. MD, con el objetivo de caracterizar y estimar el valor pronóstico de los signos de alarma del Dengue en pacientes ingresados en el IPK durante la epidemia de mayo a noviembre del año 2006.
- Tesis del Dr. Daniel González Rubio dirigida a incrementar el conocimiento en aspectos clínicos de la FHD/SCD específicamente por el serotipo 3 del virus.

El alcance que va a tener, en un inicio, el problema y las diferentes submetas trazadas:

- Alcance:
 - La clasificación temprana de los posibles casos críticos de Dengue Severo.
- Submetas trazadas:
 - Estudiar toda la bibliografía disponible.
 - Confeccionar la base de conocimiento.
 - Confeccionar la interfaz visual.

Una primera versión del Sistema Experto se realizó en SWI-Prolog, luego se desarrolló una segunda versión en AMZI! Prolog. Para desarrollar la interface visual se utilizó Borland Delphi 7 por ser un entorno de desarrollo rápido de aplicaciones y posee facilidades para interactuar con AMZI! Prolog.

2.2 Establecer conceptos.

Se definieron los conceptos para la representación del conocimiento. En esta etapa los expertos y el ingeniero del conocimiento determinaron los aspectos claves del problema, las relaciones entre ellos y sus características con el objetivo de llevar a cabo la descripción del proceso de solución del problema.

Se establecieron una serie de elementos esenciales, tales como:

- Los tipos de datos disponibles son exactos e incompletos (hasta el momento).
- No establecer factor de certidumbre, inicialmente.
- Los datos de salida constituyen alarmas y alertas para posibles casos de pacientes con Dengue Severo.
- El conocimiento se organizó atendiendo a los elementos de dominio y los conceptos descriptivos de estos como ya se había planteado anteriormente, y en base al elemento del dominio que el usuario desee consultar es que se realiza el proceso de razonamiento. Por ejemplo, si el usuario decide consultar los antecedentes patológicos como elemento del dominio, el proceso de razonamiento seguirá la siguiente estrategia: Sicklemia, Asma Bronquial, Diabetes Mellitus, Hipertensión Arterial (HTA) o Alergia, donde en cada caso se evaluará la respuesta del usuario en base a los conceptos, dando así, una posible clasificación para el paciente que se esté consultando.

2.3 Formalización.

En esta etapa se formalizaron los conceptos claves y subproblemas que estaban aislados durante la conceptualización. Se diseñaron las estructuras para organizar el conocimiento. Después de un análisis intensivo, de las diferentes formas de representación del conocimiento, se determinó utilizar las reglas de producción en lenguaje Prolog.

2.3.1 Clasificación del conocimiento

La organización del conocimiento quedó dividida en cinco módulos donde cada uno, dispone de un predicado principal que será el responsable de invocar al conocimiento almacenado en esa parte.

- Módulo 1:
 - **Nombre:** Signos de alarma:
 - **Predicado principal:** consulta.
 - El proceso de razonamiento a través del predicado principal:

consulta:- posible_caso.

posible_caso:-signos_generales.

- Módulo 2:
 - **Nombre:** Antecedentes patológicos:
 - **Predicado principal:** consulta.
 - El proceso de razonamiento a través del predicado principal:

consulta:- posible_caso.

posible_caso:- antecedente_Sicklemlia.

posible_caso:- antecedentes_Asma_Bronquial.

posible_caso:- antecedentes_Diabetes_mellitus.

posible_caso:- antecedentes_HTA.

posible_caso:-antecedentes.

- Módulo 3:

- **Nombre:** Hallazgos de laboratorio:
- **Predicado principal:** consulta.
- El proceso de razonamiento a través del predicado principal:

consulta:- posible_caso.

posible_caso:-conteo_plaquetas.

posible_caso:-conteo_plaquetas_creatinina.

posible_caso:-conteo_plaquetas_creatinina_hemoglobina.

posible_caso:-trombocitopenia.

posible_caso:-leucopenia.

posible_caso:-citólisis_hepática.

posible_caso:-hematocrito.

posible_caso:-hemoconcentración.

posible_caso:-proteinuria.

- Módulo 4:

- **Nombre:** Hallazgos de ultrasonido:
- **Predicado principal:** consulta.
- El proceso de razonamiento a través del predicado principal:

consulta:- posible_caso_critico.

posible_caso_critico:-derrame_pleural.

posible_caso_critico:-derrame_pericardico.

posible_caso_critico:-epv.

posible_caso_critico:-edema_perivesicular.

- Módulo 5:
 - **Nombre:** Manifestaciones clínicas:
 - **Predicado principal:** consulta.
 - El proceso de razonamiento a través del predicado principal:

consulta:- posible_caso.

posible_caso_critico:-manifestacion_hemorragica.

posible_caso_critico:-manifestacion_nauseas.

posible_caso_critico:-manifestacion_diarreas.

posible_caso_critico:-manifestacion_prurito.

posible_caso_critico:-manifestacion_mareos.

2.4 Implementación.

En esta fase se combinó y organizó el conocimiento formalizado para hacerlo compatible con las características del flujo de información del problema. Se llevó a cabo la formalización de las reglas que abarcan todo el conocimiento, obteniéndose un primer prototipo, que muestra sobre un esquema elemental la forma en que opera el sistema. El objetivo fundamental de este primer prototipo es obtener una solución inmediata del problema.

La realización de pruebas al primer prototipo y el análisis de las críticas que se hicieron a su funcionamiento sirvieron como punto de partida para la formalización de una versión más avanzada de la base de conocimiento.

2.4.1 Formalización de las reglas.

Algunas reglas que abarcan el conocimiento quedaron formalizadas de la siguiente manera:

antecedente_Sicklemlia:- write('¿El paciente presenta Sicklemlia como antecedente?'),nl,
read(Antecedente_Sicklemlia),
Antecedente_Sicklemlia='S',
write('Posible caso crítico').

conteo_plaquetas_creatinina:- write('¿Cuál es el valor de la creatinina?'),nl,
read(Creatinina),
asserta(creatinina(Creatinina)),
plaquetas(Plaquetas),
Plaquetas > 65.5, Plaquetas < 123.5, (Creatinina > 83;
Creatinina = 83),
write('Posible caso crítico').

conteo_plaquetas_creatinina_hemoglobina:- write('¿Cuál es el valor de la hemoglobina?'),nl,
read(Hemoglobina),
plaquetas(Plaquetas),
creatinina(Creatinina),
Plaquetas > 65.5, Plaquetas < 123.5,Creatinina <
83,Hemoglobina < 148.5,
write('Posible caso crítico').

manifestacion_nauseas:- write('¿El paciente manifiesta náuseas ¿'),nl,
read(Nauseas),
Nauseas='S',
write('Caso de alarma').

signos_generales:- write('¿El paciente presenta fiebre elevada, constante y diaria?'),nl,
read(Fiebre),
Fiebre='S',
write('¿El paciente presenta cefalea intensa, constante y opresiva?'),nl,
read(Cefalea),
Cefalea='S',
write('¿El paciente presenta dolor retrocular, al movimiento ocular e
intenso?'),nl,

```
read(Dolor_ocular),
Dolor_ocular='S',
write('Posible caso crítico').
signos_generales:- write('¿El paciente presenta gingivorragia?'),nl,
read(Gingivorragia),
Gingivorragia='S',
write('Caso de alarma ').
```

Las reglas anteriormente descritas fueron implementadas en SWI-Prolog, en una versión inicial, luego se hizo una segunda versión en AMZI! Prolog con la intención de lograr una interfaz visual más amigable mediante la interacción con Delphi.

En la segunda versión cada módulo de la base de conocimiento se divide en tres partes, un bloque de preguntas, un bloque de reglas y un bloque de explicación; tal como se muestra en el ejemplo siguiente:

/*Bloque de Preguntas*/

```
pregunta_plaquetas:- write('¿Cuál es el valor del conteo de las plaquetas ?').
pregunta_plaquetas_creatinina:- write('¿Cuál es el valor de la creatinina?').
pregunta_plaquetas_creatinina_hemoglobina:- write('¿Cuál es el valor de la hemoglobina?').
pregunta_citolisis_hepatica_ALT_AST:-write('¿Cuál es el valor de ALT y de AST?').
pregunta_leucopenia:- write('¿Cuál es el valor del leucograma?').
pregunta_hematocrito:- write('¿Cuál es el valor del Hematocrito?').
pregunta_proteinuria:- write('¿El paciente presenta proteinuria en el sedimento urinario?').
```

/*Bloque de reglas*/

```
posible_caso:-conteo_plaquetas(Plaquetas),
(Plaquetas < 65.5),
```

diagnostico.

posible_caso:-conteo_plaquetas(Plaquetas),
(Plaquetas=100 ; Plaquetas<100),

diagnostico.

posible_caso:-conteo_plaquetas(Plaquetas),
conteo_plaquetas_creatinina(Creatinina),
(Plaquetas > 65.5), (Plaquetas < 123.5), (Creatinina > 83; Creatinina = 83),

diagnostico.

posible_caso:-conteo_plaquetas(Plaquetas),
conteo_plaquetas_creatinina(Creatinina),
conteo_plaquetas_creatinina_hemoglobina(Hemoglobina),
(Plaquetas > 65.5), (Plaquetas < 123.5),(Creatinina < 83),(Hemoglobina < 148.5),

diagnostico.

posible_caso:-citolisis_hepatica_ALT(ALT),
citolisis_hepatica_AST(AST),
(((ALT > 40, AST > 40), diagnostico);
((ALT > 40; AST >40), diagnostico)).

posible_caso:-leucopenia(Leucograma),

Leucograma < 5,

diagnostico.

posible_caso:-hematocrito(Hematocrito),

Hematocrito > 38,

Hematocrito < 45,

diagnostico.

posible_caso:-proteinuria(si),

diagnostico.

diagnostico:- write('Posible caso crítico').

no_diagnostico:- write('No se puede concluir ningún diagnóstico ').

/*Bloque de explicación */

explicacion:-plaquetas(Plaquetas),

(Plaquetas < 65.5; Plaquetas=100 ; Plaquetas<100),

write('Explicación').

explicacion:- plaquetas(Plaquetas),

creatinina(Creatinina),

(Plaquetas > 65.5), (Plaquetas < 123.5), (Creatinina > 83; Creatinina = 83),

write('Explicación').

explicacion:-plaquetas(Plaquetas),

creatinina(Creatinina),

hemoglobina(Hemoglobina),

(Plaquetas > 65.5), (Plaquetas < 123.5),(Creatinina < 83),(Hemoglobina < 148.5),

write('Explicación').

explicacion:-leucopenia(Leucograma),

Leucograma < 5,

write('Explicación').

explicacion:-citolisis_hepatica_ALT(ALT),

citolisis_hepatica_AST(AST),

((ALT > 40, AST > 40), write('Explicación');

((ALT > 40; AST >40), write('Explicación').

explicacion:- hematocrito(Hematocrito),

Hematocrito > 38,

Hematocrito < 45,

write('Explicación').

explicacion:- proteinuria(si),

write('Explicación').

explicacion:-write('Ya que no se tienen argumentos necesarios. ').

Se decidió hacer la base de conocimiento de esta manera para poder manipular los bloques desde Delphi.

2.4.1.1 ¿Cómo trabajar con AMZI! Prolog?

AMZI-Prolog funciona con un par de bibliotecas DLL incluidas en **LogicServer**. Esta es una biblioteca de enlace dinámico que incorpora una máquina de inferencia Prolog y tiene soporte con Borland Delphi. El uso de LogicServer con Borland Delphi se facilita por una componente que implementa las llamadas a la DLL, así como a otros tipos de datos necesarios. Las bibliotecas disponibles por LogicServer presentan una gran variedad de procedimientos y funciones para manipulación de datos, predicados y términos de Prolog.

Los archivos necesarios para lograr esta interacción son:

- Biblioteca de enlace dinámico (DLL): \bin\AMZI.DLL
- Archivo fuente de componente para Delphi: \sapis\delphi\AMZI.PAS
- Archivo utilizado para “Linkear” un fuente de Amzi! Prolog: \abin\ALIB.XPL

Una componente de LogicServer disponible para Delphi es TLSEngine, siendo posible que esté entre sus métodos la llamada a todas las funciones de la DLL. Para instalarla se debe seguir los siguientes pasos:

- Asegurarse que la DLL (\bin\Amzi.dll) esté debidamente instalada en el sistema.

- En caso de que de error al instalar se incluye en los “USES” de la aplicación.

Para trabajar con AMZI-Prolog primeramente se debe crear un nuevo proyecto (para esto debe ir a File/New/Proyect) dentro de este se crea un nuevo fichero (para esto debe ir a New/File) y guardarlo con extensión **.pro**, en el que se desarrolla la base de conocimiento. Para la utilización de LogicServer se necesita un archivo fuente debidamente compilado y linkeado para esto en el menú se accede a “run\run as\ compiled Project” y se obtiene el fichero **.xpl** necesario para la interacción con LogicServer.

2.4.2 Diseño de los módulos del sistema.

Cada módulo posee una base de conocimiento, la cual está dividida en tres partes, un bloque de preguntas, un bloque de reglas y un bloque de explicación; un motor de inferencia y una interfaz del usuario. En la figura 2.1 se presenta el diseño.

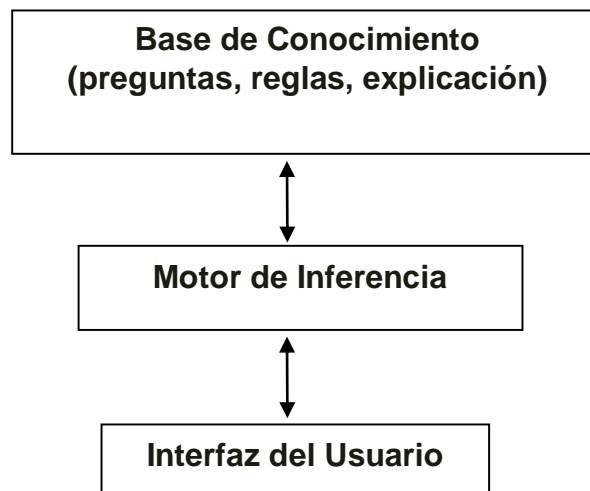


Figura. 2.1- Diseño del módulo.

2.4.3 Diálogos del sistema.

Cada módulo comienza con una pregunta del sistema para que el médico responda, en dependencia de la respuesta SEC-Dengue sigue un camino u otro en el árbol de soluciones posibles. El diálogo continúa hasta que el sistema sea capaz de concluir una clasificación o indicando que no pudo clasificar.

Diálogo del módulo Signos de Alarma:

T: ¿El paciente presenta fiebre elevada, constante y diaria? Respuesta del médico

T: Respuesta del sistema.

T: ¿El paciente presenta cefalea intensa, constante y opresiva? Respuesta del médico

T: Respuesta del sistema.

T: ¿El paciente presenta dolor retrocular, al movimiento ocular e intenso? Respuesta del médico

T: Respuesta del sistema.

T: ¿El dolor abdominal se hace intenso y mantenido? Respuesta del médico

T: Respuesta del sistema.

T: ¿El paciente presenta astenia marcada? Respuesta del médico

T: Respuesta del sistema.

T: ¿El paciente presenta vómitos reiterados? Respuesta del médico

T: Respuesta del sistema.

T: ¿El paciente presenta anorexia? Respuesta del médico

T: Respuesta del sistema.

T: ¿El paciente presenta rash? Respuesta del médico

T: Respuesta del sistema.

T: ¿El paciente presenta gingivorragia? Respuesta del médico

T: Respuesta del sistema.

T: ¿El paciente presenta mialgias? Respuesta del médico

T: Respuesta del sistema.

T: ¿El paciente presenta artralgias? Respuesta del médico

T: Respuesta del sistema.

T: ¿El paciente presenta linfadenopatías? Respuesta del médico

T: Respuesta del sistema.

Diálogo del módulo Antecedentes Patológicos:

T: ¿El paciente presenta Sicklemia como antecedente? Respuesta del médico

T: Respuesta del sistema.

T: ¿El paciente presenta Asma Bronquial como antecedente? Respuesta del médico

T: Respuesta del sistema.

T: ¿Cuál es el color de la piel del paciente? Respuesta del médico

T: Respuesta del sistema.

T: ¿El paciente presenta Diabetes Mellitus como antecedente? Respuesta del médico

T: Respuesta del sistema.

T: ¿El paciente presenta Hipertensión Arterial? Respuesta del médico

T: Respuesta del sistema.

T: ¿El paciente presenta antecedentes de Alergia? Respuesta del médico

T: Respuesta del sistema.

T: ¿Género del paciente? Respuesta del médico

T: Respuesta del sistema.

T: ¿El paciente se encuentra en la edad comprendida entre 30 y 39 años? Respuesta del médico

T: Respuesta del sistema.

T: ¿El paciente presenta Cardiopatía Isquémica? Respuesta del médico

T: Respuesta del sistema.

T: ¿El paciente presenta Hepatopatías? Respuesta del médico

T: Respuesta del sistema.

Diálogo del módulo Hallazgos de Laboratorio:

T: ¿Cuál es el valor del conteo de las plaquetas? Respuesta del médico

T: Respuesta del sistema.

T: ¿Cuál es el valor de la creatinina? Respuesta del médico

T: Respuesta del sistema.

T: ¿Cuál es el valor de la hemoglobina? Respuesta del médico

T: Respuesta del sistema.

T: ¿Cuál es el valor de transaminasas hepáticas: aspartato aminotransferasa (AST) y alanin aminotransferasa (ALT)? Respuesta del médico

T: Respuesta del sistema.

T: ¿Cuál es el valor del leucograma? Respuesta del médico

T: Respuesta del sistema.

T: ¿Cuál es el valor del Hematocrito? Respuesta del médico

T: Respuesta del sistema.

T: ¿El paciente presenta proteinuria en el sedimento urinario? Respuesta del médico

T: Respuesta del sistema.

Diálogo del módulo Hallazgos de Ultrasonido:

T: ¿En el paciente se constata derrame pleural o ascítico? Respuesta del médico

T: Respuesta del sistema.

T: ¿El paciente presenta derrame pericárdico? Respuesta del médico

T: Respuesta del sistema.

T: ¿El paciente presenta EPV? Respuesta del médico

T: Respuesta del sistema.

T: ¿El paciente presenta edema perivesicular? Respuesta del médico

T: Respuesta del sistema.

Diálogo del módulo Manifestaciones Clínicas:

T: ¿El paciente presenta como manifestación clínica hemorragias como son: metrorragia, gingivorragia o petequias? Respuesta del médico

T: Respuesta del sistema.

T: ¿El paciente manifiesta náuseas? Respuesta del médico

T: Respuesta del sistema.

T: ¿El paciente manifiesta diarreas? Respuesta del médico

T: Respuesta del sistema.

T: ¿El paciente manifiesta prurito? Respuesta del médico

T: Respuesta del sistema.

T: ¿El paciente manifiesta mareos? Respuesta del médico

T: Respuesta del sistema.

2.4.4 Diagrama de casos de uso del sistema.

El modelo de casos de uso permitió al desarrollador del sistema y a los clientes que llegaron a un acuerdo sobre las condiciones y posibilidades que debe tener el sistema. Este diagrama se muestra en la figura 2.2.

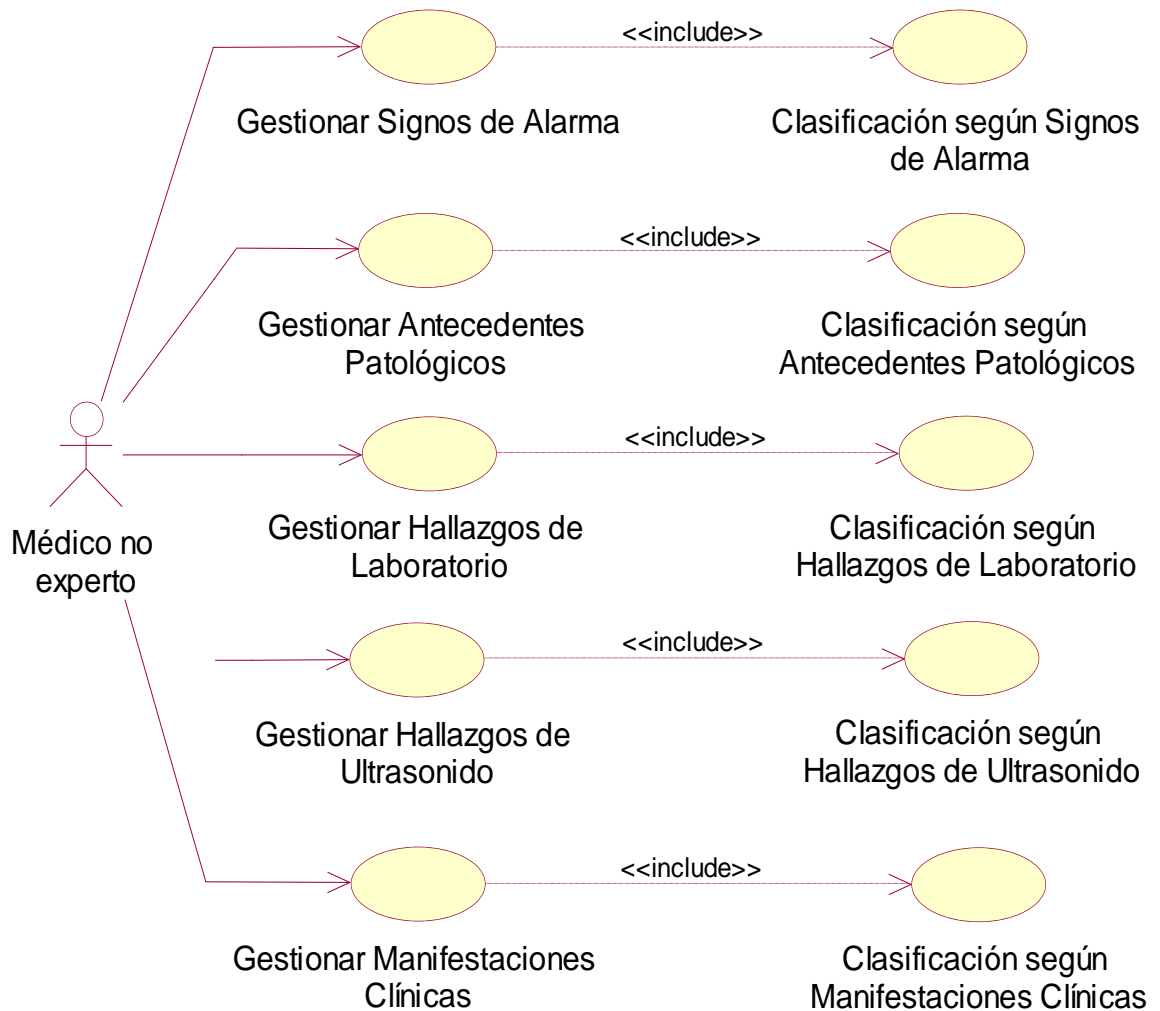


Figura. 2.2- Diagrama de caso de Uso del Sistema.

2.4.5 Descripción de los casos de usos.

Caso de uso: Signos de Alarma
Actores: Médico no experto en Dengue (inicia).
Propósito: Solicitar ayuda al sistema para clasificar a un paciente en posible caso crítico, caso de alarma o caso de alerta.
Resumen: El caso de uso se inicia cuando un médico no experto en Dengue, desea consultar el sistema para clasificar a un paciente en posible caso crítico o caso de alarma. El sistema le realiza preguntas al médico y tomando las respuestas en consideración, hace la posible clasificación. El caso de uso finaliza cuando el sistema logra una posible clasificación.
Pre-condiciones: El médico debe conocer los signos de alarma del paciente.
Pos-condiciones: Que el sistema haga una posible clasificación.
Prototipo: Ver anexo 2

Caso de uso: Antecedentes Patológicos
Actores: Médico no experto en Dengue (inicia).
Propósito: Solicitar ayuda al sistema para clasificar a un paciente en posible caso crítico, caso de alarma o caso de alerta.
Resumen: El caso de uso se inicia cuando un médico no experto en Dengue, desea consultar el sistema para clasificar a un paciente en posible caso crítico o caso de alarma. El sistema le realiza preguntas al médico y según sus respuestas se hace la posible clasificación. El caso de uso finaliza cuando se logra una posible clasificación.
Pre-condiciones: El médico debe conocer los antecedentes patológicos del paciente.

Pos-condiciones: Que el sistema haga una posible clasificación.

Prototipo: Ver anexo 3

Caso de uso: Hallazgos de Laboratorio

Actores: Médico no experto en Dengue (inicia).

Propósito: Solicitar ayuda al sistema para clasificar a un paciente en posible caso crítico, caso de alarma o caso de alerta.

Resumen: El caso de uso de inicia cuando un médico no experto en Dengue, desea consultar el sistema para clasificar a un paciente en posible caso crítico o caso de alarma. El sistema le realiza preguntas al médico y según sus respuestas el sistema hace la posible clasificación El caso de uso finaliza cuando se logra una posible clasificación.

Pre-condiciones: El médico debe conocer el resultado de los hallazgos de laboratorio del paciente.

Pos-condiciones: Que el sistema haga una posible clasificación.

Prototipo: Ver anexo 4

Caso de uso: Hallazgos de Ultrasonido

Actores: Médico no experto en Dengue (inicia).

Propósito: Solicitar ayuda al sistema para clasificar a un paciente en posible caso crítico, caso de alarma o caso de alerta.

Resumen: El caso de uso de inicia cuando un médico no experto en Dengue, desea consultar el sistema para clasificar a un paciente en posible caso crítico o caso de alarma. El sistema le realiza preguntas al médico y en dependencia de sus respuestas hace la posible

clasificación El caso de uso finaliza cuando el sistema hace una posible clasificación.
Pre-condiciones: El médico debe conocer el resultado de los hallazgos de ultrasonido del paciente.
Pos-condiciones: Que el sistema haga una posible clasificación.
Prototipo: Ver anexo 5

Caso de uso: Manifestaciones Clínicas
Actores: Médico no experto en Dengue (inicia).
Propósito: Solicitar ayuda al sistema para clasificar a un paciente en posible caso crítico, caso de alarma o caso de alerta.
Resumen: El caso de uso de inicia cuando un médico no experto en Dengue, desea consultar el sistema para clasificar a un paciente en posible caso crítico o caso de alarma. El sistema le realiza preguntas al médico y tomando en cuenta las respuestas lleva a cabo la posible clasificación El caso de uso finaliza cuando el sistema hace una posible clasificación.
Pre-condiciones: El médico debe conocer las manifestaciones clínicas del paciente.
Pos-condiciones: Que el sistema haga una posible clasificación.
Prototipo: Ver anexo 6

2.4.6 Implementación de la interfaz del usuario.

Después de la formalización de la base de conocimiento, mediante las reglas de producción y la definición de los requerimientos del sistema se pasó a la implementación de la interfaz

del usuario. Para desarrollar la interface visual se utilizó Borland Delphi 7 por ser un entorno de desarrollo rápido de aplicaciones y poseer facilidades para interactuar con AMZI-Prolog.

En AMZI- Prolog se genera un fichero **.xpl** el cual es el que se va a utilizar para consultar la base de conocimiento, a través de una componente de LogicServer disponible para Delphi (TLSEngine).

LogicServer utiliza principalmente los siguientes tipos de datos:

TTerm = pointer; Este tipo representa los términos utilizados en los métodos de la clase.

TPTYPE = (pATOM, pINT, pSTR, pFLOAT, pSTRUCT, pLIST, pTERM, pADDR, pVAR); Este tipo enumerado representa los tipos de datos utilizados para las operaciones con predicados.

TDTYPE = (dATOM, dSTR, dINT, dLONG, dSHORT, dFLOAT, dDOUBLE, dADDR, dTERM); representa los formatos de datos para Delphi, bastante útil en la interacción con predicados.

Para poder manipular el fichero **.xpl** generado se debe incluir en el proyecto de Delphi una unit llamada Amzi.pas; para acceder a esta unit debe ir a “Archivos de programa\amzi\amzi_7-0-9\lsapis\delphi”.

Hecho esto ya se puede tener acceso a los procedimientos y funciones que permiten el enlace; algunas funciones que permiten el enlace son:

procedure Init(xplname: String); Inicializa el ambiente de LogicServer, pudiéndose utilizar los demás métodos de la clase. Como argumento tiene el nombre del archivo **.xpl**, este argumento es opcional, ya que la lectura de los datos se realiza a través del método **Load**.

procedure Load(xplname: String); Este método es utilizado para cargar los predicados contenidos en el archivo compilado y linkeado el cual se pasa como parámetro, con extensión **.xpl**. Todos los predicados contenidos en este archivo podrán ser accedidos por LogicServer, pero no podrán ser alterados ni eliminados.

procedure GetArg(t: TTerm; n: integer; dt: TDTType; p: pointer); Método utilizado para poner un determinado argumento de tipo **dt** contenido en el predicado asumido por **t**. El **n**-ésimo argumento será colocado en **p** para posterior manipulación.

function GetPStrArg(t: TTerm; n: integer): string; Retorna el **n**-ésimo argumento contenido en el predicado asumido por **t**, se devuelve un argumento de tipo "String".

function GetIntArg(t: TTerm; n: integer): integer; Retorna el **n**-ésimo argumento contenido en el predicado asumido por **t**, se devuelve este argumento de tipo "Integer".

function GetFloatArg(t: TTerm; n: integer): double; Retorna el **n**-ésimo argumento contenido en el predicado asumido por **t**, se devuelve este argumento de tipo "Real".

function GetArgType(t: TTerm; n: integer): TPTYPE; Retorna el tipo de datos representado por el **n**-ésimo argumento contenido en el predicado asumido por **t**.

function TermToPStr(t: TTerm): string; Convierte el termino **t** a "String".

procedure PStrToTerm(var tp: TTerm; s: string); Convierte el "String" **s** a una representación de **tp** de tipo "TTerm".

function CallPStr(var tp: TTerm; s: string): Boolean; Realiza una llamada a la base de conocimiento de LogicServer para encontrar una representación para el predicado **s**. Si la llamada encuentra una representación, el método retornará "True", en caso contrario "False". El predicado o dato resultante de la llamada estará disponible en el término **tp**, el cual podrá ser procesado y manipulado posteriormente.

function Redo: Boolean; Continúa la búsqueda de respuestas a la consulta realizada por **CallPStr**. En caso de que sea encontrada otra representación, retorna "True" y devuelve el término pasado como parámetro a la llamada anterior como nueva ocurrencia. En caso de no encontrar otras representaciones, retorna "False". Es utilizado para soluciones de alternativa.

function ExecPStr(var tp: TTerm; s: string): Boolean; Como el **CallPStr**, procesa la consulta **s** pasada como parámetro retornando el resultado en **tp**. A diferencia de que después de la llamada a **ExecPStr** no posibilita la llamada de **Redo**, no guardando de esta forma informaciones en la pila de LogicServer para posibilitar futuras llamadas.

procedure Asserta(t: TTerm); Inserta un predicado contenido en el término **t** en la base de conocimiento dinámica de LogicServer. La inserción del predicado ocurre en el tope de la base de datos de la instancia actual.

procedure Assertz(t: TTerm); Difiere de la anterior en que la inserción del predicado ocurre al final de la base de conocimiento.

procedure Retract(t: TTerm); Utilizado para remover un predicado (asumido por **t**) de la base de conocimiento dinámica de LogicServer.

procedure AssertaPStr(s: string); Funciona exactamente como **Asserta**, el predicado a ser insertado está representado por el "String" **s**.

procedure AssertzPStr(s: string); Idéntico a la llamada de **Assertz**, pasando como parámetro el predicado en forma de "String".

procedure RetractPStr(s: string); Lo mismo que **Retract**, pero con un predicado en forma de "String".

Con todos los procedimientos antes descritos se puede pasar a la implementación de la interfaz gráfica del usuario.

2.5 Diseño de la interfaz gráfica.

Para lograr una mayor eficiencia en el proceso de trabajo, y sobre todo para lograr una coherencia formal entre todos los módulos del sistema, se han pautado una serie de elementos comunes que facilitan su uso y reconocimiento.

Se diseñó una pantalla inicial global del Sistema Experto, desde la cual se accederá a los diferentes módulos del este.

El diseño está determinado fundamentalmente por el principio de la usabilidad, donde este tiene como principal propósito facilitar su uso y comprensión manteniendo pautas estéticas y agradables.

La resolución óptima para la cual está diseñada la aplicación es de 800 x 600 px. El logo siempre estará ubicado la parte superior de cada módulo. La tipografía será siempre Arial, por su amplia legibilidad y por las facilidades conocidas que brinda para la lectura digital.

El diseño de la interfaz es muy simple de comprender por el usuario (ver anexo 1).

2.6 Conclusiones.

En el presente capítulo se identificaron los elementos del dominio y los conceptos descriptivos de sus propiedades; y las relaciones entre ellos. Se formalizaron e implementaron las reglas, se describió el modelo de los diálogos del sistema. También se definieron los casos de usos del sistema y se describió la implementación y diseño de la interfaz gráfica del sistema experto.

*“Como quien se quita un manto
y se pone otro,
es necesario poner de lado
la Universidad antigua,
y alzar la nueva”*

José Martí

Capítulo 3: Estudio de factibilidad.

3.1 Introducción.

Es necesario antes de llevar acabo de un proyecto, estimar el esfuerzo humano, el tiempo de desarrollo que se requiere para su ejecución y el costo. En este trabajo estas estimaciones se realizan a través del método de puntos de función del modelo de COCOMOII. .

3.2 Planificación.

Uno de los factores importantes a tener en cuenta en el diseño o mejoramiento de una aplicación informática, está relacionado con las ventajas del sistema propuesto que justifiquen o no su costo.

Los sistemas informáticos tienen como objetivo fundamental ofrecer la información adecuada en el momento que se solicite, pero si los ahorros que se obtienen con la información registrada y procesada, no compensan su costo, pueden no ser rentables. Sin embargo, la rentabilidad de un sistema de este tipo a veces resulta difícil de estimar, pues el valor de la información no es fácilmente cuantificable.

Entradas Externas:

Tabla 3.1 Entradas Externas.

Entrada externa	Ficheros	Elementos de Datos	Clasificación
Insertar respuesta de antecedentes de	1	1	S

Sickleemia del paciente.			
Insertar respuesta de antecedentes de Asma Bronquial del paciente.	1	1	S
Insertar respuesta de color de la piel del paciente.	1	1	S
Insertar respuesta de antecedentes de Diabetes Mellitus del paciente.	1	1	S
Insertar respuesta de antecedentes de Hipertensión Arterial del paciente.	1	1	S
Insertar respuesta de antecedentes de Alergia del paciente.	1	1	S
Insertar respuesta de género del paciente.	1	1	S
Insertar respuesta de edad del paciente.	1	1	S
Insertar respuesta de antecedentes de Cardiopatía Isquémica del paciente.	1	1	S
Insertar respuesta de antecedentes de Hepatopatías del paciente.	1	1	S
Insertar respuesta del valor del conteo de las plaquetas del paciente.	1	1	M
Insertar respuesta del valor de la creatinina del paciente.	1	1	M
Insertar respuesta del valor de la hemoglobina del paciente.	1	1	M
Insertar respuesta del valor de	1	1	M

transaminasas hepáticas del paciente.			
Insertar respuesta del valor del leucograma del paciente.	1	1	M
Insertar respuesta del valor del Hematocrito del paciente.	1	1	M
Insertar respuesta de resultados de hallazgos de ultrasonido del paciente.	1	4	M
Insertar respuesta de manifestación clínica hemorragias del paciente.	1	1	M
Insertar respuesta de manifestación de náuseas del paciente.	1	1	S
Insertar respuesta de otras manifestaciones clínicas del paciente.	1	4	M
Insertar respuesta de signos de fiebre, cefalea y dolor retroocular del paciente.	1	3	M
Insertar respuesta de signos de dolor abdominal, astenia marcada, vómitos reiterados y anorexia del paciente.	1	4	M
Insertar respuesta de signos de rash, gingivorragia, mialgias, artralgias y linfadenopatías del paciente.	1	5	M
TOTAL	Simple: 11, Media: 12, Compleja: 0.		

Salidas Externas:

Tabla 3.2 Salidas Externas.

Salida Externa	Ficheros	Elementos de datos	Clasificación
Clasificación según antecedentes patológicos personales.	1	1	M
Clasificación según hallazgos de laboratorio.	1	1	M
Clasificación según hallazgos de ultrasonido.	1	1	M
Clasificación según manifestaciones clínicas.	1	1	M
Clasificación según signos de alarma.	1	1	M
TOTAL	Simple: 0, Media: 5, Compleja: 0.		

Peticiones:

Tabla 3.3 Peticiones.

Petición	Ficheros	Elementos de datos	Clasificación
Antecedentes patológicos.	1	10	M
Hallazgos de laboratorio.	1	7	C
Hallazgos de ultrasonido.	1	4	S
Manifestaciones clínicas.	1	6	S

Signos de alarma	1	12	C
TOTAL	Simple: 2, Media: 1, Compleja: 2.		

Ficheros Internos:

Tabla 3.4 Ficheros Lógicos Internos.

Fichero Interno	Records	Elementos de datos	Clasificación
Antecedentes patológicos.	1	10	M
Hallazgos de laboratorio.	1	7	C
Hallazgos de ultrasonido.	1	4	S
Manifestaciones clínicas.	1	6	S
Signos de alarma	1	12	C
TOTAL	Simple: 2, Media: 1, Compleja: 2.		

Puntos de Función sin ajustar

Tabla 3.5 Puntos de Función sin ajustar

Elementos	S	X Peso	M	X Peso	C	X Peso	PF. Subtotal
Ficheros lógicos internos	2	(*7)	1	(*10)	2	(*15)	54
Entradas externas	11	(*3)	12	(*4)	0	(*6)	81
Salidas externas	0	(*4)	5	(*5)	0	(7)	25

Peticiones	2	(*3)	1	(*4)	2	(6)	22
TOTAL							182

Cálculo de las instrucciones fuentes:

El cálculo de las instrucciones fuentes, se basa en la cantidad de instrucciones por punto de función que genera el lenguaje de programación empleado.

Tabla 3.6 Instrucciones fuentes.

Características	Valor	
Puntos de función desajustados	182	
Lenguaje	Prolog	Object Pascal
% de utilización en la aplicación	90% (≈164)	10% (≈18)
Instrucciones fuentes por puntos de función	49	29
Instrucciones fuentes	8 036	522
Total Instrucciones fuentes	8 558	

3.3 Costos.

Multiplicadores de esfuerzo:

Tabla 3.7 Definición de los Multiplicadores de Esfuerzo (MEj).

Multiplicador	Descripción	Valor
RCPX	La complejidad del producto es alta.	1.30

RUSE	Se implementa código reutilizable para su aprovechamiento en el proyecto.	1
PDIF	La plataforma es estable. Requerimientos bajos de almacenamiento y tiempo de ejecución.	0.87
PERS	La capacidad de los especialistas es alta. La continuidad del personal es alta.	0.83
PREX	El equipo tiene poco dominio y conocimiento del lenguaje de programación, plataforma y herramientas de desarrollo utilizados. No ha desarrollado aplicaciones similares, casi ninguna experiencia.	1
FCIL	Se utilizan herramientas e instrumentos de programación modernos.	0.87
SCED	Los requerimientos de calendario de desarrollo son bajos.	1
		0.82

$$EM = \prod_{j=1}^7 E_{mi} = RCPX * RUSE * PDIF * PERS * PREX * FCIL * SCED = 0.82$$

Factores de Escala:

Tabla 3.8 Definición de los valores de los Factores de Escala (SFi).

Factor	Descripción	Valor
PREC	El sistema posee aspectos novedosos.	2,48
FLEX	El sistema cuenta con alguna flexibilidad en relación con las especificaciones de los requerimientos preestablecidos y a las especificaciones de interfaz externa.	2.03

TEAM	Interacciones principalmente cooperativas. Mediana experiencia previa operando.	1.10
RESL	La arquitectura es sólida y los riesgos generalmente se mitigan. Poca incertidumbre, riesgos no son críticos.	1.41
PMAT	Relación con el proceso de madurez del software. Nivel 3.	3.12
$\sum_{i=1}^5 SF_i$		10.14

$$SF = \sum_{i=1}^5 SF_i = PREC + FLEX + RESL + TEAM + PMAT = 10.14$$

Valores calibrados:

$$A = 2.94; B = 0.91; C = 3.67; D = 0.28$$

$$E = B + 0.01 * \sum SF_i = 0.91 + 0.01 * 10.14 = 1.01$$

$$F = D + 0.2 * (E - B) = 0.28 + 0.2 * (1.01 - 0.91) = 0.30$$

Cálculo del esfuerzo (PM):

$$PM = A * (MSLOC)^E * II Emi = 2.94 * (8,558)^{1.01} * 0,82 = 21 \text{ Hombres/Mes.}$$

Cálculo del tiempo de desarrollo:

$$TDEV = C * PM^F = 3.67 * (21)^{0.30} = 9.14 \approx \mathbf{9 \text{ meses (Estimado)}}$$

Cálculo de la cantidad de hombres:

$$CH = PM / TDEV = 21 / 9 = \mathbf{2,33 \approx 2 \text{ hombres}}$$

Como el trabajo lo realizó una personas, se recalcula el tiempo de desarrollo para la cantidad real de hombres.

$$CH^* = \mathbf{1 \text{ hombres.}}$$

TEDV = PM/CH* = 21/1= 21 **meses**.

Cálculo del costo:

Asumiendo como salario promedio mensual (SP) \$275.00

$$CHM = CH * SP = 1 * \$275.00 = \$275,00$$

$$\text{Costo} = CHM * PM = \$275,00 * 21 = \$ 5 775,00$$

Cálculos:

Tabla 3.9 Cálculo del esfuerzo, tiempo de desarrollo, cantidad de hombres y costo.

Cálculo de:	Valor
Esfuerzo (PM: Hombres - mes)	21 Hombres/Mes
Tiempo de Desarrollo(meses)	9 Meses
Cantidad de Hombres	1
Costo	\$ 5 775
Salario medio	\$ 275.00

3.4 Beneficios tangibles e intangibles.

Beneficios intangibles:

Los beneficios que se obtendrán con el desarrollo del sistema propuesto, son fundamentalmente intangibles, debido a que el sistema permitirá multiplicar la experticia lograda por los médicos del IPK, en la clasificación temprana de pacientes con Dengue Severo.

Beneficios tangibles:

En paralelo a la etapa de modelado y de implementación del Sistema Experto para la clasificación temprana de Dengue Severo en el IPK de Ciudad de la Habana, se establecen las fórmulas comerciales necesarias para lograr la introducción estable e incremental de este producto en el mercado internacional, de forma tal que los beneficios tangibles esperados se materialicen.

3.5 Análisis de costos y beneficios.

El desarrollo de todo producto informático va asociado a un costo, el justificarlo depende de los beneficios tangibles e intangibles que trae consigo.

Este recurso permitirá disminuir el tiempo en el proceso de clasificación de los casos de Dengue. Con la información que brindará el sistema, se podrá detectar de forma rápida si existe algún paciente que vaya a debutar con Dengue Severo, lo cual facilitará la toma de decisiones e implementar las medidas pertinentes con mayor rapidez y seguridad.

Analizando el costo del proyecto que es de \$ 5 775.00 contra los beneficios que reportará, en relación con un tema tan sensible como la salud de la población y la necesidad de contar con un recurso informático y que al mismo tiempo sea capaz de multiplicar la experticia lograda por los médicos del IPK, en la clasificación temprana de los pacientes con Dengue Severo, se determinó que es factible implementar el sistema experto que contribuya a facilitar la prevención.

3.6 Conclusiones

Una vez terminado el estudio de factibilidad del sistema, se estima un tiempo de 9 meses para su construcción por un hombre y su costo asciende a \$ 5 775.00.

CONCLUSIONES

Con la realización de la presente investigación se arribaron a las siguientes conclusiones:

1. Se evidenció a través del análisis bibliográfico que, las reglas de producción son una forma efectiva para describir el conocimiento en un sistema experto. El lenguaje de programación Prolog, resulta apropiado para la construcción de sistemas expertos.
2. Se determinó construir un Sistema Experto para clasificar pacientes adultos, tomando en cuenta los Signos de Alarma, los Antecedentes Patológicos, los Hallazgos de Laboratorio, los Hallazgos de Ultrasonido o las Manifestaciones Clínicas de estos.
3. Se seleccionó la metodología del Dr. Mateo Lezcano (Lezcano Brito, 1995) para la construcción del Sistema Experto y de forma complementaria el Lenguaje Unificado del Modelo.
4. El análisis de costos-beneficios permitió determinar que es factible la construcción del Sistema experto.
5. Se implementó un Sistema Experto, utilizando como herramienta para desarrollar la base de conocimiento AMZI! Prolog y para la interfaz visual Borland Delphi 7.

RECOMENDACIONES

Al concluir esta investigación se hacen las siguientes recomendaciones:

- Profundizar en las explicaciones que se ofrecen al usuario.
- Incorporarle a las reglas planteadas en la base de conocimiento el grado de certeza asociado a ellas.
- Incorporar nuevo conocimiento al sistema experto de modo que pueda clasificar pacientes de diferentes edades.

REFERENCIAS BIBLIOGRÁFICAS

- Allende, H., & Rodrigo, S. (08 de Marzo de 2003). Gsync.es. Recuperado el 10 de Enero de 2010, de Gsync.es: <http://gsync.es/jmplaza/papers/tutorialRN.ps.gz>
- Bello, R., Gálvez, D., García, M., & Lezcano, M. (1995). *Modelos Computacionales Avanzados*. Santa Clara, Cuba: Universidad Central "Marta Abreu" de Las Villas.
- Bucknall, J. (2006). *The Tomes of Delphi™ Algorithms and Data Structures*. Texas: Wordware Publishing.
- Cañas Plaza, J. M. (29 de Agosto de 1997). Gsync.es. Recuperado el 08 de Marzo de 2010, de Gsync.es: <http://gsync.es/jmplaza/papers/tutorialAG.ps.gz>
- Castillo, E., Gutiérrez, J. M., & Hadi, A. S. (1999). *Sistemas Expertos y Modelos de Redes Probabilísticas*. España: Editorial Academia de Ingeniería, Universidad de Cantabria.
- Coello Coello, C. A. (2005). *La Computación Evolutiva en el Contexto de la Inteligencia Artificial*. México: LANIA.
- Coello Coello, C. A. (29 de Agosto de 1997). *Redcientifica.com*. Recuperado el 08 de Marzo de 2010, de Redcientifica.com: <http://www.redcientifica.com/doc/doc199904260011.html>
- Elaine, R., & Kevin, K. (1994). *Inteligencia Artificial, Segunda edición*. Madrid: McGRAW-HILL/INTERAMERICANA DE ESPAÑA, S.A.
- Gálvez Lio, D. (1998). *Curso de Sistemas Basados en el Conocimiento*. Santa Clara: Ediciones Universidad Central de las Villas.
- García Valdivia, Z. Z., Bello Pérez, R. E., Gálvez Lío, D., Lezcano Brito, M. G., & Reynoso Lobato, A. (2000). *Introducción a la Inteligencia Artificial*. La Nogalera, Guadalajara, Jalisco: Editorial Pandora, S.A. de C.V. .
- Goicoechea Castaño, M. (2005). *Sistema experto de información para la toma de decisiones en la Industria Textil*. España: Universidad de Vigo.
- Gutiérrez de Piñerez Reyes, R. E. (2008). Módulo Tutor Inteligente como Aplicación Pedagógica de la Ley Distributiva. *Revista de Enseñanza y Tecnología* , 4.

- Guzmán Tirado, M. (1984). *Dengue hemorrágico: algunos aspectos clínicos y virológicos de la epidemia ocurrida en Cuba en 1981*. La Habana: Instituto de Ciencias Médicas de La Habana.
- Jacobson, I., Booch, G., & Rumbauch, J. (2006). *El Proceso Unificado de Desarrollo de Software*. Madrid, España: Addison Wesley Longman Inc.
- Kasabov, N. K. (1998). *Foundations of Neural Networks, Fuzzy Systems, and Knowledge Engineering*. Cambridge, Massachusetts, London, England: Massachusetts Institute of Technology.
- Lezcano Brito, M. (1995). *Prolog y los Sistemas Expertos*. Santa Clara: Ediciones Universidad Central de las Villas.
- M, G., O, P., G, K., I, Q., S, V., M, P., y otros. (2006). *Caracterización final y lecciones de la epidemia de Dengue 3 en Cuba, 2001-2002*. . Panamá: Salud Pública.
- Martínez Viera, R., Blanco Sánchez, N., & González Ávila, M. (1989). *Diccionario Terminológico de Biología*. Ciudad de la Habana: Científico-Técnica.
- PAHO. (1994). *Dengue and dengue hemorrhagic fever in the Americas: guidelines for prevention and control*. Washington: Scientific publication no.548.
- PAHO-OMS. (1986). *Review of the current status of yellow fever, dengue and dengue hemorrhagic fever in the region*. . San Juan, Puerto Rico.
- Peralta, M. (2005). *Estimación por esfuerzo basada en casos de uso*. Buenos Aires, Argentina.
- Pérez Martínez, C. (29 de diciembre de 2009). *andaluciainvestiga.com*. Recuperado el 26 de enero de 2010, de andaluciainvestiga.com: http://www.andaluciainvestiga.com/espanol/noticias/8/retinopatiadiabetica_9130.asp
- Ramírez, D. (1986). Sistemas Orientados al Diagnóstico Médico. *Revista Universidad EAFIT*, 62.
- Ramos Ríos, L. R. (07 de 01 de 2010). *APA Prolog*. Recuperado el 27 de 01 de 2010, de APA Prolog.
- Ramos Ríos, L. R. (2009). Sistema integral basado en mapas conceptuales para la enseñanza de la Programación Lineal.

- Reyes Olivans, C. (20 de Agosto de 2005). *monografía.com*. Recuperado el 12 de Marzo de 2010, de *monografía.com*: <http://www.monografias.com/trabajos26/sistemas-expertos/sistemas-expertos.shtml?monosearch>
- Ruiz, M. E. (2006). *Sistema Experto para realización de diagnóstico de parálisis facial con electromiografía: PARFAC*. San Marcos, Lima-Perú.
- Sánchez Arboláez, A. (7 de Diciembre de 2006.). *monografias.com*. Recuperado el 12 de Marzo de 2010, de *monografias.com*: <http://www.monografias.com/trabajos43/prolog-dengue/prolog-dengue2.shtml?monosearch>
- Señor García, R. F. (2007). *Revista de Ciencias Médicas La Habana*. Recuperado el 26 de enero de 2010, de *Revista de Ciencias Médicas La Habana*: http://www.cpicmha.sld.cu/hab/vol6_2_00/hab070200.htm
- Stair, R. M., & Reynolds, G. E. (2005). *Principios de Sistemas de Información. Inteligencia Artificial y Sistemas expertos*. Thomson: Editorial Thomson.
- Stevens, L. (1984). *Artificial Intelligence. The Search for the Perfect Machine*. Nueva York: Hayden Book Company, Hasbrouck Heights, N.J.
- Suárez García, O. J. (2009). *Caracterización y valor pronóstico de los signos de alarma del Dengue*. La Habana: Instituto de Medicina Tropical "Pedro Kouri".
- Tito, F., Aliendre, R., & Juaniquina, J. (2006). *Interfaces de Programación Lógica con Amzi Prolog para Delphi y Web*.
- Tkachuk, D. C., & Hirschmann, J. V. (2007). *Wintrobe's Atlas of Clinical Hematology, 1st Edition*. Lippincott Williams & Wilkins.
- Universidad de Alicante. (10 de noviembre de 2007). *Dipinnova.com*. Recuperado el 10 de enero de 2010, de *Dipinnova.com*: <http://www.dipinnova.com>
- Winton, P. H. (2005). *Inteligencia Artificial*. Ciudad de la Habana: Felix Varela.

Anexos

Anexo 1 Ventana Principal



Anexo 2 Ventana Signos de Alarma

Signos de Alarma

SEC-Dengue

Preguntas del sistema

¿El paciente presenta fiebre elevada, constante y diaria?

Responda con cuidado las preguntas

Seleccione la respuesta

Si No

Aceptar Cancelar

2010. Universidad de Sancti Spiritus "José Martí Pérez"

Anexo 3 Ventana Antecedentes Patológicos

Antecedentes Patológicos

SEC-Denque

¿El paciente presenta Sicklemia como antecedente?

Seleccione la respuesta

Si No

Aceptar Cancelar

2010. Universidad de Sancti Spiritus "José Martí Pérez"

Anexo 4 Ventana Hallazgos de Laboratorio

Hallazgos de Laboratorio

SEC-Dengue

Preguntas del sistema

¿Cuál es el valor del conteo de las plaquetas ?

Escriba la respuesta

Valor del conteo de plaquetas *10⁹/L

2010. Universidad de Sancti Spiritus "José Martí Pérez"

Anexo 5 Ventana Hallazgos de Ultrasonido

Hallazgos de Ultrasonido

SEC-Dengue

?

Ayuda

Preguntas del sistema

¿En el paciente se constata derrame pleural o ascitico?

Seleccione la respuesta

Si No

Aceptar Cancelar

2010. Universidad de Sancti Spiritus "José Martí Pérez"

Anexo 6 Ventana Manifestaciones Clínicas

Manifestaciones Clínicas

SEC-Dengue

Preguntas del sistema

¿El paciente presenta como manifestación clínica hemorragias como son: metrorragia, gingivorragia o petequias?

Seleccione la respuesta

Si No

Aceptar **Cancelar**

2010. Universidad de Sancti Spiritus "José Martí Pérez"