

**PRÁCTICA EMPRESARIAL EN LA FUNDACIÓN CARDIOVASCULAR DE
COLOMBIA, ÁREA DE INVESTIGACIÓN EN PEDIATRÍA EN LA LÍNEA DE
SISTEMAS DE AYUDA A LA TOMA DE DECISIONES MÉDICAS.**

ANDRÉS FELIPE VILLAMIZAR VECINO

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FISICOMECAÑICAS
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
BUCARAMANGA
2006**

**PRÁCTICA EMPRESARIAL EN LA FUNDACIÓN CARDIOVASCULAR DE
COLOMBIA, ÁREA DE INVESTIGACIÓN EN PEDIATRÍA EN LA LÍNEA DE
SISTEMAS DE AYUDA A LA TOMA DE DECISIONES MÉDICAS.**

ANDRÉS FELIPE VILLAMIZAR VECINO

**Informe Final de práctica empresarial presentado como requisito
para optar al título de Ingeniería de Sistemas**

Tutor:

**Ing. XIOMARA PATRICIA BLANCO
INGENIERA DE INVESTIGACIONES DE LA FCV**

Director:

**Ing. LEONEL PARRA PINILLA
PROFESOR ESCUELA DE INGENIERÍA DE SISTEMA E INFORMÁTICA**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FISICOMECAÑICAS
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
BUCARAMANGA
2006**

A mi familia, por apoyarme en
todo lo que he hecho en la vida.
A mi Angelito, por ser tan
especial conmigo.
A mi tutora, excelente jefe y
amiga.
A mis amigos, como ellos no hay
igual.

CONTENIDO

	Pág.
INTRODUCCIÓN	13
1 PRESENTACIÓN	15
1.1 ANTECEDENTES	15
1.2 OBJETIVOS DEL PROYECTO	16
1.2.1 Objetivo general	16
1.2.2 Objetivos específicos	16
1.3 ALCANCE DEL PROYECTO	17
1.4 CONTENIDO DEL INFORME	17
2 AMBIENTE DE DESARROLLO	18
3 MARCO TEÓRICO	19
3.1 INTELIGENCIA ARTIFICIAL	19
3.1.1 Áreas de acción de la inteligencia artificial	19
3.1.1.1 Procesamiento de lenguaje natural	19
3.1.1.2 Robótica	19
3.1.1.3 Reconocimiento de Imágenes	19
3.2 SISTEMAS EXPERTOS	20
3.2.1 Historia	20
3.2.2 Definiciones de un sistema experto	21
3.2.3 Arquitectura de los sistemas expertos	22
3.2.4 Cuando se justifica un sistema experto	23
3.2.5 Representación del conocimiento	24
3.2.6 Encadenamiento de reglas	25
3.3 PATOLOGÍA DE TRANSPOSICIÓN DE GRANDES ARTERIAS	26
4 DESCRIPCIÓN DEL SISTEMA	28
4.1 ANÁLISIS DEL SISTEMA	28
4.1.1 Situación actual	28
4.1.2 Sistema propuesto	28
4.2 DESARROLLO DE LA PRÁCTICA	29
4.3 FASE DE INICIO	32
4.3.1 Modelado del negocio	32
4.3.1.1 Casos de uso del modelado del negocio	32
4.3.2 Motor de inferencia CLIPS	32
4.3.2.1 Que es CLIPS	32
4.3.2.2 Representación del conocimiento en CLIPS	33
4.3.2.3 Elementos esenciales en CLIPS	34
4.3.2.4 Estrategias de resolución de conflictos	35
4.3.2.5 Diseño de módulos en CLIPS	35
4.3.3 Conocimiento transformado en reglas	36
4.3.4 Glosario	37
4.3.5 Estándar de desarrollo	37
4.3.6 Prototipo DOS	37
4.3.7 Pruebas al prototipo	42
4.4 FASE DE ELABORACIÓN	42

4.4.1	Elicitar requerimientos	42
4.4.2	Casos de uso de interacción con el usuario	43
4.4.3	Jerarquizacion del contenido	43
4.4.4	Diseño e implementación de interfaces graficas	46
4.4.5	Primer prototipo grafico	51
4.4.6	Pruebas al sistema	52
4.5	FASE DE CONSTRUCCIÓN	54
4.5.1	Modulo para la administración de reglas	54
4.5.2	Cambios en el sistema	60
4.5.3	Administración de perfiles	61
4.5.4	Copia de seguridad	63
4.5.5	Formularios de dialogo y acceso y control	64
4.5.6	Nuevos hechos	66
4.6	FASE DE TRANSICIÓN	68
4.6.1	Cambios	68
4.6.2	Estándar de desarrollo	68
4.6.3	Integración y pruebas	69
4.7	SEGUNDO CICLO	72
4.7.1	Implementación	74
4.7.2	Diseño de páginas Web para la administración del sistema	77
4.8	OTRAS ACTIVIDADES	79
5	CONCLUSIONES	81
6	PERSPECTIVAS HACIA EL FUTURO	82
	BIBLIOGRAFÍA	83
	ANEXOS	85

LISTA DE FIGURAS

	Pág.
Figura 1. Arquitectura de un sistema experto	22
Figura 2. Encadenamiento de reglas hacia delante	26
Figura 3. Transposición de grandes arterias	27
Figura 4. Proceso unificado	29
Figura 5. Principales actividades realizadas	31
Figura 6. Funcionamiento del primer programa en CLIPS	36
Figura 7. Obtención de hechos	38
Figura 8. Prototipo DOS	41
Figura 9. Diagrama de estructuras	44
Figura 10. Dialogo de ingreso de atención del paciente	47
Figura 11. Dialogo de fin de turno	47
Figura 12. Dialogo de preguntas EKG	48
Figura 13. Frame de datos del paciente	48
Figura 14. Dialogo de modificar temperatura	49
Figura 15. Dialogo de modificar presiones	49
Figura 16. Frame de alarmas y sugerencias	50
Figura 17. Diagrama de clases	51
Figura 18. Prueba del sistema experto con simuladores	53
Figura 19. Pruebas al sistema en un paciente	54
Figura 20. Modelo de bases de datos	56
Figura 21. Formulario de administración de reglas	57
Figura 22. Dialogo de creación y edición de reglas	58
Figura 23. Formulario de alarmas, sugerencia y tratamientos	61
Figura 24. Dialogo de dosis de medicamento	61
Figura 25. Tablas de usuarios	62
Figura 26. Perfiles de usuario	63
Figura 27. Copia de seguridad	64
Figura 28. Formulario de opciones del sistema	65
Figura 29. Formulario de administrar el sistema	65
Figura 30. Dialogo de password	66
Figura 31. Obtención de hechos	67
Figura 32. Interfaz web	73
Figura 33. Diagnósticos asociados a una variable	74
Figura 34. Metas y controles	75
Figura 35. Radiología	76
Figura 36. Ventana emergente de radiografías	76
Figura 37. Pagina de medicamentos	77
Figura 38. Opciones del sistema	78
Figura 39. Administración de reglas	78
Figura 40. Administración de variables	79
Figura 41. Diagrama de casos de uso del negocio	85
Figura 42. Diagrama de secuencias para el ingreso a la UCIP	89

Figura 43. Diagrama de actividades de ingreso a la UCIP	80
Figura 44. Diagrama de actividades evolucionar paciente	91
Figura 45. Diagrama de actividades egreso de la UCIP	92
Figura 46. Diagrama de casos de uso de interacción del usuario	93
Figura 47. Diagrama de secuencia ingresar atención	97
Figura 48. Diagrama de secuencias ingresar fecha y hora de fin	98
Figura 49. Diagrama de secuencias analizar EKG	98
Figura 50. Diagrama de secuencias mostrar alarmas y sugerencias	99
Figura 51. Diagrama de secuencias parametrizar signos	99
Figura 52. Diagrama de casos de uso	100
Figura 53. Diagrama de secuencia de ingreso de un protocolo	106
Figura 54. Diagrama de secuencia para crear un archivo CLP	107
Figura 55. Diagrama de secuencias de creación de reglas	108
Figura 56. Diagrama de secuencias de editar una regla	109
Figura 57. Diagrama de secuencias para la eliminación de una regla	110
Figura 58. Diagrama de secuencias de copias de seguridad	111
Figura 59. Árbol de directorios y nomenclatura	117
Figura 60. Tabla para calcular los puntos de función	121

LISTA DE TABLAS

	Pág.
Tabla 1. Relación edad, peso y frecuencias respiratorias	68
Tabla 2. Pruebas del formulario de administración de reglas	69
Tabla 3. Pruebas del dialogo de creación de reglas	70
Tabla 4. Pruebas del formulario de perfiles de usuario	70
Tabla 5. Pruebas al modulo de copias de seguridad	71
Tabla 6. Pruebas al sistema integrado	71
Tabla 7. Pruebas de rendimiento	72
Tabla 8. Componentes swing	118
Tabla 9. Contenedores swing	118
Tabla 10. Menus swing	118
Tabla 11. Tipos de datos	118

LISTA DE ANEXOS

	Pág.
Anexo A. Diagramas UML del modelado del negocio	45
Anexo B. Diagramas UML de interacción con el usuario	93
Anexo C. Diagramas UML de administrar reglas	100
Anexo D. Diagrama de secuencia para copias de seguridad	111
Anexo E. Glosario	112
Anexo F. Estándar de desarrollo	116
Anexo G. Capacitación de puntos de función	120
Anexo H. Formato de idea de proyecto de CRF	122
Anexo I. Artículo sobre el Sistema Experto	124

RESUMEN

TITULO: PRÁCTICA EMPRESARIAL EN LA FUNDACIÓN CARDIOVASCULAR DE COLOMBIA, ÁREA DE INVESTIGACIÓN EN PEDIATRÍA EN LA LÍNEA DE SISTEMAS DE AYUDA A LA TOMA DE DECISIONES MÉDICAS*.

AUTOR: Andrés Felipe Villamizar Vecino**.

PALABRAS CLAVE: Sistemas Expertos, Inteligencia Artificial, proceso unificado, administración de reglas, experiencia.

CONTENIDO:

Los sistemas expertos (SE) son una rama de la inteligencia artificial (IA) y se emplean para ejecutar tareas que son normalmente realizadas por personas que son consideradas expertas en un dominio específico; y su principal virtud es que se enfrentan a problemas del mundo real.

En las unidades de cuidados intensivos pediátricas (UCIP) se encuentran situaciones en la labor diaria como el error humano y además es uno de los escenarios médicos que dispone de más información que cualquier otro, manejando una gran cantidad de variables que la hacen muy susceptible a este error. Para combatir estas situaciones se plantea por la Fundación Cardiovascular de Colombia (FCV) la construcción de un prototipo de SE, desarrollado mediante la metodología de proceso unificado (PU), con sus cuatro fases y las disciplinas que se creyeron necesarias. Para esto la FCV necesito un estudiante que apoyara las actividades que corresponden a las disciplinas de modelado de negocio, requisitos, diseño, prueba, despliegue, gestión de la configuración y cambios, gestión del proyecto y entorno.

Dentro de las actividades se encuentran la revisión del marco teórico y estado del arte de los SE, el diseño y la implementación de los elementos fundamentales de la arquitectura de un SE. También se realizaron las actividades de los módulos necesarios para la administración del sistema, como la asignación de permisos a los diferentes usuarios, las copias de seguridad y la administración de reglas. Con todos los elementos resultantes de estas actividades se obtuvo un prototipo de SE que sentó las bases para continuar con el proyecto hasta generar un producto con mayores características, que inclusive llegue a aprender de la experiencia adquirida y así ayudar mucho mas al experto en el área medica.

* Proyecto de Grado.

** Facultad de Ingenierías Físico Mecánicas. Escuela de Ingeniería de Sistemas e Informática.
Director Leonel Parra Pinilla.

ABSTRACT

TITLE: ENTERPRISE PRACTICE IN THE COLOMBIA'S CARDIOVASCULAR FOUNDATION, PEDIATRIC INVESTIGATION AREA IN THE LINE OF SYSTEMS OF HELPING TO TAKE MEDICAL DECISIONS*.

AUTHOR: Andrés Felipe Villamizar Vecino**.

KEY WORDS: Expert Systems, Artificial Intelligence, unified process, rules administration, experience.

CONTENT:

The expert systems (ES) are a division of de artificial intelligence (IA) y they are used to make works that are normally made by people that are considered experts in a specific dominion; and their principal characteristic is they fight against problems of the true life.

In the pediatric intensive care units (PICU) we usually found situations in the daily duty as the human mistake y beside the PICU is one of the medical sceneries that have more information than any other, leading with a big amount of variables that make it very susceptible to the mistakes. To fight against these situations is planed by the Cardiovascular Foundation (FCV) to make a prototype of SE, developed by using the methodology of Unified Process (UP) with its four phases y the disciplines that was thought necessities. For making this the FCV needed one student to support the activities that correspond to the disciplines of business modeling, requisites, design, probe, release, configuration gestation and changes, project gestation and environment.

Among all the activities it's found the theory mark and state of art of the ES, the design and the implementation of the principal elements of the SE architecture. Besides was made all the activities of the necessities modules to the administration of the system like the assignation of permissions to the different users, the back ups and the rules administration. With all these resultant elements of those activities was made an ES prototype that sat the bases to continue with the project until generate a product with bigger characteristics and that it learns of the acquired experience and in that way to help much more to the expert in medical area.

* Final Project.

** Physical Mechanic Engineering Faculty. System and Informatics Engineering School. Director Leonel Parra Pinilla.

INTRODUCCIÓN

La inteligencia artificial (IA) fue introducida en el ámbito científico en la década de los 50's por Alan M. Turing en un artículo conocido como "Maquina Computacional e Inteligencia". Él básicamente intentaba responder la pregunta, pueden pensar las maquinas?; para esto propuso una prueba conocida como la prueba de Turing.

En esta prueba se trata de identificar la existencia de inteligencia en una maquina, por medio de una conversación de un humano, con otro y con una maquina; sin que este sepa con cual de los dos esta hablando realmente. Cuando el humano es incapaz de distinguir si quien le esta respondiendo es el otro humano o la maquina, en ese momento se dice que la maquina es inteligente.

Tiempo más tarde se realiza el congreso de Dartmouth donde se introdujo el término de inteligencia artificial.

La IA tiene una gran cantidad de aplicaciones o áreas de acción, una de estas es los sistemas expertos(SE). Los SE son aplicaciones informáticas que simulan el proceso de razonamiento mediante la aplicación de conocimiento obtenido de un experto y la aplicación de inferencias sobre este.

Los SE tienen sus inicios cuando se desarrolla un programa llamado General Problem Solver (GPS), el cual podía resolver problemas matemáticos como el de las torres de Hanoi. Luego se restringe el alcance de este tipo de programas a un dominio específico y se intenta simular la experiencia humana, naciendo de esta forma los SE.

Un área donde se han realizado varios SE es la medicina. Entre los SE mas destacados se encuentra MYCIN, un SE para la consulta y diagnostico de infecciones de la sangre. Una de sus características mas importantes, es que logró funcionar de manera correcta, dando conclusiones muy similares a las de un experto humano, considerándose así como el primer SE que simula confiablemente el razonamiento de un experto.

Revisando los diferentes SE que han sido elaborados, se encuentra que en la escuela de Ingeniería de Sistemas de la UIS también ha realizado varios SE como por ejemplo "ABC 1.0" un SE para la reanimación y diagnostico de pacientes politraumatizados correspondientes a casos clínicos concretos por medio de la selección de sus síntomas y signos.

Como la mortalidad en una institución medica es un factor de calidad y según ésta, la institución puede ser catalogada como una de las mejores. La mortalidad en lo mejores centros del mundo es cercana al 2%. El error en una institución no es ajeno a la labor diaria y al observar el error humano como una constante en la práctica clínica, surge la necesidad de implementar estrategias orientadas a su reducción.

Otra situación que se presenta en las instituciones médicas es el manejo de grandes cantidades de información por parte de los médicos que en muchos casos esta desorganizada y es difícilmente utilizable.

En una unidad de cuidados intensivos (UCI) pediátrica es común el manejo de postoperatorios complejos, que a pesar de tener muy bien definidas sus complicaciones y procesos médicos para resolverlas, tienen muy baja tolerancia la error. Sin embargo, existen factores como errores de comunicación, perdida e incorrecta transcripción de datos, manejo de grandes cantidades de información de paciente y del manejo de éste(valores de signos a monitorear, resultados de exámenes, protocolo de manejo de una patología específica, etc.), que esta desorganizada o que no es registrada a tiempo; que convierten a la UCI pediátrica en un sitio ideal para la aplicación de herramientas informáticas basadas en técnicas de IA que permitan una mayor eficiencia en el manejo de los pacientes, reduciendo la mortalidad.

A través de este informe se encuentran las diferentes actividades realizadas durante la práctica empresarial realizada en la Fundación Cardiovascular de Colombia, para la construcción de un prototipo de SE. En el capítulo uno se describen los objetivos de la práctica, en el segundo los elementos necesarios para la ejecución del sistema, en el tercero se encuentra toda la teoría básica sobres SE, en el capítulo cuatro se describe la metodología de desarrollo utilizada y las actividades de la práctica mas detalladamente. En los siguientes dos capítulos se encuentran las conclusiones y algunas ideas para la continuación del proyecto. Por ultimo se encuentran los anexos resultantes de las actividades realizadas.

1 PRESENTACIÓN

1.1 ANTECEDENTES

Los sistemas expertos (SE) son una rama de la Inteligencia Artificial y se emplean para ejecutar una variedad muy complicada de tareas que anteriormente solamente podían llevarse a cabo por un número limitado de persona expertas intensamente entrenadas.

Una de la característica más interesante y que distingue a los SE de la mayoría de las aplicaciones tradicionales de la computación, es su capacidad para enfrentar problemas que constituyen un reto del mundo real, realizando procesos que se asemejan al discernimiento y la intuición del ser humano.

Una de las principales áreas de aplicación de los SE es la medicina, durante muchos años se han diseñado e implementado numerosas aplicaciones, Una de los primeros SE médicos es MYCIN, que se desarrollo entre 1972 y 1980 en la universidad de Stanford, su principal función era la de aconsejar a los médicos en la determinación de diagnósticos en el campo de enfermedades infecciosas de la sangre.

También se puede mencionar otra gran cantidad de SE que se han desarrollado a través del tiempo en el área de la medicina, entre estos tenemos a DIAVAL que es un SE desarrollado en la UNED de Madrid y esta destinado a ayudar a los médicos en el diagnostico de enfermedades cardiacas a partir de los datos ecocardiográficos, También podemos mencionar otros como INTERNIST-1/CADUCEUS, SE de medicina interna y PUFF que diagnostica enfermedades pulmonares.

En la escuela de Ingeniería de Sistemas de la UIS, también se han desarrollado algunos SE en el área de la medicina, uno de estos es el Sistema Basado En EL Conocimiento Para La Reanimación y Diagnostico De Pacientes Politraumatizados "ABC 1.0", este SE ayuda en la preanimación y diagnostico de pacientes politraumatizados correspondientes a casos clínicos concretos, por medio de selección de sus síntomas y signos¹.

¹ **PINILLA**, Lucenith y **PARADA**, Yasmitd. Sistema Basado En EL Conocimiento Para La Reanimación y Diagnostico De Pacientes Politraumatizados "ABC 1.0"

1.2 OBJETIVOS DEL PROYECTO

1.2.1 Objetivo general

Aplicar los conocimientos obtenidos a lo largo de los niveles cursados en la carrera de Ingeniería de Sistemas y profundizar en ellos a través de capacitaciones en nuevas metodologías y tecnologías, que sean necesarios y útiles para la vida empresarial, así como adquirir experiencia a nivel profesional y personal que sólo puede ser vivenciada en un entorno laboral.

1.2.2 Objetivos específicos

- Adquirir conocimientos sobre el manejo de herramientas software necesarias para participar en la elaboración de un primer prototipo de un sistema experto, a través de la capacitación en su instalación, configuración, uso y mantenimiento.
- Analizar la viabilidad de usar la herramienta Clips como motor de inferencia para el primer prototipo de sistema experto, identificando las falencias y las virtudes de dicha herramienta.
- Hacer parte de las actividades para la elaboración de la interfaz de usuario, la cual debe aceptar la información que no se encuentra disponible en la historia clínica electrónica, por parte del personal medico y traducirla a una forma aceptable para el resto del sistema experto.
- Participar en la planeación, diseño e implementación de los módulos de Interfaz de usuario y acceso a hechos mediante la aplicación de la metodología de modelo unificado.
- Conocer y familiarizarse con la base de datos y la historia clínica electrónica, de manera que la información requerida pueda ser recopilada y pasada a un lenguaje de programación, para luego interactuar con el motor de inferencia.
- Tomar parte de las actividades para la realización de los medios para la explicación, ya que además de lograr simplemente una conclusión cuando enfrenta un cierto tipo de problema, un sistema experto debe ser capaz de explicar las razones por las cuales llega a dicha decisión o conclusión.

1.3 ALCANCE DEL PROYECTO

El desarrollo de esta práctica consiste en apoyar las actividades correspondientes al proyecto de SE de la Fundación Cardiovascular de Colombia, el cual ofrecerá una alternativa novedosa y de alta tecnología para el diagnóstico oportuno de las complicaciones más frecuentes y con mayor impacto sobre la morbilidad y mortalidad de los pacientes pediátricos sometidos a un procedimiento quirúrgico cardiovascular. La implementación de un sistema de inteligencia artificial como el presente podría en un futuro ser extendido a otras disciplinas y centros de atención de alta complejidad. Se ofrece un instrumento de atención especializado de bajo costo y enormes beneficios tanto para los pacientes como para el sector público y privado.

Además la interacción de grupos de investigación multidisciplinarios para el desarrollo del proyecto, permitirá dar aplicabilidad clínica a los procesos de investigación biomédica; adicionalmente constituirá una herramienta útil dentro de los procesos de enseñanza tanto de la medicina como de la ingeniería.

1.4 CONTENIDO DEL INFORME

En el presente capítulo (capítulo 1), se describen los detalles de la práctica, como su objetivo principal y los objetivos secundarios, el alcance y los antecedentes. En el segundo capítulo se describen todas las condiciones necesarias para la ejecución del SE. En el tercer capítulo se describe toda la teoría acerca de los SE y la transposición de grandes arterias que fueron esenciales para el desarrollo de la práctica. En el cuarto capítulo se describe la metodología de desarrollo utilizada para la práctica y se describen todas las actividades que se realizaron para el desarrollo de ésta. En el quinto capítulo se exponen las conclusiones del proyecto. En el sexto capítulo se propone algunas ideas para que el proyecto continúe. El séptimo capítulo son los anexos del presente proyecto.

2 AMBIENTE DE DESARROLLO

En el transcurso de la práctica empresarial se desarrolló una aplicación stand alone, usando como lenguaje de programación para su desarrollo JAVA. Los servicios y/o aplicaciones necesarias para su funcionamiento son los siguientes:

- Java Runtime Enviroment (jre) 1.5_02 para la ejecución de las clases compiladas de java.
- Gestor de base datos de Microsoft Access 2000 o superior.
- Drivers JDBC para Microsoft SQL Server 2003.
- Acceso a la Historia Clínica Electrónica.
- Librería precompilada JMatLink v1.3.0.
- Instalación de la consola de comandos y las librerías de dibujo de Matlab 6.5 o superior.
- Librerías precompiladas del motor de inferencia CLIPS 6.22.
- Librería jclips.jar 1.2.
- Un equipo que contenga el sistema del monitor de signos vitales, con un procesador de 800 Mhz y una memoria ram de 512 o superior

También se implemento una versión web, creada hacienda uso de la tecnología JSP, que requiere de los todos los anteriores junto con los siguientes:

- En el servidor, Tomcat5.x
- En el cliente, un navegador de Internet que soporte la tecnología JSP

3 MARCO TEÓRICO

3.1 INTELIGENCIA ARTIFICIAL

La inteligencia artificial (IA) es una subdisciplina informática; una técnica de software de software que se utiliza en el desarrollo de programas para la solución de problemas complejos, mediante las aplicación de procesos que son análogos al método de razonamiento humano y por tanto las soluciones son en términos simbólicos mas que una simple respuesta numérica.²

3.1.1 Áreas de acción de la inteligencia artificial.

Las áreas de acción de la inteligencia artificial más conocidas son: Procesamiento de lenguaje natural, robótica, reconocimiento de imágenes y sistemas expertos.

3.1.1.1 Procesamiento de lenguaje natural

Esta área investiga todo lo relacionado con técnicas que permitan desarrollar sistemas en los cuales la información está en lenguaje natural y posee algoritmos para el análisis sintáctico, semántico y pragmático para interpretar y generar respuestas en un lenguaje natural.

3.1.1.2 Robótica

En esta área se desarrollan las investigaciones que tienen como propósito la obtención de nuevas y mejores características para los robots. Las tareas en las que se usan los robots, son aquellas que se desarrollan en entornos peligrosos, actividades en las cuales se exige altísima precisión o respuesta en tiempo real y en aquellas donde la manipulación de piezas es difícil.

3.1.1.3 Reconocimiento de imágenes

En esta área se investiga el reconocimiento de imágenes usando procesos de finidos y se desarrollan especialmente para que los robots tengan capacidad de visión, para que estos puedan moverse y trabajar de forma inteligente. En este campo se encuentra el reconocimiento de patrones y el tratamiento de imágenes.

² **CORREDOR**, Martha Vitalia. Principios de Inteligencia Artificial y Sistemas Expertos. Bucaramanga. Ediciones UIS. 1992 p3

3.2 SISTEMAS EXPERTOS (SE)

3.2.1 Historia

El inicio de los SE inicia más o menos en la década de los sesentas, en esta época se desarrolla un programa llamado General Problem Solver (Solucionador General de Problemas). Este SE podía resolver problemas de criptoaritmética, las torres de Hanoi y otros problemas matemáticos muy similares. El único inconveniente era que este sistema no podía resolver problemas del mundo real.

Después de esto los investigadores en inteligencia artificial cambiaron el rumbo de la investigación, decidieron entonces restringir el alcance de los SE a un dominio específico e intentando simular la experiencia humana. De esta manera nacen los SE, los cuales simulan el proceso de razonamiento humano.

A partir de 1965 Edward Feigenbaum, comenzó a desarrollar un SE utilizando bases de conocimiento definidas muy cuidadosamente. Debido a esto, años más tarde se construye en Standford, DENDRAL, el cual es considerado como el primer SE. La finalidad de este SE era identificar estructuras químicas moleculares a partir de su análisis espectrográfico.

En la década de los setentas, se desarrolló MYCIN para consulta y diagnóstico de infecciones de la sangre. Este sistema introdujo nuevas características: utilización de conocimiento impreciso para razonar y posibilidad de explicar el proceso de razonamiento. Lo más importante es que funcionaba de manera correcta, dando conclusiones análogas a las que un ser humano daría tras largos años de experiencia, es por esto que se considera el primer SE que simula confiablemente el razonamiento de un experto. En MYCIN se manejan por aparte el motor de inferencia y la base de conocimientos. Al separar esas dos partes fundamentales de un SE, se puede considerar el motor de inferencias aisladamente, dando como resultado un sistema vacío o shell.

Después de esto surgieron nuevos SE médicos como el EMYCIN (MYCIN Esencial) con el que se construyó SACON, utilizado para estructuras de ingeniería, PUFF cuya función es estudiar la función pulmonar y GUIDON para elegir tratamientos terapéuticos.

En esa época se desarrollaron también: HERSAY, que intentaba identificar la palabra hablada, PROSPECTOR que tenía por objetivo la exploración mineral y ser una herramienta de identificación de yacimientos de minerales, TEIRESIAS una herramienta para transformación de conocimiento. En el año de 1975 se desarrolla en la universidad de Pittsburg un SE de diagnóstico para medicina interna.

En la década de los ochenta se ponen de moda los SE, numerosas empresas de alta tecnología investigan en este área de la inteligencia artificial, desarrollando SE para su comercialización. Se llega a la conclusión de que el éxito de un SE depende casi exclusivamente de la calidad de su base de conocimiento. El inconveniente es que codificar la pericia de un experto humano puede resultar difícil, largo y laborioso.³

En la actualidad podemos encontrar diferentes SE en diferentes que se han desarrollado en diferentes Dominios específicos e inclusive varios de estos desarrollados en la Escuela De Ingeniería De Sistemas De la UIS, como por ejemplo Sistema Basado En EL Conocimiento Para La Reanimación y Diagnostico De Pacientes Politraumatizados “ABC 1.0”

3.2.2 Definiciones de un sistema experto (SE)

En la diferente literatura que se encuentra acerca de SE, se han encontrado múltiples definiciones sobre lo que es un SE, algunas de estas son las siguientes:

Un SE es una aplicación informática que soluciona problemas complicados que de otra manera exigirían ampliamente la pericia humana. Para lograr esto, se simula el proceso de razonamiento humano mediante la aplicación específica de conocimientos y de inferencias.⁴

Es un software que imita el comportamiento de un experto humano en la solución de un problema. Pueden almacenar conocimientos de expertos para un campo determinado y solucionar un problema mediante deducción lógica de conclusiones.⁵

Son SE aquellos programas que se realizan haciendo explicito el conocimiento en ellos, que tienen información específica de un dominio concreto y que realizan una tarea relativa a este dominio.⁶

Programas que manipulan conocimiento codificado para resolver problemas en un dominio especializado en un dominio que generalmente requiere de experiencia humana.

Programas que contienen tanto conocimiento declarativo (hechos a cerca de objetos, eventos y/o situaciones) como conocimiento de control (información a cerca de los cursos de una acción), para emular el proceso de razonamiento de los expertos humanos en un dominio en particular y/o área de experiencia.⁷

³ **MONTES**, Jesús. Sistemas expertos (on line). Yucatán Mexico .

<http://www.monografias.com/trabajos16/sistemas-expertos/sistemas-expertos.shtml>

⁴ **ROLSTON**, David. Principios de Inteligencia Artificial y Sistemas Expertos. McGraw Hill. 255 p

⁵ **CRIADO BRIZ** José Mario. Introducción a los sistemas expertos (on line).

http://ingenieroseninformatica.org/recursos/tutoriales/sist_exp/cap1.php.

⁶ **WIKIPEDIA**. Sistema experto. http://es.wikipedia.org/wiki/Sistema_experto

⁷ **CASTRO**, Marcel. Sistemas expertos (on line).

http://strix.ciens.ucv.ve/~iartific/Material/PP_Sistemas_Expertos.pdf.

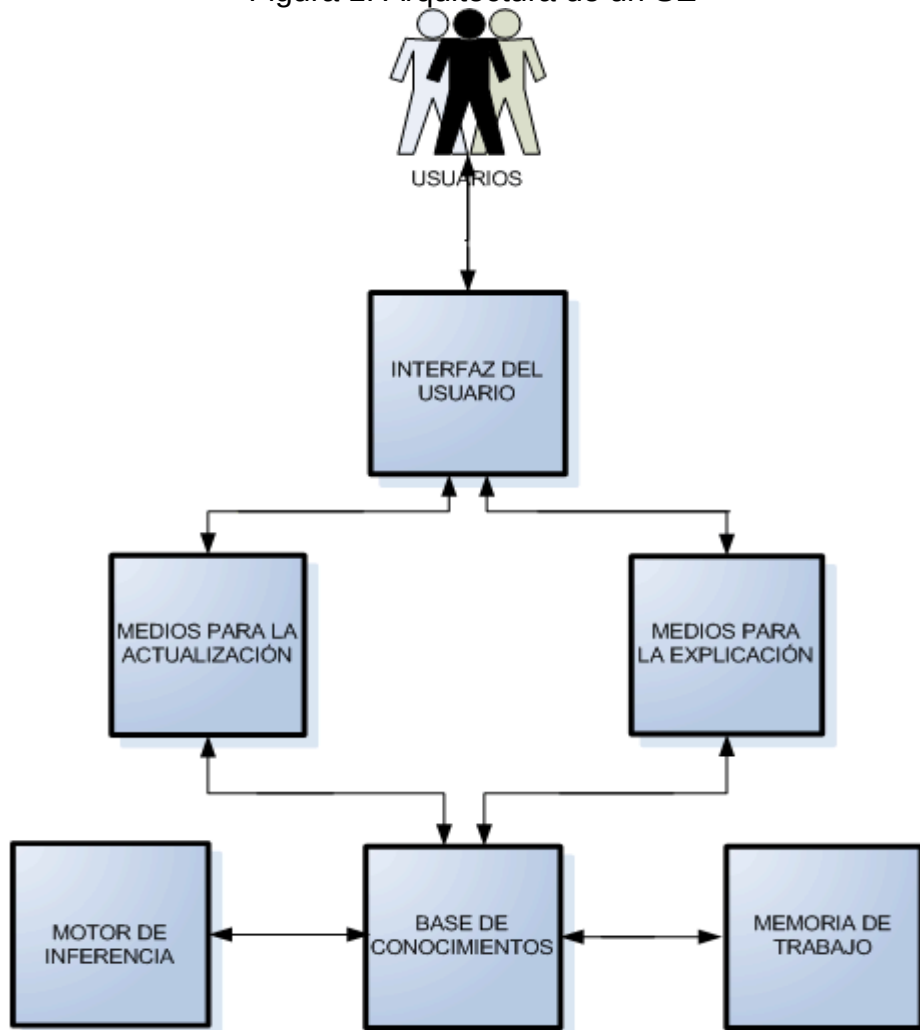
Software que incorpora conocimiento de experto sobre un dominio de aplicación dado, de manera que es capaz de resolver problemas de relativa dificultad y apoyar la toma de decisiones inteligentes en base a un proceso de razonamiento simbólico.⁸

3.2.3 Arquitectura de los sistemas expertos (SE)

Los SE emplean una amplia variedad de arquitecturas específicas en sus sistemas, principalmente porque una arquitectura es más aplicable que otra cuando se considera una aplicación dada.

A pesar de las diferencias, la mayoría de las arquitecturas tienen muchos componentes en común. A continuación se muestra una arquitectura general con los componentes típicos:

Figura 1. Arquitectura de un SE



⁸ **VIEJO**, Hernando Diego. Sistemas expertos (on line). <http://www.divulgacion.com/cursos/cursos.xml?numero=2&nombre=2003-9-26a&numLecc=1>.

- Base de conocimientos: Es la parte del SE que contiene el conocimiento sobre el dominio. Hay que obtener el conocimiento del experto y codificarlo en la base de conocimientos. La forma mas común de representar el conocimientos son las reglas, aunque existen otras formas de representar el conocimiento como los scripts y los marcos.
- Base de hechos (Memoria de trabajo): Contiene los hechos sobre un problema que se han descubierto durante una consulta. Durante una consulta con el SE, el usuario introduce la información del problema actual en la base de hechos.
- Motor de inferencia: El SE modela el proceso de razonamiento humano con un componente conocido como el motor de inferencia. El motor de inferencia junto con la base de conocimientos forman el núcleo de un SE. El motor de inferencia contiene un mecanismo de búsqueda y los algoritmos necesarios para deducir las respuestas, conclusiones y soluciones.
- Medios para la explicación: Una característica de los SE es su habilidad para explicar su razonamiento. Usando el módulo del subsistema de explicación, un SE puede proporcionar una explicación al usuario de por qué está haciendo una pregunta y cómo ha llegado a una conclusión. Este modulo es de gran utilidad ya que ayuda en la depuración del sistema, facilitando la aceptación de los usuarios y clarificando términos y conceptos manejados.
- Interfaz de usuario: Es la encargada de aceptar la información del usuario y traducirla a una forma aceptable para el resto del sistema o aceptar información proveniente del sistema y convertirla a una forma que pueda entender el usuario. La interacción entre un SE y un usuario se realiza en lenguaje natural. Para conducir este proceso de manera aceptable para el usuario es especialmente importante el diseño del interfaz de usuario.

3.2.4 Cuando Se Justifica Un Sistema Experto (SE)

La justificación de un SE depende de diferentes circunstancias que se presentan, por ejemplo una empresa podría justificar la elaboración de un SE cuando existe la posibilidad de que un proceso mal ejecutado ocasione un resultado no adecuado en el producto final o produzca daños o pérdidas en las herramientas utilizadas.

Existe el caso en el que la experiencia es perecedera, esto es cuando una persona perteneciente a una institución o empresa ha trabajado por años encontrando trucos y obteniendo conocimientos para resolver este tipo de problemas pero por alguna razón se alejan (son despedidos, se retiran, mueren, etc) llevándose con ella toda la información.

Otra justificación y una de las mayores motivaciones para la construcción de un SE, es la complejidad de un problema en un área específica y la escasez de los

expertos necesarios para resolver el problema en cuestión. Al no encontrarse la cantidad necesaria de expertos hace que su experiencia en el área donde se desenvuelven la gran demanda por ellos los hace muy costosos.

La diferente literatura recerca de SE muestra también unos aspectos claves para determinar cuando es apropiado el desarrollo de un SE. Estos aspectos son básicamente tres, el primero de estos es la Naturaleza del problema. Un problema debe poder resolverse mediante la manipulación de símbolos y estructuras de símbolos para ser resuelto por un SE, ya que la habilidad para llevar a cabo un razonamiento mediante el uso de símbolos es algo que caracteriza a los SE.

También se encuentra el aspecto de la complejidad, el problema a resolver por el SE no debe ser muy fácil, debe ser un problema que requiera el dominio de un tema para el cual se requieren muchos años de estudios y experiencias vividas (como en el caso de los expertos de este SE, que son médicos que tras varios años de estudio y de experiencia han logrado dominar el tema) para llegar al nivel de experto

El tercer aspecto es el alcance, cualquier problema que se desee resolver por medio de un SE debe tener un alcance apropiado. Este alcance se debe limitar lo suficiente para que el problema pueda ser manejable, pero debe ser tan amplio que tenga un interés practico.

3.2.5 Representación del conocimiento

La representación del conocimiento básicamente es escribir en algún lenguaje o medio de comunicación, descripciones o cuadros que corresponden en forma casi exacta al mundo o al estado del mundo.⁹

Los esquemas o formas para representar el conocimiento son, las redes semánticas, marcos, scripts, sistemas de producción (reglas).

Las redes semánticas hacen énfasis en la representación grafica de las relaciones entre los elementos de un dominio. Sus componentes básicos son los nodos y los enlaces. Estos nodos representan lo elementos y los enlaces son relaciones que se consideran ciertas de un elemento a otro.

Un marco consiste en una serie de ranuras o slots que contienen la información asociada al objeto, cada una de estas ranuras puede contener diferentes tipos de valores como, valores por defecto, apuntadores a otros marcos y procedimientos por cada valor a obtenerse.

Los guiones son básicamente una especialización del concepto general de un marco, es una estructura utilizada para guardar prototipos de secuencias de sucesos. Definen situaciones perfectamente conocidas y que no sufren

⁹ CORREDOR, Op. Cit. p 26

cambios significativos una vez almacenadas y al igual que los marcos están compuestos por ranuras.

La forma de representar el conocimiento mayormente empleada son los sistemas de producción, estos utilizan reglas para representar el conocimiento. Constan de una porción de memoria que se utiliza para rastrear el estado actual del universo bajo consideración, un conjunto de reglas de producción y un interpretador que examine el estado actual y ejecute las reglas de producción aplicables. Las reglas de producción facilitan la construcción de las bases de conocimiento debido a su versatilidad, facilidad en la representación del conocimiento y permiten utilizar estrategias de búsqueda concretas como el encadenamiento de reglas.

La selección del tipo de representación del conocimiento es algo crucial para el desarrollo de un SE. Los criterios que se deben tener en cuenta para elegir la forma de representación el conocimiento, son los siguientes:

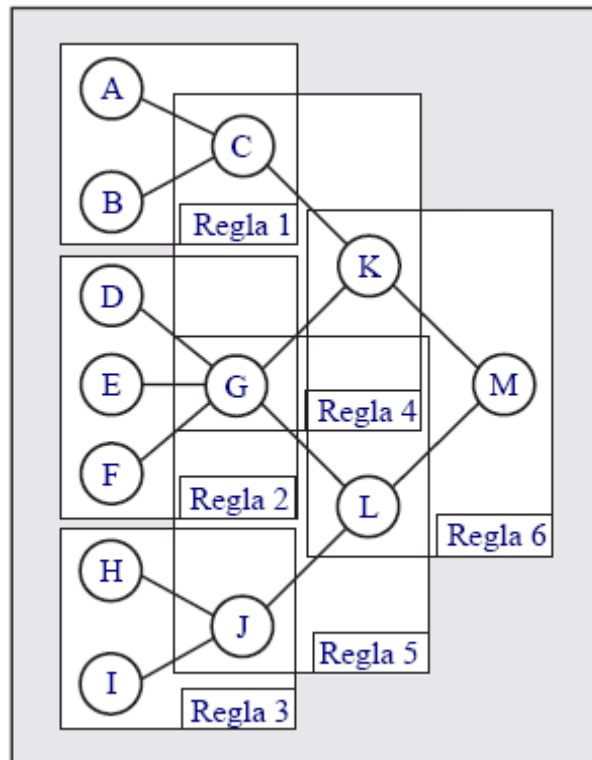
- Transparencia: Hasta que punto podemos identificar fácilmente el conocimiento almacenado.
- Claridad: hasta que punto el conocimiento se puede representar directamente (creación de reglas).
- Naturalidad: Hasta que punto el conocimiento se puede representar en su forma original
- Eficiencia: La facilidad relativa con la cual se puede acceder a conocimientos específicos (reglas y hechos) durante la ejecución.
- Adecuación: Hasta que punto una estructura dada se puede emplear para representar todos los conocimientos que requiere el sistema.
- Modularidad: hasta que punto los fragmentos de conocimientos (en este caso los protocolos), se pueden almacenar independientemente uno de otro, en la base de conocimientos.

3.2.6 Encadenamiento De Reglas

Una de las estrategias de inferencia mas utilizadas para obtener conclusiones compuestas es el llamado encadenamiento de reglas hacia adelante. Esta estrategia puede ser utilizada cuando todo el lado derecho o alguna de sus premisas de la regla coinciden con la conclusión de otra (Figura 2). Cuando se encadenan las reglas, los hechos utilizarse para dar nuevos hechos. Esto se repite sucesivamente hasta que n pueden obtenerse más conclusiones.¹⁰

¹⁰ **CASTILLO** ,Enrique y otros. Sistemas Expertos y Modelos de Redes Probabilísticas. p 39

Figura 2. Encadenamiento de reglas hacia adelante



El encadenamiento de reglas hacia adelante tiene el siguiente procedimiento:

- Se ingresan al motor de inferencia todos los hechos conocidos.
- El motor de inferencia ejecuta todas las reglas que se activaron (gracias a los hechos conocidos) y estas concluyen nuevos hechos que activan nuevas reglas.
- Se repite el paso anterior hasta que no sea posible la generación de nuevos hechos.

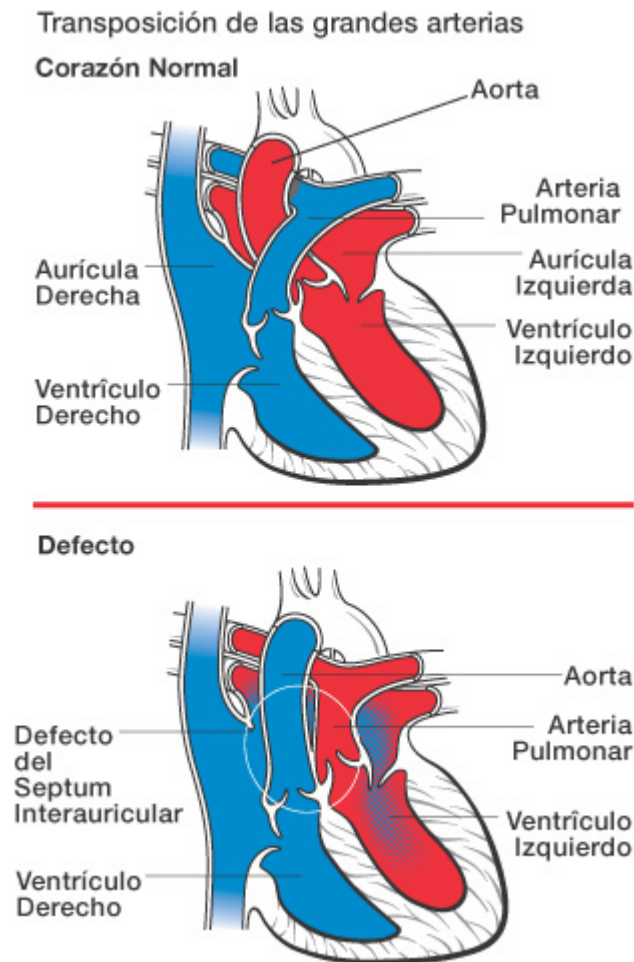
3.3 PATOLOGÍA DE TRANSPOSICIÓN DE GRANDES ARTERIAS.

La transposición de los grandes vasos es un defecto cardíaco congénito. Debido a un desarrollo anormal del corazón del feto durante las primeras ocho semanas de embarazo, los grandes vasos que llevan sangre desde el corazón hacia los pulmones o hacia el cuerpo se conectan inadecuadamente.

Normalmente, la sangre pobre en oxígeno (azul) circula desde el cuerpo hacia la aurícula derecha, pasa por el ventrículo derecho y luego, a través de la arteria pulmonar, se dirige hacia los pulmones donde recibe oxígeno. La sangre

rica en oxígeno (roja) vuelve a la aurícula izquierda desde los pulmones, pasa por el ventrículo izquierdo y, a través de la aorta, se dirige hacia todo el cuerpo.

Figura 3. Transposición de grandes arterias (TGA)



En la transposición de los grandes vasos, la aorta está conectada al ventrículo derecho y la arteria pulmonar, al ventrículo izquierdo; exactamente lo opuesto a la anatomía de un corazón normal, produciendo así que la sangre pobre en oxígeno vuelve a la aurícula derecha desde el cuerpo, pasa por la aurícula y el ventrículo derechos y luego regresa al cuerpo a través de la aorta (que está incorrectamente conectada). La sangre rica en oxígeno (roja) vuelve a la aurícula izquierda desde los pulmones, pasa por la aurícula y el ventrículo izquierdos y luego, a través de la arteria pulmonar, regresa a los pulmones.¹¹

Para el prototipo de sistema experto se decidió empezar por conocer, entender e implementar el protocolo de transposición de grandes vasos, ya que esta cardiopatía es una de las más complejas debido a su manejo postoperatorio y contiene una gran cantidad de los signos a monitorear y de los controles que se encuentran en los protocolos de otras cardiopatías.

¹¹ **Universidad de Virginia**, Health System (on line)
http://www.healthsystem.virginia.edu/UVAHealth/peds_cardiac_sp/tga.cfm

4 DESCRIPCIÓN DEL SISTEMA

4.1 ANÁLISIS DEL SISTEMA

4.1.1 Situación Actual

La Unidad de Cuidados Intensivos (UCI) Pediátrica, es uno de los escenarios de desempeño médicos que dispone de más información que cualquier otro y aunque en su mayor proporción ésta se obtiene electrónicamente (Historia clínica electrónica y monitor de signos vitales), la interpretación de la misma se lleva a cabo de forma manual en la mayoría de las veces, con la subsiguiente pérdida e incorrecta transcripción de los datos obtenidos. En el ambiente de la UCI Pediátrica se manejan gran cantidad de variables que la hacen un lugar particularmente susceptible al error médico, dado la enorme dificultad que representa para el humano el procesar esas grandes cantidades de información, dificultando la capacidad efectiva de análisis y síntesis. Otro problema que existe es la demora en la obtención de registros que alerten sobre anomalías y permitan anticiparse a posibles complicaciones en los pacientes.

Un indicador de calidad en las unidades de cuidados intensivos es la mortalidad acumulada, en los mejores centros del mundo, la mortalidad acumulada es cercana al 2%. En la Fundación Cardiovascular de Colombia (FCV), la tasa de mortalidad acumulada en los últimos 8 años es del 5.8%. Dentro de los análisis internos de mortalidad y de complicaciones, se encontró que el 50% de las muertes fueron producto de complicaciones totalmente prevenibles.

4.1.2 Sistema propuesto

Dada la situación actual, la UCI Pediátrica se perfila como el sitio ideal para la aplicación de una serie de herramientas de inteligencia artificial (IA), que permitan una mayor eficiencia en el cuidado de los pacientes, además al observar el error médico, como algo constante en la práctica clínica, aparece la necesidad de implementar estrategias para lograr su reducción; por esto la FCV plantea el proyecto de Sistema Experto (SE) y para el desarrollo de éste se hace necesario el apoyo de estudiantes de ingeniería de sistemas.

La práctica empresarial busca mejorar todos los conocimientos adquiridos durante la carrera de ingeniería de sistemas e informática y la adquisición de nuevos conocimientos que permitan participar en el desarrollo de un prototipo de un SE que facilite el proceso de diagnóstico de pacientes ubicados en la

UCI pediátrica que han sido sometidos a un procedimiento quirúrgico cardiovascular.

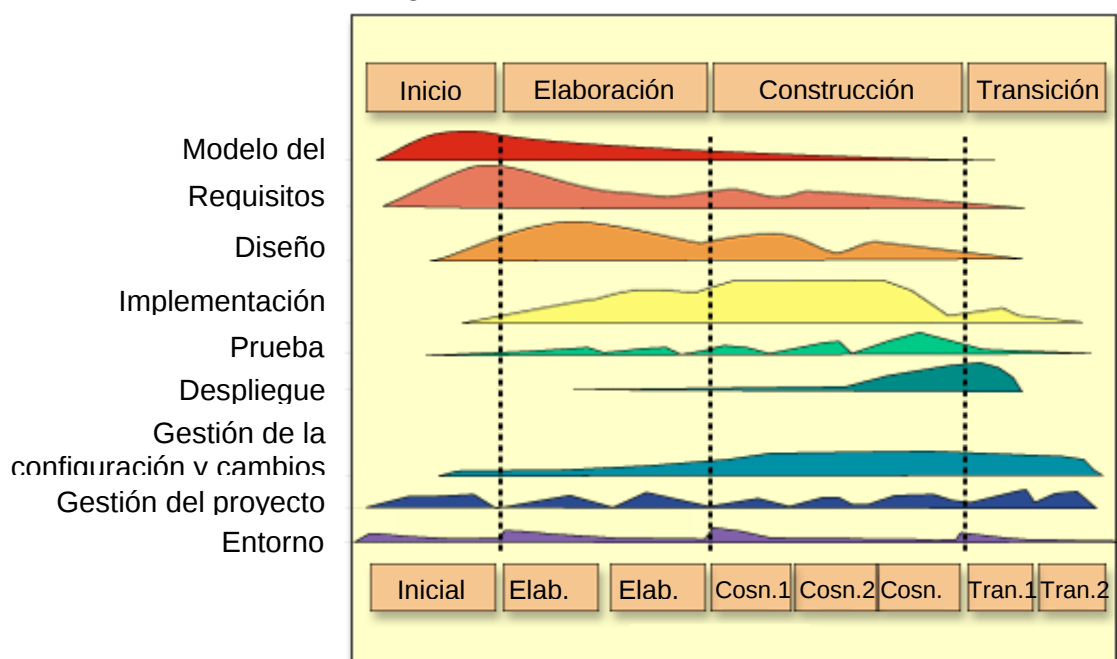
4.2 DESARROLLO DE LA PRÁCTICA

Para el desarrollo de la práctica se eligió la metodología de desarrollo de software llamada Proceso Unificado (PU). Éste utiliza el UML para expresar gráficamente todos los esquemas de un sistema software. Pero, realmente, los aspectos que definen este Proceso Unificado son tres: es iterativo e incremental, dirigido por casos de uso y centrado en la arquitectura.

EL UP se divide en cuatro fases o etapas: Inicio o concepción, elaboración, construcción y transición. Cada una de estas a su vez se divide en iteraciones que van a través varias disciplinas, una disciplina es un conjunto de actividades en un área determinada (modelado del negocio, requisitos, análisis, diseño, implementación, etc). Cada una de estas disciplinas tiene artefactos relacionados, un artefacto es cualquier producto del trabajo realizado: código, diagramas de secuencia, etc.

En la Figura 4 se puede apreciar que durante una iteración, el trabajo se desarrolla por la mayoría de las disciplinas. Sin embargo el esfuerzo en cada una de ellas cambia con el tiempo. En las primeras iteraciones, se tiende a aplicar un esfuerzo mayor a los requisitos y al diseño; y en las posteriores disminuye cuando los requisitos y el diseño se estabilizan por medio de un proceso de adaptación y realimentación.

Figura 4. Proceso Unificado



Para el desarrollo del proyecto SE, la FCV lo dividió en varios ciclos, obteniéndose así en cada uno de estos ciclos, un prototipo evaluable que cumple parcialmente con los requisitos del proyecto. A cada uno de estos ciclos se les aplica la metodología del PU, desarrollándose a lo largo de las disciplinas (flujos de trabajo) que se consideraron necesarias, abarcando cada una de las diferentes fases del PU. Durante la práctica empresarial se abordó el primer ciclo, del cual se obtuvo un prototipo evaluable; también se abordó el inicio del segundo ciclo del cual se alcanzó a implementar partes de la primera pantalla de la versión web (segundo prototipo evaluable) del SE.

En la figura 5 se pueden apreciar las principales actividades que se realizaron a través de la práctica empresarial, enmarcadas en la metodología PU. También se puede observar cada una de las fases (inicio, elaboración, construcción y transición) junto con la cantidad de iteraciones que duró cada una de estas; cada iteración duró alrededor de 3 semanas.

Las actividades se clasificaron dentro de cada una de las disciplinas definidas por el UP. La intensidad del color indica en que momento el esfuerzo fue mayor en la actividad a través de cada una de las iteraciones.

Figura 5. Principales actividades realizadas

PRIMER CICLO								
		FASES						
DISCIPLINAS	PRINCIPALES TAREAS	Inicio		Elaboración		Construcción		Transición
		Iteración 1	Iteración 2	Iteración 3	Iteración 4	Iteración 5	Iteración 6	Iteración 7
Mdelado del negocio	Casos de uso del negocio							
	Diagramas de casos de uso del negocio							
	Glosario Médico							
	Diagramas de actividades							
Definición del Producto	Documentación sobre SE							
	Aprendizaje acerca del motor de inferencia CLIPS							
	Transposición de grandes vasos TGA							
Elicitar requerimientos	Listado de requerimientos críticos del sistema							
	Requerimientos del modulo de administracion de reglas							
Diseño	Jerarquización del contenido							
	Casos de uso del sistema							
	Diagramas de secuencia del sistema							
	Interfaces Graficas							
	Base de datos de Reglas y Usuarios del sistema							
Implementación	Reglas de TGA en CLIPS							
	Primer prototipo -DOS							
	Conexión Clips con JAVA							
	Primer prototipo gráfico							
	Copias de seguridad							
	Desarrollo de administración de reglas							
	Integración de los Módulos							
	Perfiles de usuarios del sistema							
Pruebas	De módulo independiente							
	De módulos integrados							
	De rendimiento del sistema							
Gestión del proyecto	Definición del estándar de programación							
	Estructura de árbo de directorios y adm. De versiones							

A continuación se describe cada una de las actividades realizadas durante la práctica empresarial, en cada una de las diferentes fases del PU.

4.3 FASE DE INICIO

El propósito principal de esta fase es definir y acordar el alcance del proyecto con los clientes o personas interesadas en el proyecto, identificar los riesgos asociados al proyecto y proponer una visión muy general de la arquitectura de software.

4.3.1 Modelado del negocio

La finalidad del modelado del negocio es describir cada proceso del negocio, especificando sus datos, actividades o tareas, roles o agentes y reglas de negocio.

En desarrollo de software denominamos negocio a la empresa o al área de la empresa a la que le queremos dar algún tipo de solución informática. Para el caso de la presente práctica el negocio es la UCI (Unidad de Cuidados Intensivos) pediátrica, en la cual son recibidos y se le presta atención a pacientes pediátricos (niños) que se encuentran en un estado delicado de salud después de una cirugía.

4.3.1.1 Casos de uso del modelado del negocio

Después de leer los diferentes manuales de protocolos y procesos de la UCI pediátrica se puede tener una idea del negocio para a continuación, realizar el modelado del negocio, identificando los principales actores, realizar y describir cada caso de uso, hacer el diagrama de casos de uso, diagramas de actividades y los diagramas de secuencia correspondientes a cada uno de estos. [Anexo A]

4.3.2 Motor de inferencia CLIPS

El motor de inferencia es una de las partes primordiales del SE, por esta razón era necesario, conocer el motor de inferencia CLIPS, buscando conocer sus principales características.

4.3.2.1 Que es clips?

CLIPS (C Language Integrated Production System), es una herramienta en el año de 1984 por la STB (Software Technology Branch) para el desarrollo de SE.

Esta herramienta fue diseñada para desarrollar software que modele el conocimiento humano (conocimiento que poseen personas que son consideradas expertas en su área), con unos propósitos específicos como alta portabilidad, bajo coste (es libre) y facilidad de integración. CLIPS permite ser integrado con diferentes lenguajes de programación como C, ADA¹², Visual Basic 6.0, C# y Java entre otros. La integración puede ser de dos maneras, que CLIPS sea llamado desde un programa hecho en alguno de estos lenguajes procedurales, realizando su función y devolviendo el control al programa que lo llamo; o también se puede definir el código del programa como una función externa que es llamada desde CLIPS y cuando finaliza su ejecución esta devuelve el control a CLIPS.

CLIPS es un entorno completo para la construcción de SE basados en reglas y/o objetos¹³. La versión de CLIPS para Windows dispone de un entorno de desarrollo tipo consola u orientado a texto, posee una herramienta de depuración, un editor de texto integrado, un manual y ayuda on-line. Existen también versiones para diferentes plataformas como Macintosh, Windows 3.1 y X Window.

Clips al estar basado en C es case sensitive, es decir distingue entre las mayúsculas y las minúsculas.

4.3.2.2 Representación del conocimiento en CLIPS

Para la representación del conocimiento, CLIPS ofrece paradigmas heurísticos (reglas) y procedurales (funciones y objetos).

Conocimiento heurístico o reglas: Se utilizan reglas para representar heurísticos que plantean un conjunto de acciones que deben realizarse al presentarse una situación dada. La persona encargada de la elaboración del SE crea una colección de reglas que resuelven un problema, aunque estas reglas en su estructura parecen sentencias SI -ENTONCES como las de C u otros lenguajes, actúan en su mayoría como sentencias SIEMPRE QUE – ENTONCES.

Conocimiento procedural o funciones y objetos: Es el conocimiento representado en funciones definidas por el programador, funciones genéricas y por la POO (Programación Orientada a Objetos). En CLIPS se pueden manejar las características principales de la POO como lo son: clases, mensajes, abstracción, encapsulamiento, herencia y polimorfismo.

¹² Lenguaje de programación estructurado, que permite varias características orientadas a objetos. Es el lenguaje empleado por el Departamento de Defensa de los Estados Unidos.

¹³ **SÁNCHEZ**, Aitor San Juan. Sistema CLIPS (Apuntes).

4.3.2.3 Elementos esenciales en CLIPS

El shell de CLIPS posee los elementos esenciales de todo SE, una memoria global de datos o MT (Memoria de Trabajo), una base de conocimiento y el motor de inferencia.

MT (Memoria De Trabajo): Los elementos que se encuentran en la MT son entidades que pueden corresponder a hechos o instancias de una clase. La MT posee una lista de hechos (fact-list) y una lista de instancias (instante-list). Un hecho (fact) es una forma básica de alto nivel para representar información y es la unidad fundamental de datos utilizados en las reglas. Los hechos son compuestos por campos, un campo es un lugar que puede o no tener un nombre y lleva asociado un valor.

Base de conocimiento: La base de conocimiento es un depósito de las primitivas de conocimiento, que están disponibles para el sistema como por ejemplo las reglas y los hechos fundamentales. Como la potencia de un experto se debe más a al conocimiento amplio del área específica que a la comprensión del desempeño genérico de un experto, el conocimiento guardado en la base de conocimiento, establece la capacidad del SE para actuar como un experto.

Reglas: Una regla contiene un antecedente, llamado también Lado Izquierdo (LI) o Left Hand Side (LHS), que consiste de una serie de elementos condición o "SI's" que describen las diferentes condiciones que se deben cumplir para que la regla sea aplicada. Cuando se cumplen todas las condiciones del LI y la regla no se ha ejecutado, se puede decir que la regla esta activa o es aplicable. Las reglas también contienen un consecuente, o mejor dicho una parte de acción, la cual es llamada como Lado Derecho (LD) o Right Hand Side (RHS), en este lado de la regla es donde se especifican todas las acciones a realizar cuando la regla se dispara (se cumple con las condiciones del LI). Entra las acciones que se pueden llevar a cabo, las mas comunes son las de agregar elementos o hechos a la memoria de trabajo (MT), eliminar hechos de la MT, modificar atributos de los diferentes elementos de la MT, realizar operaciones a partir de variables especificadas, aceptar datos por parte del usuario, enviar mensajes al usuario y el llamado a procedimientos definidos por el usuario ya sea en CLIPS o en otro lenguaje con el que se haya integrado CLIPS.

Sintaxis de una regla en CLIPS:

```
(defrule <nombre-regla> [<comentario>][<declaración>]
<elemento-condición>* ;;Hasta aquí va la parte izq de la regla
=>
<acción>*) ;;Esta es la parte derecha de la regla
```

Motor de Inferencia: El motor de inferencia es el que decide que reglas deben ser ejecutadas y cuando. Las acciones que están en el lado derecho de las reglas que son aplicables son realizadas cuando al motor de inferencia se le es indicado que las ejecute. Si es posible la ejecución de más de una regla, el

motor de inferencia usa una de las estrategias de resolución de conflictos para decidir cual de las reglas es la que debe realizar sus acciones. Las acciones de la regla seleccionada se ejecutan, lo cual puede afectar la lista de las reglas aplicables, luego el motor de inferencia selecciona la otra regla y ejecuta sus acciones. Este proceso continua mientras existan reglas aplicables.

4.3.2.4 Estrategias de resolución de conflictos

La agenda es la lista de todas las reglas cuyas condiciones se cumplen pero no se han ejecutado. La agenda se puede comparar con una pila, donde la ultima regla que entra a la agenda, es la primera en ejecutarse. Cuando una nueva regla es activada, esta se ubica en la agenda de acuerdo a ciertas condiciones, primero la regla es ubicada por encima de las reglas de menor prioridad y debajo de las de mayor prioridad, después de entre la reglas de igual prioridad se utiliza la estrategia de resolución de conflictos que se encuentre activa en CLIPS en ese momento para determinar su ubicación. Cuando los pasos anteriores no son suficientes para su ubicación, la regla se colocara de manera arbitraria.

CLIPS posee 7 estrategias de resolución de conflictos: *depth* (estrategia por defecto en CLIPS), *breadth*, *simplicity*, *complexity*, *lex*, *mea*, y *random*, cada una de estas estrategias tiene un criterio diferente para resolver los conflictos, por ejemplo, *depth* ubica las nuevas reglas activadas encima de todas las reglas de igual prioridad.

4.3.2.5 Diseño de módulos en CLIPS

CLIPS provee el soporte necesario para el desarrollo y la ejecución modular de bases de conocimiento con el constructor *defmodule*. Los módulos permiten agrupar un conjunto de constructores (*defrule*, *deftemplate*, *defclass*, etc) de modo que se mantenga un control explícito sobre la restricción del acceso de otros módulos a estos constructores.

Los módulos se definen mediante el uso del constructor *defmodule* .

Sintaxis en CLIPS:

```
(defmodule <nombre-modulo> [<comentario>]
<port-especificación>*)
```

```
<port-especificación> ::= (export <port-item>) | (import <nombre-modulo> <port-
item>)
```

```
<port-item> ::= ?ALL | ?NONE | <port-constructor> ?ALL |
<port-constructor> ?NONE |
<port-constructor> <nombre-constructor>+
<port-constructor> ::= deftemplate | defclass | defglobal | deffunction |
defgeneric
```

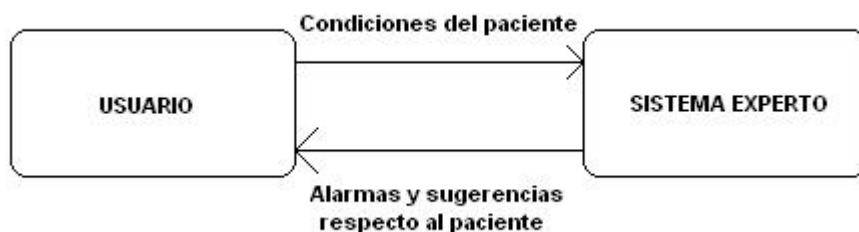
4.3.3 Conocimiento transformado en reglas

Para el prototipo de SE se decidió empezar por conocer y entender que es la transposición de grandes vasos y el protocolo de manejo, ya que esta cardiopatía es una de las mas complejas y contiene la mayoría del monitoreo de signos y los controles que se encuentran en los protocolos de otras cardiopatías.

Una vez que se entendió de qué se trata la cardiopatía de transposición de grandes arterias y el protocolo de manejo de ésta, se procedió a pasar el conocimiento de los expertos a reglas en CLIPS, para así realizar un primer programa. Este programa es algo sencillo, se inicia cargando en la base de conocimiento las reglas que se encuentran en un archivo con extensión clp, entre las reglas existe una que es activada con el initial-fact, que es el hecho que se produce en el momento que ejecutamos la instrucción reset en CLIPS. Una vez utilizado el comando run, esta regla se activa y se ejecuta generando unos hechos que activan otras reglas que le pedirán la información necesaria al usuario del SE y esta información será agregada en forma de hechos a la memoria de trabajo y activara las demás reglas dando las alarmas y sugerencia necesaria respecto al paciente.

El programa básicamente funciona como se muestra en la Figura 6.

Figura 6. Funcionamiento del primer programa en CLIPS



La ejecución del programa en CLIPS mostraría algo como lo siguiente:

Ingrese el valor de la presión sistólica arterial sistémica

34

Ingrese el valor de la presión arterial media

23

Ingrese el valor de la presión venosa central

54

Resultado:

ALARMA, la presión sistólica arterial sistémica esta por debajo de la meta establecida, con un valor de: 34

La presión arterial media esta por encima de la meta establecida, con un valor de: 23

La presión venosa central esta por encima de la meta establecida, con un valor de: 54

Como se observa en esta ejecución del programa en CLIPS, se piden todos los datos necesarios al usuario y después de esto ejecuta las reglas que se activaron para darle las correspondientes sugerencias y alarmas para que estas sean revisadas y corroboradas en el paciente y poder hacer las correcciones necesarias en el tratamiento del paciente para que este tenga una pronta recuperación.

4.3.4 Glosario

Con el fin de entender la diferente y variada terminología utilizada en el día a día de nuestros expertos (que para el caso de este prototipo de sistema experto son los médicos) y en los diferentes manuales de protocolos y procesos de la UCI Pediátrica, se elaboro un glosario.

Este glosario contiene diferentes palabras y abreviaturas que se fueron encontrando a medida que se leían los diferentes manuales y que se dialogaba con los médicos participantes de este proyecto. [Anexo 5]

4.3.5 Estándar de desarrollo

Con el fin de tener un mejor control y entendimiento entre los participantes del proyecto, se decidió elaborar un estándar de desarrollo en el cual se especifica las diferentes reglas a seguir en el momento de elaborar algún elemento (documento, archivo java, etc) perteneciente al desarrollo del proyecto.

En este estándar se incluye una jerarquía de directorios, la cual especifica los diferentes directorios que deben ser creados y que elemento debe ir en cada uno de estos. Se decidió que estos directorios deben ser creados en el equipo de trabajo de cada uno de los participantes, para que estos mantengan ahí sus versiones de los diferentes elementos y tambien deben ser creados en un equipo aparte donde se mantendrán ahí las versiones ya aprobadas por cada uno de los participantes.

El estándar también incluye la nomenclatura a utilizar al programar, las abreviaturas y lo que se debe tener en cuenta en le momento de darle el nombre a una clase, una función, una variable, etc. [Anexo 6]

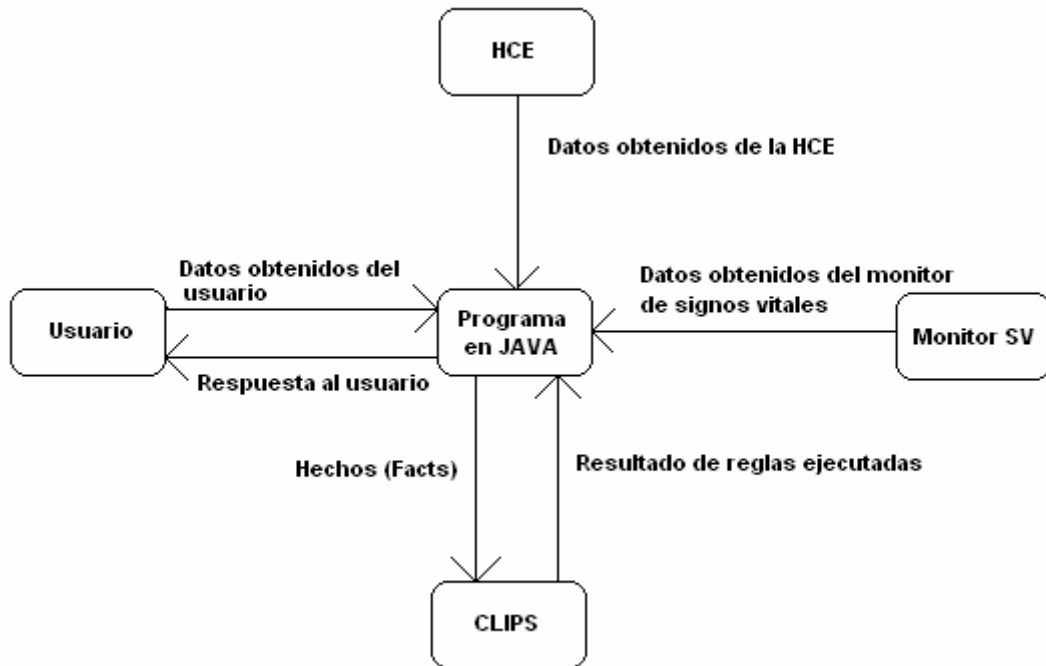
4.3.6 Prototipo DOS

El lenguaje escogido para llevar a cabo la implementación del prototipo DOS de sistema experto, es el lenguaje de programación java, este es un lenguaje orientado a objetos, independiente de la plataforma y puede ejecutarse en cualquier plataforma sin hacer cambios.

Para su correcto funcionamiento, el SE debe obtener la información de diferentes lugares, uno de estos es la HCE (Historia Clínica Electrónica), en la cual son registrados periódicamente datos relevantes del paciente (como el potasio, el calcio, la saturación venosa mixta, etc), también se toman datos del

monitor de signos vitales, el cual, monitorea los signos del paciente en tiempo real y por ultimo, tomamos información del usuario. Todos estos datos son necesarios para la ejecución de las diferentes reglas obtenidas de pasar la experiencia de los expertos (los médicos en este caso) al lenguaje de CLIPS.

Figura 7. Obtención de Hechos



Como se puede apreciar en la figura 7, el programa escrito en java toma datos que son obtenidos del monitor de signos vitales, de la historia clínica y del usuario. El programa en java enviara esta información en forma de hechos a CLIPS y este le devuelve los resultados que arrojaron las reglas que se activaron y ejecutaron, estas respuesta serán mostradas por el programa escrito en java al usuario en un lenguaje comprensible para el.

Para llevar a cabo el prototipo DOS fue necesario integrar CLIPS con java, para esto, después de una investigación se encontró la existencia de JClips¹⁴ (Clips for Java). Pero, ¿que es JClips?, JClips es una pieza de software o puente (bridge) que permite a los programadores en java, usar CLIPS en combinación con java, integrando el motor CLIPS en aplicaciones java. Con JClips, es posible controlar directamente el motor de inferencia de CLIPS, se pueden cargar reglas, ejecutar los diferentes comandos, correr CLIPS y recibir las respuestas junto con toda la información que este produzca en forma de mensajes tipo *String*.

Para integrar CLIPS con java, se necesita primero la librería compartida de CLIPS, la cual se encuentra en un .dll precompilado para Windows o se

¹⁴ JClips es un software libre escrito por Maarten R. Menken, M.Sc. en Computer Science. Maarten R. Meneen pertenece al Departamento de Inteligencia Artificial de Ámsterdam.

encuentra el código fuente original de CLIPS para ser compilado en cualquier compilador de C. Se necesita también la librería en java de JClips que se encuentra también precompilada en un .jar que puede ser usado en cualquier sistema operativo. El .jar debe ser ubicado en el CLASSPATH.

En CLIPS los mensajes son enviados hacia java usando una función llamada *send-to-java*, la cual toma como argumento la cadena de caracteres que se quiere enviar a la aplicación java y debe ser ejecutada por una regla. La sintaxis es de la siguiente manera:

```
(defrule envio-mensaje
(Respuesta ?string-respuesta)
=>
(send-to-java ?string-respuesta)
)
```

Una vez se tenga el .dll y el .jar lo siguiente es crear una clase en java donde se vaya a controlar CLIPS, esta clase debe implementar la interfaz nativa de java llamada *Observer* que se encuentra en la librería java.util.Observable, y el método de esta interfaz llamado *update*, el cual es el encargado de recibir el mensaje tipo *String* enviado desde CLIPS.

También debemos declarar un objeto o una instancia de la clase JClips con la cual ejecutaremos los diferentes comandos y funciones de CLIPS.

Otro requisito necesario para la creación del primer prototipo es el acceso a la base de datos de la HCE desde java. Java como la mayoría de lenguajes de programación tiene su forma de establecer una conexión. En java se utiliza JDBC, para esto primero cargamos los drivers a utilizar (los cuales son diferentes según la base de datos), por medio de la instrucción *Class.forName("Driver a usar")*. Después se establece la conexión con una variable tipo *Connection*, luego se instancia una variable tipo *Statement* con la cual ejecutaremos las diferentes instrucciones SQL y si alguna de estas instrucciones devuelve valores (como la instrucción *select * from tabla*), éstos serán guardados en una variable tipo *ResultSet* para su posterior utilización.

Los datos generados por el monitor de signos vitales se encuentran en una interfaz, que en este caso es un archivo de texto plano con un formato especial, el cual contiene que signos se están monitoreando en ese momento y el valor de estos. Para esto es necesario la lectura de este archivo plano desde java, para lo cual java posee clases nativas especiales para la lectura y escritura de archivos. Con estos elementos fue posible pasar a la construcción del prototipo DOS de sistema experto.

Al estar elaborando este primer prototipo evaluable se pudo ver que el sistema no sirve de nada si no se esta ejecutando continuamente, ya que este debe estar en funcionamiento durante toda la estancia del paciente en la UCI Pediátrica y además hay datos como los de la HCE que son agregados a esta

cada cierto tiempo y no deben estarse generando hechos con estos datos sino cada vez que estos cambien.

Después de una profunda investigación y de múltiples sugerencias de cómo hacer para que el programa se mantuviera en ejecución se decidió probar con hilos (Threads), debido a que éstos pueden pasar a un estado en el cual pueden permanecer inactivos por el tiempo que sea necesario y luego ejecutarse paralelamente a otros procesos.

Un hilo de ejecución es similar a un proceso en que ambos representan una secuencia simple de instrucciones ejecutada en paralelo con otras secuencias. Los hilos permiten dividir un programa en dos o más tareas que corren simultáneamente, por medio de la multiprogramación.

El prototipo DOS de sistema experto tiene básicamente lo expuesto en la figura 7, se trae la información necesaria para enviar en forma de hechos a CLIPS y la información que este genere se devuelve en un lenguaje entendible para el usuario. También existen dos hilos, uno que realizara las preguntas necesarias acerca del electrocardiograma, estas preguntas se realizan cada vez que exista un cambio de turno medico, por esta razón se le pide al usuario la fecha y hora de finalización de su turno para así poder bloquear este hilo durante el tiempo necesario y así en el momento que éste vuelva a activarse, nuevamente pedir el cambio de turno y realizar de nuevo las preguntas. El otro hilo es el encargado de extraer la información del monitor de signos vitales y la HCE que también se bloqueara por ciertos periodos de tiempo.

La siguiente figura muestra la ejecución del prototipo DOS del sistema experto.

Figura 8. Prototipo DOS

```
*** PROTOTIPO SISTEMA EXPERTO APU 1.0 ***

Ingrese la atencion del paciente: 144234

*** INFORMACION DEL PACIENTE ***

NOMBRE: OSCAR JULIAN MOLINA HIGGINS

CIRUGIAS A LAS QUE HA SIDO SOMETIDA:

* TRANSPOSICION ARTERIAL CON REIMPLANTE DE CORONARIAS [JATENE]
* REPARACION DE DEFECTO DE TABIQUE INTERVENTRICULAR CON PROTESIS
* COLOCACION DE CATETER PARA DIALISIS PERITONEAL

*** CARGANDO INFORMACION DEL TURNO ***

Ingrese la Fecha y hora de finalizacion de su turno en el siguiente
formato (yyyy/mm/dd h:min am-pm): 2006/04/08 7:00 am

Fin de turno: *** 2006/04/08 7:00 am ***

Confirmar la fecha programada para el fin de turno (si/no)? : si

Hay extrasistole (si/no)?: si
|
El segmento ST esta elevado (si/no)?: si

La onda T es bifasica (si/no)?: si

Se observa la onda T de manera tardia (si/no)?: si

Esta presente la onda Q (si/no)?: si

Es ancha la onda Q (si/no)?: si

Es profunda la onda Q? (si/no): si

Es el segmento PR corto o largo (si/no)?: si

* SE HA PRESENTADO INSUFICIENCIA CORONARIA
* REMITASE A PROTOCOLO DE ARRITMIAS

Hay bajo gasto cardiaco (si/no)?: si

El paciente esta hemodinamicamente estable (si/no)?: si

CONSULTA MONITOR DE SIGNOS VITALES RECIBIDO A LAS 09:41:00 ***
□
La presion venosa central esta por encima de la meta establecida(10),
actualmente tiene un valor de:80.0mmhg
```

4.3.7 Pruebas al prototipo

En esta fase se realizaron las primeras pruebas de rendimiento al sistema experto, además de esto el sistema se presentó a un conjunto de usuarios para que ellos miraran el funcionamiento del sistema hasta el momento y nos hicieran sus comentarios y aportes.

4.4 FASE DE ELABORACIÓN

Esta fase permite la resolución de los riesgos altos, identificación de mas requisitos y su alcance con estimaciones mas realistas que en la etapa anterior.

4.4.1 Elicitar Requerimientos

El proceso de elicitar requerimientos consiste en todo el proceso de obtención de requisitos, se divide en varias etapas: La primera parte es obtener información sobre el dominio del problema, en esta parte se conoce el dominio o la situación actual del problema; la siguiente etapa es la identificación de los diferentes tipos de requerimientos de los sistemas (funcionales, no funcionales, de manejo de información, etc.); y la ultima parte es priorizar los requisitos del sistema, que consiste en ver cuales son los requerimientos mas importantes y que deben ser desarrollados mas prontamente.

Después de haber realizado el proceso de elicitar requerimientos, se obtuvieron como los más prioritarios los siguientes requisitos:

Requerimientos funcionales:

- Monitorear Signos de Monitor: El sistema debe tomar el valor de los signos monitoreados en tiempo real del archivo plano proveniente del monitor de signos vitales cada 5 minutos.
- Monitorear Signos de la HCE: El sistema debe tomar la información de los signos registrados en la historia clínica electrónica cada 30 minutos.
- Identificar cambio de turno: El sistema debe estar en la capacidad de identificar el inicio y la finalización de los turnos.
- Identificar al paciente: EL sistema debe identificar al paciente por medio del código de la atención, suministrada por el usuario y con esta saber el nombre del paciente y las cirugías a las que ha sido sometido el paciente.

- Preguntas de EKG: El sistema debe hacer que el usuario conteste unas preguntas relacionadas con el electro cardiograma.
- Ausencia de variables y signos: El sistema debe avisar al usuario cuando no encuentre registros de algún signo de la HCE.

Requerimientos no funcionales:

- El sistema debe presentar una interfaz para que el usuario pueda introducir la atención del paciente.
- El sistema debe permitir al usuario pueda ver los datos y las cirugías del paciente; y que también pueda modificar desde allí los valores de las presiones y temperatura.
- Debe existir una interfaz para la modificación de la temperatura.
- Debe existir una interfaz para la modificación de las diferentes presiones.
- Una interfaz donde el usuario pueda visualizar las diferentes alarmas y sugerencias.
- Una interfaz para que el usuario introduzca la hora y fecha para el fin de turno

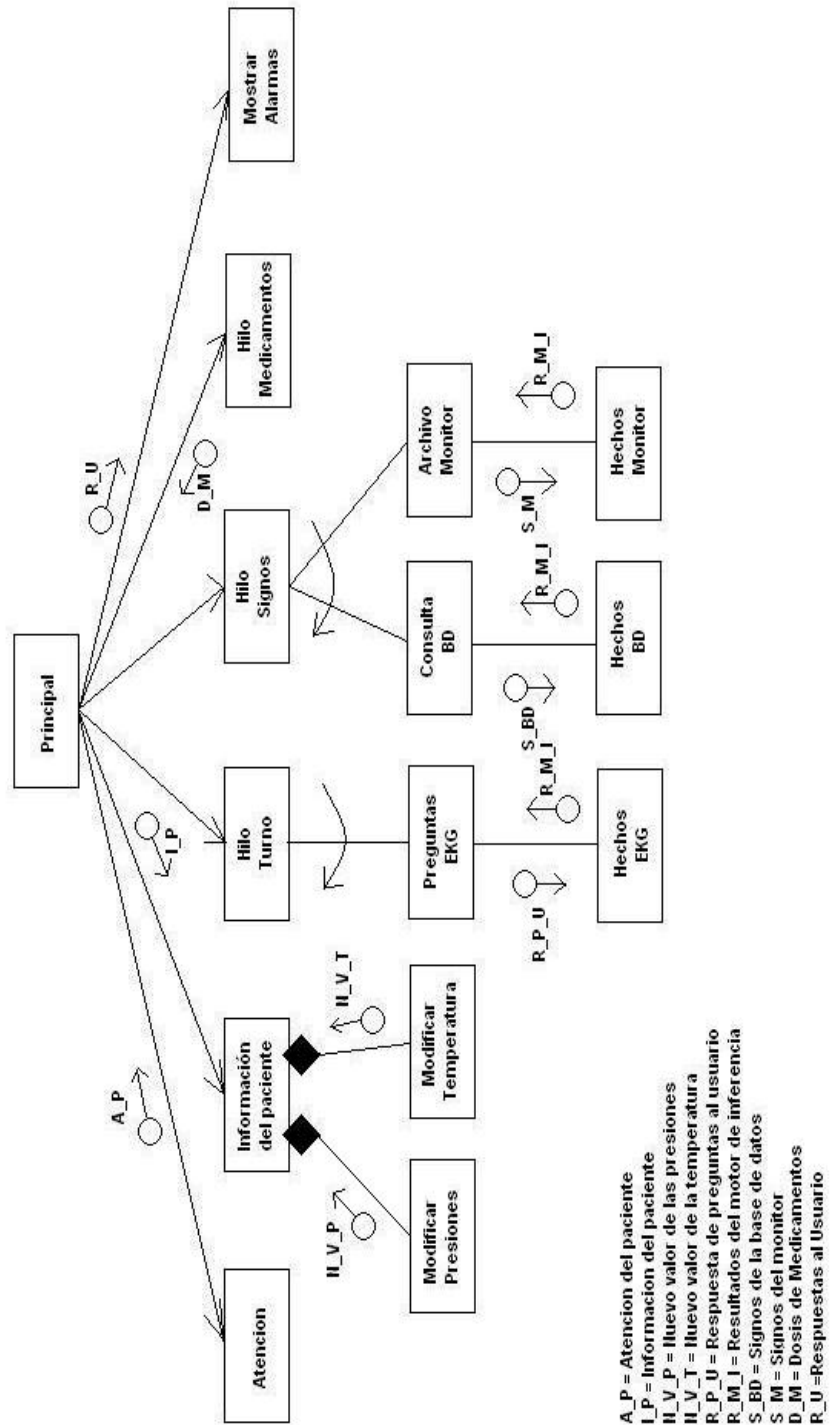
4.4.2 Casos de uso y diagramas de secuencia de interacción del usuario con el sistema

Para una mejor comprensión de cómo el sistema debe interactuar con el usuario y poder diseñar la interfaz con el usuario (diálogos y formularios) y hacer una correcta integración de todos los componentes, se realizó el diagrama de casos de uso de interacción con el usuario y los diagramas de secuencias correspondientes. [Anexo 2]

4.4.3 Jerarquización del contenido

Después de haber realizado una clasificación del contenido, haberle dado una jerarquía, haber identificado todos los procesos necesarios y haber estructurado la información que maneja cada una de las diferentes clases hechas en java del sistema experto, se llegó al siguiente diagrama de estructuras:

Figura 9. Diagrama de estructuras.



En Métrica¹⁵ este diagrama se le denomina diagrama de cuadros de Constantine. El orden de las llamadas a las clases en un diagrama de estructuras es de arriba abajo y de izquierda a derecha.

¹⁵ Metodología de planificación y desarrollo diseñada por un grupo de trabajo constituido al efecto por personal procedente de distintos Ministerios de España.

Los elementos del diagrama de estructuras son los siguientes:

Los módulos que son representados por los rectángulos que se pueden apreciar en la figura, las conexiones entre módulos, la comunicación entre módulos, estructuras repetitivas (representadas por una flecha curva) y las estructuras alternativas (representadas por rombos negros).

Módulos: Una definición muy coloquial de modulo es que un modulo es la unidad mas pequeña de código que puede ser compilada independientemente.

Conexión entre módulos: Un sistema esta compuesto por módulos organizados jerárquicamente, cooperando y comunicándose entre si para realizar un conjunto de tareas. Esta conexión se representa con una flecha como se aprecia en la figura 9.

Comunicación entre módulos: La comunicación entre los módulos se realiza a través de los datos y los flags. Los datos se procesan, mientras que los flags solo sirven para comunicar condiciones entre módulos. Los datos y los flags se representan por medio de un círculo con una flecha, el sentido de la flecha indica el sentido de los datos o los flags y el círculo es negro cuando se trata de un flag.

La siguiente es una breve descripción del contenido, de los procesos y de la información de cada una de las clases:

Principal: Clase encargada de correr el sistema. Esta clase recibe el número de atención del paciente, con esto es identificado el paciente con toda su información correspondiente la cual es enviada hacia otra clase. También envía la información proveniente de las respuestas del motor de inferencia que debe ser mostrada al usuario hacia la clase que se encarga de esto.

Atención: Esta clase se encarga de recibir del usuario la atención del paciente.

Información Paciente: Muestra al usuario la información correspondiente al paciente y permite modificar las presiones y la temperatura.

Modificar Presiones: Sirve para modificar los valores por defecto para las diferentes presiones en el protocolo de TGA. Esto es opcional, puede ser ejecutado o no por el usuario.

Modificar Temperatura: Modifica el valor por defecto de la temperatura. Es opcional para el usuario.

Hilo Turno: Hilo que se ejecuta (pasa a estado activo) cada vez que se cumple el tiempo de un turno. Produce que se hagan las preguntas correspondientes al electrocardiograma cada vez que el hilo este activo.

Preguntas EKG: Realiza al usuario las preguntas correspondientes al electrocardiograma (EKG).

Hechos EKG: Recibe las respuestas a las preguntas del EKG, se las envía a CLIPS (quien realiza la inferencia y devuelve las respuestas de las reglas) y respuestas que el motor de inferencia envíe, luego son enviadas hacia el usuario.

Hilo Monitorear: Encargado de controlar la consulta a la base de datos y de leer el archivo del monitor de signos vitales.

Consulta DB: Se encarga de realizar la consulta de los diferentes signos que se encuentran en la base de datos de la historia clínica electrónica.

Hechos BD: Recibe los diferentes signos de la base de datos, los envía al motor de inferencia en forma de hechos, recibe las respuestas del motor de inferencia y los envía al sistema.

Archivo Monitor: Lee los signos (tomados en tiempo real) del archivo plano proveniente del monitor de signos vitales.

Hechos Monitor: recibe los datos del archivo del monitor y envía los hechos a CLIPS. Devuelve las respuestas del motor de inferencia.

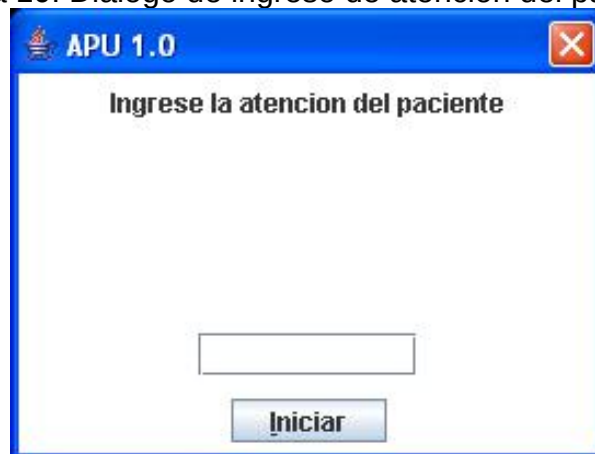
Hilo Medicamentos: Encargado de sugerir los medicamentos y tratamientos cada determinado tiempo. Envía las sugerencias para que sean vistas por el usuario.

Mostrar Alarmas: Muestra al usuario todas las alarmas, sugerencias y dosis de medicamentos al usuario en una forma comprensible y clara para el.

4.4.3 Diseño e implementación de interfaces

Unas de las características de un sistema experto son las facilidades de interfaz con el usuario, es sistema debe aceptar información del usuario y traducirla a una forma aceptable para el resto del sistema (hechos, reglas, parámetros, etc) y también debe aceptar información proveniente del sistema experto (respuestas de reglas ejecutadas, información acerca de hechos faltantes, etc) y convertirla a un lenguaje lo mas natural posible para el usuario. Para esto se diseñaron e implementaron 7 interfaces graficas, las cuales se muestra a continuación.

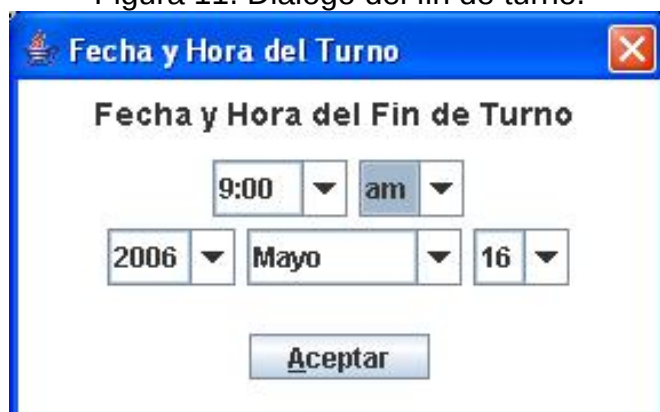
Figura 10. Dialogo de ingreso de atención del paciente.



The screenshot shows a Windows-style dialog box titled 'APU 1.0' with a red 'X' button in the top right corner. The main text inside the dialog is 'Ingrese la atencion del paciente'. Below this text is a single-line text input field. At the bottom center of the dialog is a button labeled 'Iniciar'.

En este dialogo el usuario digita la atención correspondiente al paciente para su respectiva identificación en la base da datos. Esta atención se utiliza en todo el sistema, se utiliza para conocer las diferentes intervenciones a las que fue sometido el paciente y para buscar los signos correspondientes al paciente en la base de datos de la historia clínica electrónica.

Figura 11. Dialogo del fin de turno.



The screenshot shows a Windows-style dialog box titled 'Fecha y Hora del Turno' with a red 'X' button in the top right corner. The main text inside the dialog is 'Fecha y Hora del Fin de Turno'. Below this text are two rows of dropdown menus. The first row contains a time dropdown set to '9:00' and a period dropdown set to 'am'. The second row contains a year dropdown set to '2006', a month dropdown set to 'Mayo', and a day dropdown set to '16'. At the bottom center of the dialog is a button labeled 'Aceptar'.

En este dialogo el usuario introduce la hora y la fecha en la que terminara su turno para bloquear el hilo correspondiente por ese determinado tiempo.

Figura 12. Dialogo de preguntas del EKG

Electrocardiograma

ELECTROCARDIOGRAMA

Hay extrasistole? ▼

El segmento ST esta elevado? ▼

La onda T es bifasica? ▼

Se observa la onda T de manera tardia? ▼

Es el segmento PR corto o largo? ▼

Esta presente la onda Q? ▼

Profunda ☐ Ancha ☐

Este dialogo que aparece cada vez que se inicie un nuevo turno, le pide al usuario información correspondiente al electrocardiograma del paciente, en forma de preguntas de si o no (excepto en la pregunta de la onda Q, ya que si esta existe el usuario debe indicar si es Profunda y/o Ancha).

Figura 13. Frame de datos del paciente

APU

Menu

Presiones

Temperatura

el paciente: **OSCAR JULIAN MOLINA HIGGINS**

CIRUGIAS A LAS QUE HA SIDO SOMETIDO EL PACIENTE:

TRANSPOSICION ARTERIAL CON REIMPLANTE DE CORONARIAS [JATENE]
REPARACION DE DEFECTO DE TABIQUE INTERVENTRICULAR CON PROTESIS
COLOCACION DE CATETER PARA DIALISIS PERITONEAL

En este *Frame* se le presenta al usuario el nombre del paciente, las intervenciones quirúrgicas y un menú en el cual puede cambiar los valores por defecto del protocolo de TGA de las presiones y de la temperatura.

Figura 14. Dialogo de modificar temperatura



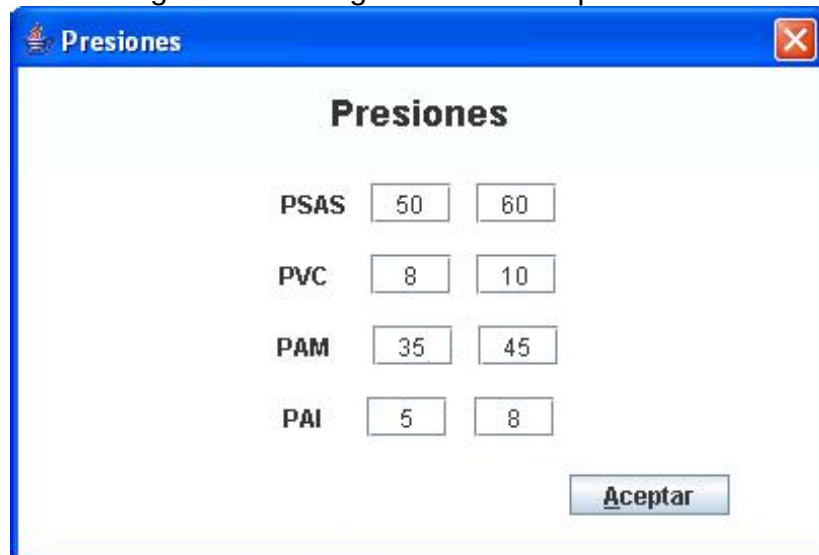
Temperatura

36 36.5

Aceptar

Este dialogo permite la modificación de los valores (por defecto) considerados normales en el protocolo de transposición de grande vasos (TGA). Este dialogo aparece cuando el usuario entre en la opción correspondiente del menú que se encuentra en el *frame* de la información del paciente.

Figura 15. Dialogo de modificar presiones



Presiones

PSAS 50 60

PVC 8 10

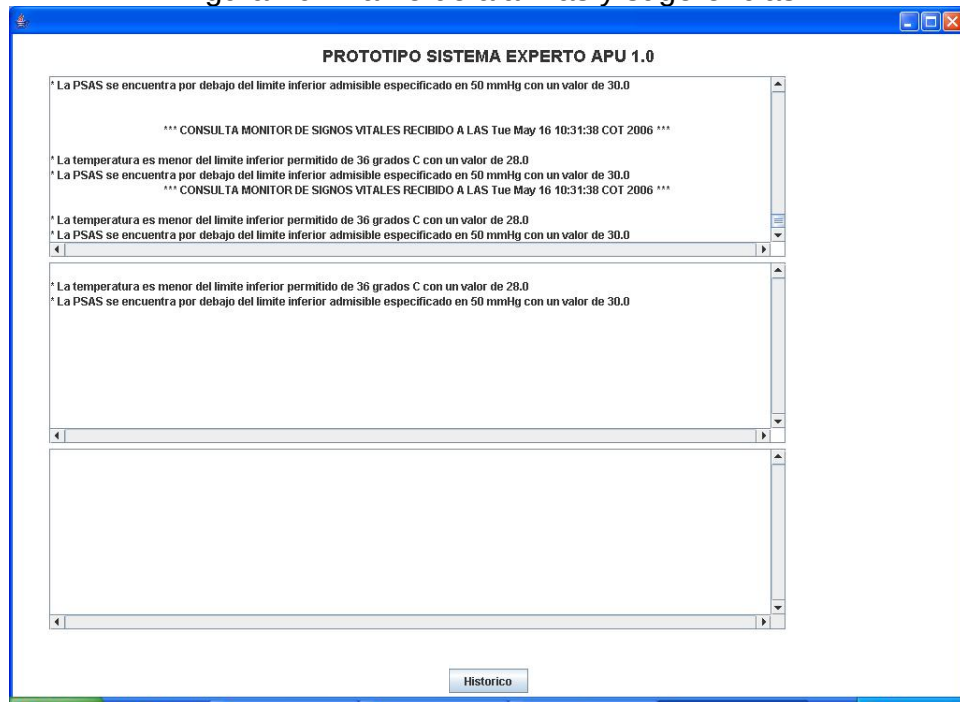
PAM 35 45

PAI 5 8

Aceptar

Este dialogo que se muestra en el momento que el usuario selecciona la opción correspondiente en el menú del *Frame* de información del paciente, permite cambiar los rangos que se consideran normales para las presiones en el protocolo de TGA. Los rangos de las presiones y las temperaturas son modificados según el medico de turno y el concepto del paciente que el tenga.

Figura 16. Frame de alarmas y sugerencias



En este *Frame* se muestra todas las sugerencias y alarmas generadas por el motor de inferencia, diferenciadas por el lugar de proveniencia de los hechos, en la primera área de impresión se presentan las que se generaron a partir de los signos obtenidos del archivo plano que genera el monitor de signos vitales y en la segunda se presentan las que el motor de inferencia genero a partir de los signos del paciente que se encuentran en la base de datos de la historia clínica electrónica. En la tercera área de impresión se muestran las sugerencias de medicamentos y posibles tratamientos de acuerdo a los signos del paciente y el tiempo que haya transcurrido en la UCIP.

Las alarmas y sugerencias se separaron para poder tener un mejor control de la aplicación, ya que así se puede apreciar con mayor facilidad donde hay fallas (como por ejemplo que no encuentre el archivo de monitor o que falten signos) y así poder localizarlas y corregirlas con mayor facilidad.

4.4.5 Primer prototipo gráfico

Para la realización del primer prototipo gráfico se hizo necesaria la integración de las diferentes clases de java hechas para el sistema experto, esto se realizó aprovechando las características de la programación orientada a objetos.

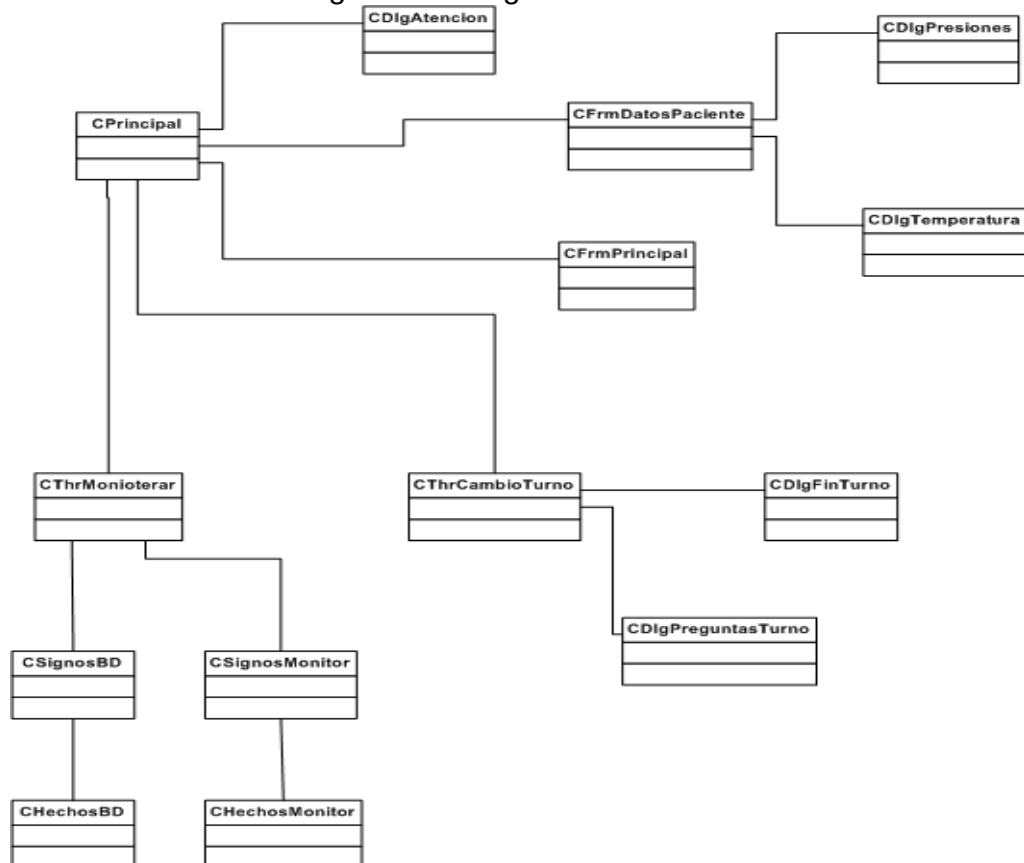
Hasta el momento se implementaron diferentes tipos de clases, unas que se encargan de adquirir de diferentes lugares (Historia clínica electrónica, monitor de signos vitales y usuario), otras que se encargan de enviar los hechos (estas clases forman parte del modulo de acceso a hechos) y de cargar las reglas en el motor de inferencia y de recibir las respuestas de este. También se

implementaron otras clases que extienden o heredan de clases ya existentes de en java como JFrame y JDialog, (estas clases son graficas y forman parte del modulo de comunicación con el usuario).

Al estar planeando la integración de los diferentes componentes del sistema experto, se encontró que no era necesaria la existencia de tres hilos (como se había planteado en el diagrama de estructuras), las funciones del hilo de medicamentos podían ser implementadas en el hilo de monitorizar los signos, por esta razón se elimino el hilo de medicamentos.

Para realizar la integración se elaboro un diagrama de clases (figura 17) en el cual se puede apreciar como debía ser la integración de las diferentes clases que se habían implementado hasta el momento.

Figura 17. Diagrama de clases



Las clases se integraron de la siguiente manera:

Dentro de la clase CPrincipal se creó una instancia de las clase CDlgAtencion, esta es instancia muestra al usuario el dialogo donde el digita la atención del paciente para que éste sea identificado. Después de ser identificado el paciente por la clase CPrincipal, ésta envía todos los datos del paciente (nombre e intervenciones quirúrgicas) a un objeto de la clase CFrmDatosPaciente el cual

es un *Frame* que le muestra al usuario la información del paciente. CPrincipal también tiene una instancia de CThrMonitorear, una instancia de CThrCambioTurno y una instancia de CFrmPrincipal, el cual es un *Frame* en el que se muestra al usuario todas las alarmas y sugerencias generadas por el motor de inferencia, diferenciadas por el lugar de adquisición de los hechos, es decir si los hechos provienen del monitor de signos vitales, las alarmas generadas por estos hechos se mostraran en un lugar diferente al las generadas por los hechos de que provienen de la historia clínica electrónica.

En la clase CFrmDatosPaciente se crearon dos objetos, uno de la clase CDlgTemperatura y CDlgPresiones, estos objetos son dos diálogos que sirven para modificar los valores normales por defecto en el protocolo de manejo de TGA, de las diferentes presiones y de la temperatura.

El hilo CThrCambioTurno tiene una instancia u objeto de la clase CDlgFinTurno, en el cual el usuario introduce la fecha de finalización de su turno el cual será el tiempo que permanecerá el hilo bloqueado. También tendrá un objeto de la clase CDlgPreguntasTurno, dialogo en el cual se harán las preguntas correspondientes al EKG.

Se creo un objeto de la clase CSignosBD y uno de la clase CSignosMonitor en la clase CThrMonitorear, estas dos clases son las encargadas de obtener los signos del paciente provenientes de la historia clínica electrónica y del monitor de signos vitales respectivamente.

CSignosBD tendrá una instancia de CHechosBD, clase encargada de enviar los datos obtenidos de la base de datos en forma de hechos al motor de inferencia.

Finalmente un objeto de la clase CHechosMonitor monitor será instanciado en la clase CSignosMonitor.

4.4.6 Pruebas del sistema

Para poder realizar las pruebas con mayor facilidad y tener un histórico de las reglas que se ejecutan cada vez que se corre la aplicación, al sistema se le agrego la capacidad de realizar un informe cada media hora que contiene las repuestas de las reglas ejecutadas y los medicamentos sugeridos como tratamiento del paciente.

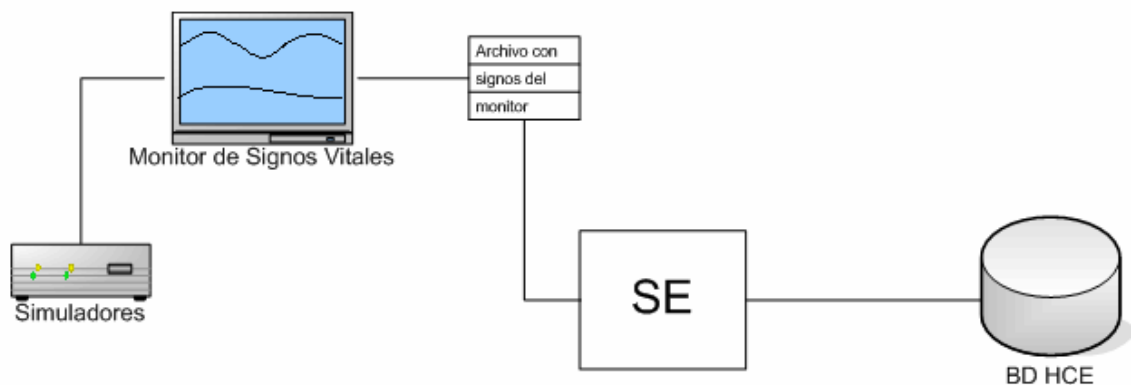
Para probar el funcionamiento del prototipo de sistema experto, se hicieron dos clases diferentes de pruebas; se hizo uno prueba con unos simuladores y con los datos de un paciente que se encuentran almacenados en una base de datos para pruebas; también se hizo una prueba con un paciente en la UCIP.

La prueba realizada con los simuladores se hizo de la siguiente manera:

El monitor de signos vitales se conectó a los simuladores. Estos simuladores generan señales que son muy similares a las señales generadas por el cuerpo humano y que son tomadas del paciente que se encuentra en cuidados intensivos. Estos simuladores generan las señales de los diferentes signos que puede medir el monitor, como la frecuencia cardíaca, la presión venosa central, la presión auricular izquierda, etc.

Con el monitor conectado a los simuladores, se genera un archivo con valores con valores mas realistas y muy similares a los que tendría un paciente. Para esta prueba, este archivo se genera para ser leído por el sistema experto en un tiempo menor al que lo haría normalmente.

Figura 18. Prueba del sistema experto con simuladores



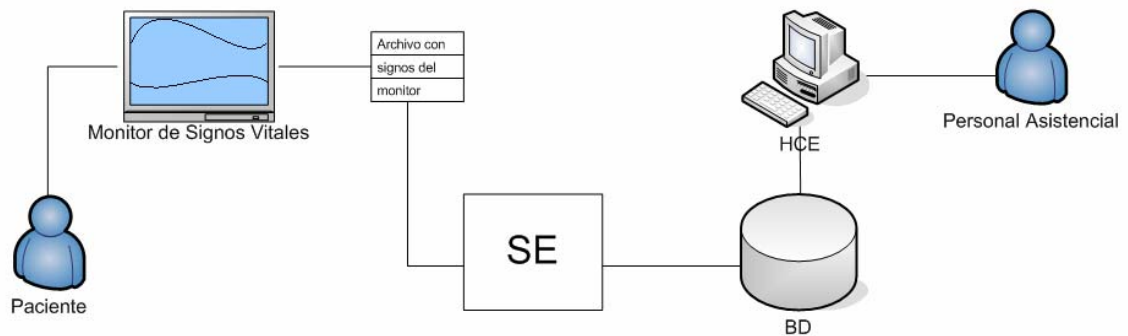
También el sistema accede también a una base de datos para pruebas, la cual contiene datos de un paciente que alguna vez fue operado de TGA y estuvo hospitalizado en la UCIP de la institución.

Con todo esto listo (ver figura 18) se puso en marcha el sistema y se pudo corroborar que con los valores generados por los simuladores y los que se encuentran en la base de datos (estos valores son pasados al motor de inferencia en forma de hechos), se activaran (o no se activaran) las reglas correspondientes.

Las pruebas con el paciente de la UCIP se realizaron así:

Para estas pruebas se puso a funcionar el sistema experto en el computador donde se instalo el monitor de signos vitales. El sistema leía los datos generados por el monitor de signos vitales que estaba conectado al paciente y accedía a los datos que eran consignados en la Historia Clínica Electrónica (HCE) por el personal asistencial (ver figura 19).

Figura 19. Prueba del sistema con un paciente



El sistema era revisado periódicamente para encontrar los posibles errores que se produjeran y que no hubieran podido ser detectados en la prueba con los simuladores.

Esta prueba fue de gran utilidad ya que al funcionar por tanto tiempo (el sistema funcionó el tiempo que estuvo el paciente hospitalizado en la UCIP, aproximadamente 3 días), se encontraron ciertos errores que no se podían apreciar o que no se generaban al probar el sistema en periodos cortos de tiempo. El error más notable y que era de gran importancia corregirlo era uno que se producía cuando después de varias horas de trabajo se llenaba la memoria que se asigna para la máquina virtual de Java.

Después de suspendido el funcionamiento del sistema, se procedió a revisar todos los informes generados por el sistema experto para ver que reglas se ejecutaron, corroborando con los datos de la HCE y con los valores de los signos que se encontraban en los archivos generados por el monitor de signos vitales si estas reglas eran las que tenían que haberse ejecutado o cuáles reglas no se ejecutaron cuando había valores con los cuales tenían que haberse ejecutado.

4.5 FASE DE CONSTRUCCIÓN

En esta fase se realiza una implementación iterativa del resto de requisitos de menor riesgo y elementos más fáciles.

4.4.1 Modulo para la administración de reglas

Para poder seguir incrementando la base de conocimientos se creó una interfaz con el usuario que permite la elaboración de nuevas reglas, que después de realizarse una primera versión, fue revisada y modificada para cumplir con nuevos requerimientos.

Para la creación de nuevas reglas se incluyó una nueva estrategia para crear reglas que logren obtener conclusiones que dependan de conclusiones tomadas con anterioridad por otras reglas, es decir que dependen de que una regla (o varias) se haya ejecutado; de esta manera es posible llegar a obtener una conclusión que dependa de conocimiento previo. Esta estrategia se conoce como encadenamiento de reglas.

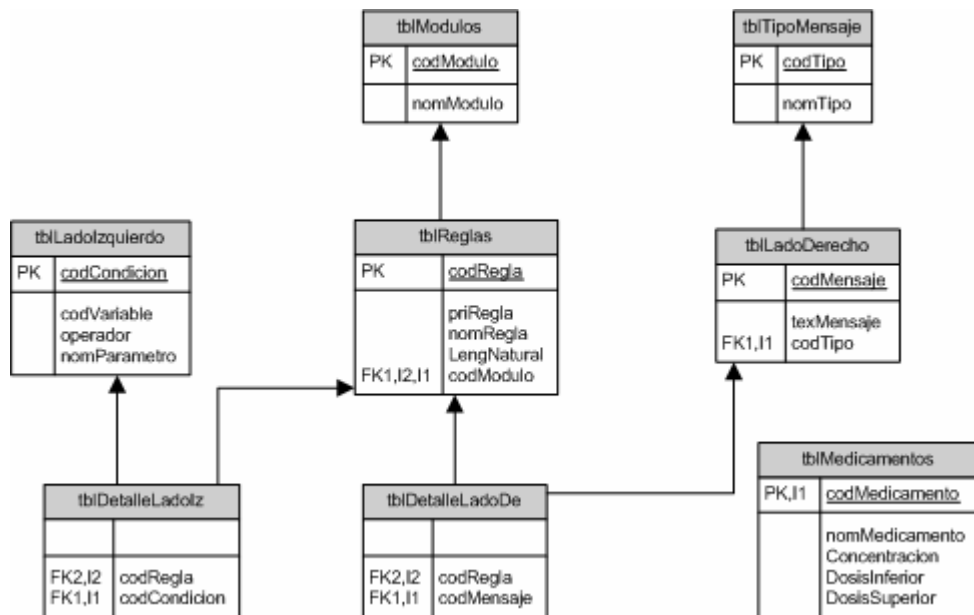
Los requerimientos finales de este modulo son los siguientes:

- Realizar los procedimientos necesarios para la creación de reglas de producción que alimenten la base de conocimientos del sistema experto.
- Obtener todas las diferentes variables que se encuentran en la base de datos del sistema experto, diferenciando el tipo de variable.
- Obtener el nombre de cada uno de los protocolos (módulos) existentes para ser utilizados en la creación de las diferentes reglas.
- Seleccionar uno de los diferentes módulos existentes para saber a cual de los módulos pertenecerá la regla que se va a crear.
- Las reglas creadas deben poder efectuar alguno de tres tipos de acciones, enviar al usuario mensajes (diferentes alarmas y sugerencias que puedan darse al activar una regla), sugerir medicamentos o cambiar de protocolo (modulo) cuando esto sea necesario.
- El lado izquierdo de la regla debe poder armarse con la combinación de una o varias de las variables que provienen de la base de datos del sistema experto.
- Las diferentes reglas ya existentes en la base de datos se podrán eliminar o editar según crea conveniente el usuario.
- La regla debe ir siendo armada con elementos en un lenguaje natural, así el usuario podrá ver como queda la regla que el a creado en un lenguaje claro y entendible (En español y no en la sintaxis del motor de inferencia).
- Mostrar al usuario los diferentes valores que puede tomar la variable cuando esta es cualitativa.
- Cada protocolo de manejo de la Unidad de Cuidado Intensivo se convierte en una base de conocimiento (modulo).
- El modulo debe mostrar las diferentes reglas pertenecientes a protocolo donde se este creando la regla para permitir realizar el encadenamiento de reglas.

Para este modulo también se realizaron los diferentes casos de uso junto con los diagramas de secuencias correspondientes. [Anexo C]

El siguiente paso para la elaboración de este modulo fue el diseño e implementación de la base de datos, que después de realizarle los ajustes necesarios esta fue la base de datos resultante:

Figura 20. Modelo de base de datos.



La notación es la siguiente: FK es llave foránea (Foreign Key), PK es llave primaria (Primary Key) y las flechas representan relaciones uno a muchos (muchos la cola de la flecha y uno la punta).

Con todo lo anterior listo se procedió a la implementación del modulo.

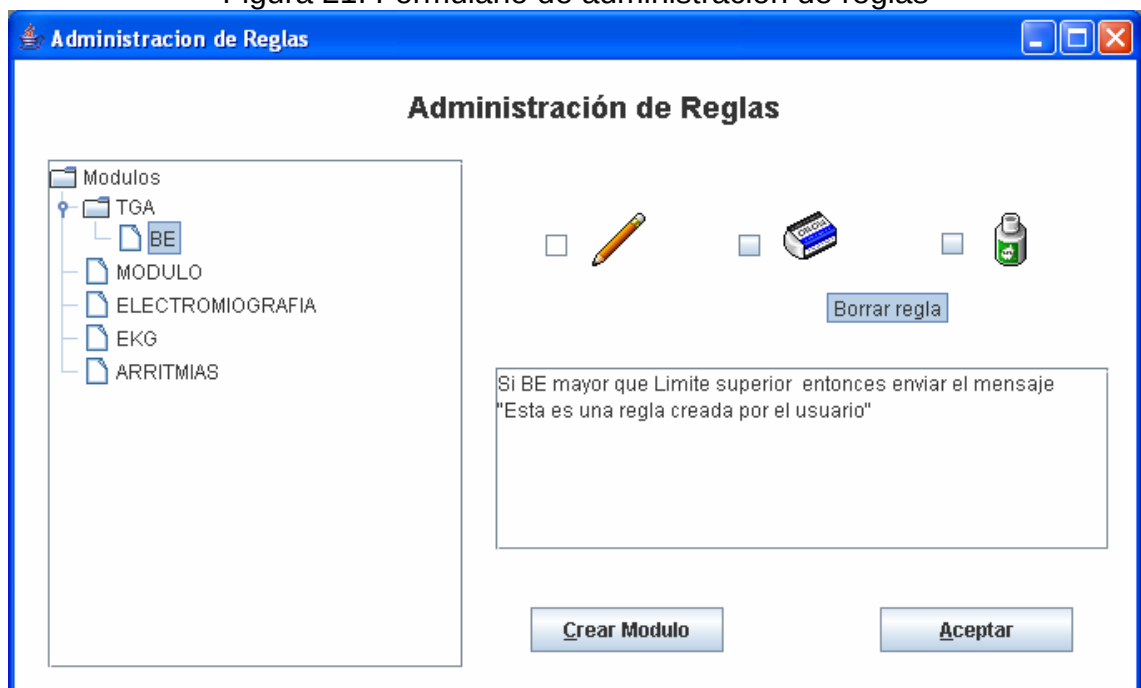
La interfaz del modulo de creación de reglas, presenta un formulario principal donde se realiza la administración de las reglas y un dialogo donde el experto crea o edita las reglas.

En el momento que el usuario entra en la sección de administración de reglas, el sistema presentara el formulario (figura 21), aquí el usuario puede observar en el árbol (JTree de java) que se encuentra en la izquierda el nombre de cada uno de de los módulos existentes (representados por carpetas cuando han creado reglas para ese modulo ó representados por hojas cuando no se ha creado ninguna regla), de esta manera el usuario selecciona el modulo al cual pertenecerá la regla haciendo clic sobre el nombre de éste. Si el usuario hace doble clic y el modulo tiene reglas creadas, los nombres de éstas se muestran y el usuario puede seleccionar la regla haciendo clic sobre el nombre de ésta, mostrándose así la regla en lenguaje natural en el área de texto que se encuentra en la parte media del lado derecho.

En la parte superior derecha se encuentran tres dibujos con cajas de chequeo a su lado (JCheckBox de java), estos dibujos representan las tres opciones que tiene el usuario (crear, editar y eliminar), en el momento que el usuario selecciona un modulo, la caja de chequeo que se encuentra al lado de la imagen del lápiz (crear regla) se habilita y cuando el usuario selecciona una regla, se habilitan las otras dos.

Si el usuario selecciona la opción de crear o la de editar regla, automáticamente el sistema muestra al usuario el dialogo que se aprecia en la figura 22. Cuando el usuario selecciona eliminar la regla se muestra un dialogo pidiendo una confirmación o una cancelación.

Figura 21. Formulario de administración de reglas



En el dialogo que se presenta a continuación el usuario puede seleccionar entre las diferentes reglas que existen en el modulo (cuando desea hacer encadenamiento de reglas) y las diferentes variables que se han establecido para el sistema, para mayor comodidad se crearon dos cajas de texto en la parte superior de cada una de esta lista para que el usuario pueda ubicar con mayor facilidad la regla o la variable deseada. Cuando un usuario selecciona una regla, los combobox desaparecerán; cuando el usuario seleccione una variable, el contenido de los combos cambiara de acuerdo al tipo de variable (rango numérico, cualitativo, estándar por tablas). El usuario también debe llenar las cajas de texto de nombre y prioridad, si el usuario no establece una prioridad esta se toma automáticamente como cero. Después el usuario selecciona el tipo de lado derecho y según esto escoge o llena ciertos datos, si es un mensaje, llena la caja de texto correspondiente, si es cambio de protocolo, escoge entre alguno de los protocolos existentes y si es un medicamento escoge entre alguno de los medicamentos disponibles.

Cuando el usuario haya terminado de llenar todo oprime el botón que dice mostrar, el sistema muestra al usuario la regla en lenguaje natural y este la acepta o la modifica hasta obtener la regla deseada.

Figura 22. Dialogo de creación y edición de reglas

REGLAS

Buscar Regla:

Codigo	Prioridad	Nombre Regla	Seleccionada
1	5896	BE	☒

Buscar Variable:

Codigo Variable	Nombre Variable	Seleccionada
12	PVCmmHg	☐
9	PPSmmHg	☐
15	Oxigenoterapia	☐
109	Sangrado	☐
19	O2Lmin	☐
6	MAP. (mmHg)	☐
3	P.A.S. (mmHg)	☒
5	P. Arterial Media	☐
23	FR	☐
123	BE	☐
41	Peso (Kg)	☐
61	Respuesta Verbal	☐
25	PEEP	☐
22	Halogenado	☐

Nombre de la regla: Prioridad:

BE

P.A.S. (mmHg)

Seleccione un elemento

Seleccione un elemento

Seleccione un elemento

Respuesta de la Regla:

☒ Mensaje ☐ Dosis ☐ Protocolo

Si ocurrió BE y P.A.S. (MMHG) menor que Limite superior entonces enviar el mensaje "Esta es una regla que depende de otra regla"

Un ejemplo de las reglas que se generan por medio de esta sección son los siguientes:

Reglas de mensajes:

(defrule TGA::RITMO_Y_PESO (declare (salience 2000))

(Regla REGLA_DEL_RITMO)

(PESOKGA ?pesokga) (PESOKG ?pesokg &: (> ?pesokg ?pesokga))

=>

(assert (Mensaje "La regla RITMO_Y_PESO se empleo para establecer que: El paciente presenta sobre peso con una aceleración del ritmo cardiaco *1"))

(assert (Mensaje1 (format nil "Esta decisión se tomo debido a que se activo con anterioridad la regla REGLA_DEL_RITMO y PESOKG > PESOKGA con los siguientes valores PESOKG = %.2f y PESOKGA = %.2f *1" ?pesokg ?pesokga)))

(assert (Regla RITMO_Y_PESO)))

En el lado izquierdo de la regla de envío de mensajes se puede apreciar que primero tiene una definición, la definición con tiene el modulo al cual pertenece la regla, junto con el nombre de esta separados por el símbolo "::". La línea (Regla REGLA_DEL_RITMO) es un hecho que debe existir, es decir debió haberse ejecutado con anterioridad la regla llamada REGLA_DEL_RITMO para que esta regla se active y sea ejecutada por el motor de inferencia. El lado derecho presenta tres aserciones (assert) con los siguientes hechos: el hecho Mensaje es un mensaje que se le envía a Java por medio del puente jclips, este mensaje contiene parte de la explicación al usuario; el segundo hecho, Mensaje1 contiene la razón por la cual se disparo la regla junto con los valores del peso y de el limite del peso que se encuentra en la base de datos. El símbolo *1 es una guía para indicar en que sección del formulario de alarmas, sugerencias y tratamiento debe imprimirse el mensaje. La última aserción es un hecho que contiene el nombre de la regla para indicar al motor de inferencia que la regla fue ejecutada.

Regla de medicamentos:

```
(defrule TGA::ADRENALINA (declare (salience 2000))
(Regla REGLA_DEL_RITMO)
(PASMMHGB ?pasmmhgb) (PASMMHG ?pasmmhg &: (< ?pasmmhg
?pasmmhgb))
=>
(assert (Mensaje "CAPTOPRIL ** *2" ))
(assert (Mensaje1 (format nil "El paciente presenta síntomas que sugieren el
uso del medicamento CAPTOPRIL. Esta decisión se tomo debido a que se
activo con anterioridad la regla REGLA_DEL_RITMO y PASMMHG <
PASMMHGB con los siguientes valores PASMMHG = %.2f y PASMMHGB =
%.2f *2" ?pasmmhg ?pasmmhgb)))
(assert (Regla ADRENALINA))
)
```

Esta regla en su lado derecho presenta el assert Mensaje con el nombre del medicamento que debe aparecer para que el medico indique su dosis en la sección correspondiente para el caso en el formulario de alarmas, sugerencias y tratamiento (ver numeral 4.5.2), el assert Mensaje 1 presenta el porque se disparo la regla.

Regla de cambio de protocolo:

```
(defrule TGA::CAMBIO_A_EKG (declare (salience 2000))
(PRESIÓNAURICULAIZQUIERDAA ?presiónauriculaizquierdaa)
(PRESIÓNAURICULAIZQUIERDA ?presiónauriculaizquierda &: (>=
?presiónauriculaizquierda ?presiónauriculaizquierdaa))
=>
(assert (Mensaje "La regla CAMBIO_A_EKG se empleo para establecer que se
procederá a cambiar al modulo: EKG *1" ))
(assert (Mensaje1 (format nil "Esta decisión se tomo debido a que
PRESIÓNAURICULAIZQUIERDA >= PRESIÓNAURICULAIZQUIERDAA con
los siguientes valores PRESIÓNAURICULAIZQUIERDA = %.2f y
```

```

PRESIÓNAURICULAIZQUIERDAA = %.2f *1" ?presiónauriculaizquierda
?presiónauriculaizquierdaa)))
(assert (Regla CAMBIO_A_EKG))
(focus EKG)
)

```

EL lado derecho de esta regla presenta además de los asserts con la explicación y con el nombre de la regla, contiene la instrucción focus, la cual hace que el motor de inferencia cambie al modulo indicado, en este caso cuando se ejecuta la regla el motor de inferencia pasara al modulo EKG y evaluara las reglas que se encuentren en este.

4.5.2 Cambios en el sistema

En esta fase se decidió hacer ciertos cambios en el sistema, el primero de estos fue que el dialogo de pedir la fecha y la hora de cambio de turno se elimino, ahora las preguntas del electrocardiograma se realizan cada 12 horas y no cada vez que el medico de inicie turno.

El segundo se efectuó en el formulario de alarmas sugerencias y medicamentos, este formulario (Figura 23) se modifiko para que el medico pudiera indicar la dosis de los diferentes medicamentos sugeridos por las reglas de medicamentos que se introducen en la sección de creación de reglas. Cada vez que se sugiere un medicamento este aparecerá en la lista que se encuentra en la parte inferior derecha, el medico hará clic en la caja de chequeo (check box) y aparecerá un dialogo (Figura 24) donde aparecerán los datos necesarios del paciente (nombre, edad, etc) y los datos del medicamento (nombre y concentración), así el usuario podrá indicar al sistema la dosis que desea suministrar del medicamento, esta dosis sirve para calcular el suministro o "velocidad" con que se le va a suministrar el medicamento en la bomba de infusión, esto se calcula de la siguiente manera: $(\text{dosis} * \text{peso del paciente}) / \text{concentración del medicamento}$. El resultado de esta operación es mostrado en el área de texto de la parte inferior izquierda.

Figura 23. Formulario de alarmas, sugerencias y tratamiento

Resultado Monitoreo Signos Vitales:

*** CONSULTA MONITOR DE SIGNOS VITALES RECIBIDO A LAS Sat Jul 15 14:30:50 COT 2006 ***

- * La PVC se encuentra por fuera del rango admisible especificado entre 8.00 y 10.00 mmHg
- * El valor de la presion venosa central es de 10.50 mmhg

Resultado Monitoreo Variables Historia Clinica:

- * El valor de la presion Arterial de dióxido de carbono sobrepasa la meta con un valor de: 47.40
- * El PH esta por encima del valor requerido, con un valor de: 7.46
- No se encuentra registrado en la base de datos el valor de Ritmo

Resultado Tratamientos Y Controles:

- * Se requiere control de Sodio
- * Se requiere control de Potasio
- * Se requiere control de Calcio Ionico
- * Infusion 0.3 de potasio, control cada media hora

Medicamento	Seleccionar
NOREPIREFRINA	☐

Figura 24. Dialogo de dosis del medicamento

Datos de medicamento:

Medicamento: NOREPIREFRINA

Concentracion: 1234

Dosis:

Datos del paciente:

Nombre: OSCAR JULIAN MOLINA HIGGINS

Peso: 12.0

Nacimiento: 2 años 6 meses 6 dias

Aceptar **Cancelar**

4.5.3 Administración de perfiles

El sistema experto tiene diferentes secciones pero no todo sus usuarios tienen la necesidad, ni están en la capacidad de entrar a todas estas, así por ejemplo el experto que tiene el conocimiento necesario para incrementar la base de

conocimiento, esta en la capacidad de ingresar, modificar o eliminar reglas y definir las variables necesarias para el funcionamiento del sistema, así que él tiene que acceder a las secciones del sistema que permitan la realización de estas actividades; pero en cambio el personal asistencial (encargado de ubicar el paciente en la UCI Pediátrica) solo puede ejecutar el sistema para el monitoreo del paciente. Por esto es que se hace necesario definir los diferentes perfiles de usuario, para que personal acceda solo a las secciones que se crea conveniente y de esta manera proteger secciones que contengan contenido de gran importancia (como por ejemplo las reglas de la base de conocimiento).

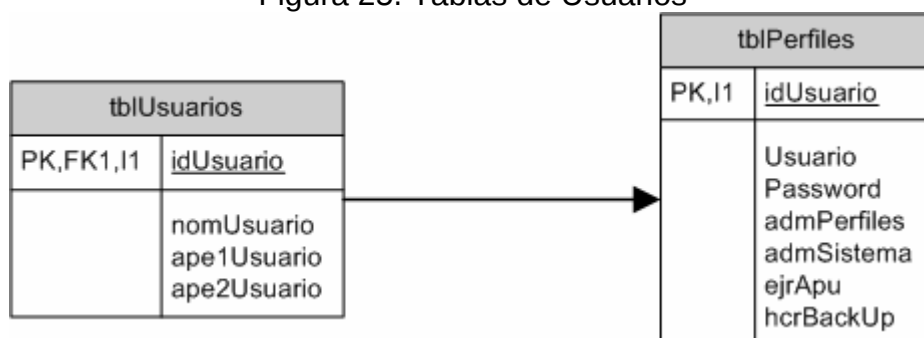
En el sistema existe un administrador general, el cual es creado directamente en la base de datos, este administrador es el único que posee los permisos necesarios para entrar a la sección del sistema para la creación y administración de perfiles de usuario. Los usuarios que el administrador cree solo se le asignaran permisos para entrar a las restantes secciones del sistema, de acuerdo a donde se crea conveniente que tiene que entrar el usuario.

Para crear nuevo usuarios primero se diseñó, implementó y ajusto una base de datos para almacenar el administrador general y todos los usuarios que este cree.

La base de datos consta de dos entidades, estas entidades son: Perfiles y Usuarios. La entidad Usuarios tiene como atributos todos los datos necesarios del usuario (Nombre completo del usuario y sus dos apellidos) y una llave principal, el cual es un código que identifica al usuario. En la entidad Perfiles encontramos el usuario, su clave de entrada (password) y los diferentes permisos que este posee. Las dos entidades poseen una relación uno a uno.

Teniendo ya el diseño de la base de datos, se paso a implementarla, el diagrama de modelo de bases de datos resultante es:

Figura 25. Tablas de Usuarios



En la anterior figura se pueden apreciar las dos tablas resultantes con sus diferentes atributos, llaves principales (PK), llaves foráneas (FK) y su relación.

El siguiente paso fue el diseño e implementación de un formulario (figura 26) para la creación de los diferentes perfiles de usuario, en él, el administrador

general introduce todos los datos del usuario, junto con el usuario (nombre con el que se va identificar en el sistema) y la clave de entrada (password).

Figura 26. Perfiles de Usuario

Perfiles de Usuarios

Identificacion: Usuario:

Nombre: Contraseña:

Primer Apellido: Confirmar:

Segundo Apellido: Permisos: ☐ Ejecutar APU

☐ Administrar Sistema

☐ Hacer Back Up

☒ Nuevo ☐ Modificar ☐ Eliminar

4.5.4 Copia de seguridad

Como el sistema experto tiene una base de conocimiento que es de vital importancia para el, aparece la necesidad de hacer una copia de seguridad, como esta base de conocimiento esta compuesta por reglas que se encuentran en la base de datos, la copia de seguridad debe hacerse sobre los archivos de ésta. Otros archivos que son necesarios copiar, son los informes que se generan cada vez que el sistema es puesto en marcha, estos informes son de gran utilidad para la validación de reglas y por lo general esta validación se hace en otro equipo por lo cual se hace necesario hacer una copia para el transporte de estos.

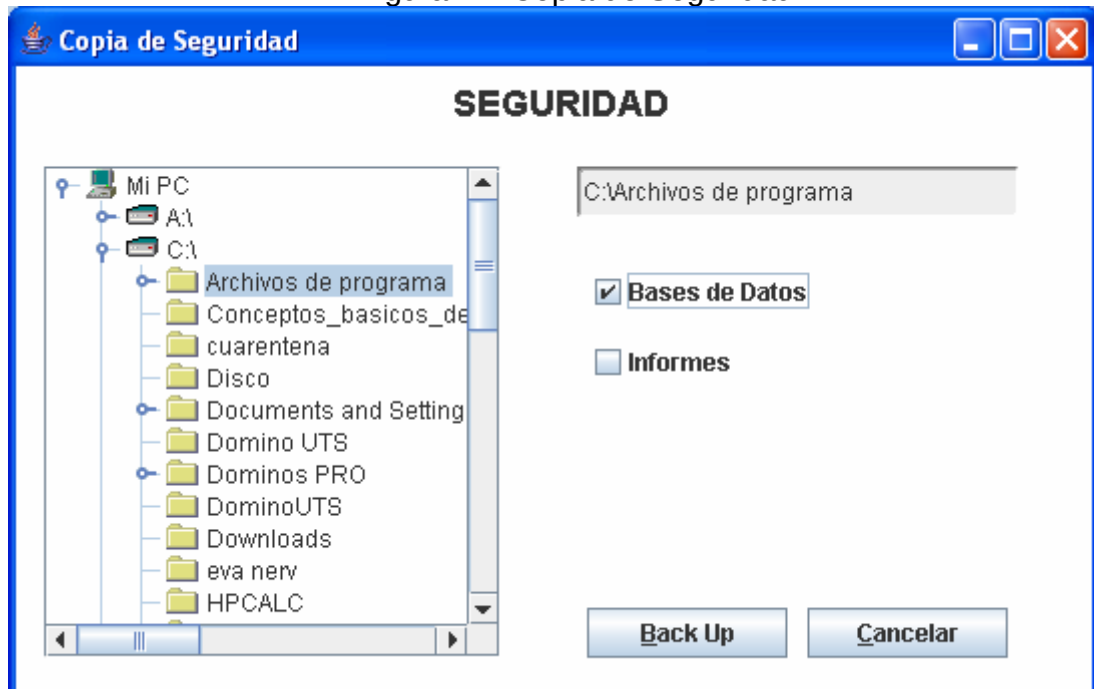
La mejor forma de hacer la copia es en un formato que comprima los archivos para que puedan ser transportados con mayor facilidad y que mejor para esto que un archivo ZIP.

Teniendo claro como debe ser el funcionamiento (para esto se realizo un diagrama de secuencias [Anexo D]) de esta sección se procede a diseñar e implementar una interfaz grafica que fuera muy amigable con el usuario y que le permitiera hacer con mucha facilidad las copias de seguridad.

El formulario que se implementó (ver figura 27), tiene un árbol (JTree de java) que muestra al usuario las diferentes unidades que tiene el equipo donde se esta corriendo el sistema (java a diferencia de otros lenguajes no posee un

control especial para esto, en java se implementa por medio de un árbol), al seleccionar cada una de estas unidades y hacer doble click sobre esta, se muestran los diferentes directorios que hay en la unidad.

Figura 27. Copia de Seguridad



En el formulario el usuario selecciona la unidad y/o directorio donde desea realizar la copia en el árbol que esta a la izquierda, la ruta completa donde se realizará la copia aparece en la parte superior del formulario. Luego el usuario selecciona opción de lo que desea copiar y acepta, para que el sistema genere los archivos ZIP que el usuario necesita y le indique si fue posible su creación.

Los archivos creados tendrán el nombre de “base datos” ó “informes” (según lo que haya seleccionado para hacer la copia) junto con la fecha y la hora de creación de este, esto con la finalidad de tener un mejor control sobre las copias que se van realizando.

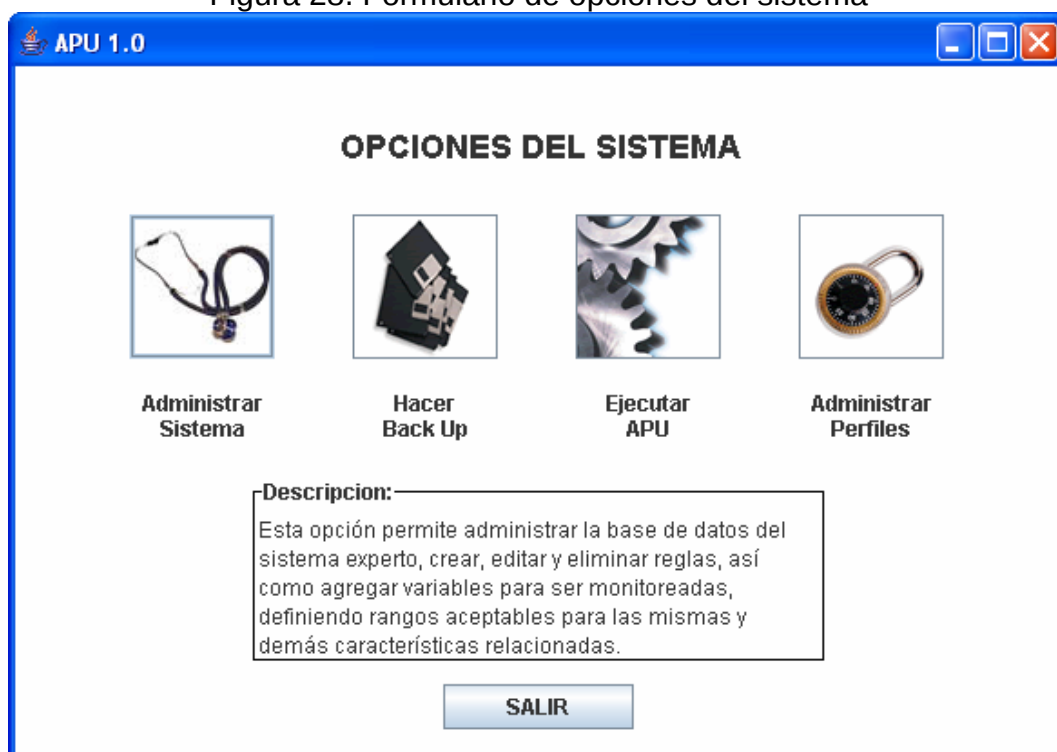
4.5.5 Formularios y diálogos de acceso y control

Para que el usuario pueda acceder a las diferentes secciones del sistema, se diseño e implementaron dos formularios y un dialogo. Los dos formularios muestran al usuario las diferentes secciones a las que se puede acceder y el dialogo valida el usuario y el password para saber si el usuario tiene permiso de entrar a esta sección.

Formulario de opciones del sistema: este formulario (figura 28) muestra al usuario las cuatro secciones principales del sistema, estas secciones son: Administrar Sistema (en esta parte del sistema se encuentran las secciones de agregar, editar y eliminar reglas; junto con la sección de incluir variables en el

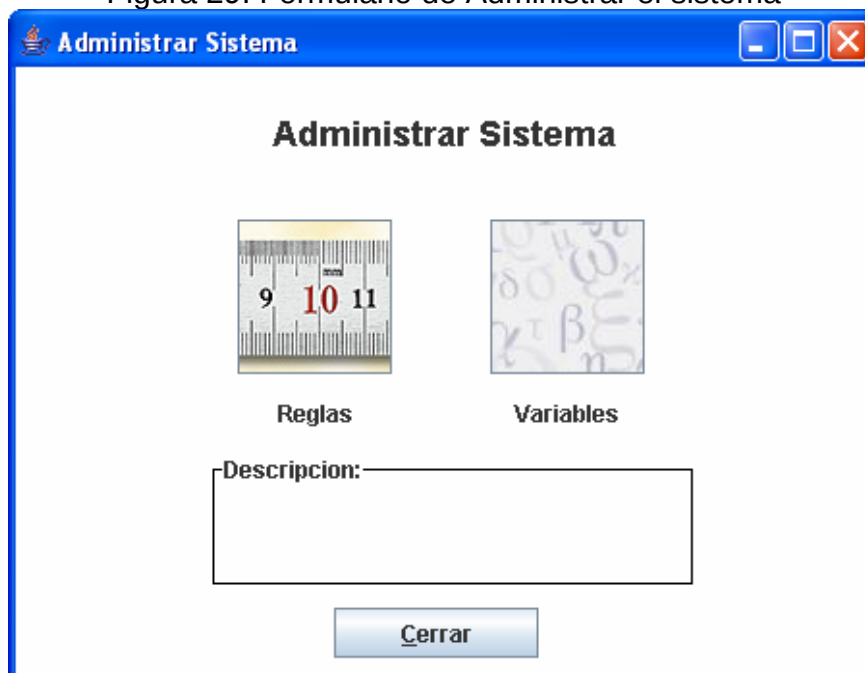
sistema.), hacer Back Up, Ejecutar APU (aquí se ejecuta el sistema experto para que monitoree al paciente) y Administrar Perfiles.

Figura 28. Formulario de opciones del sistema



Formulario de Administrar Sistema: en este formulario que se presenta en la figura a continuación, el usuario puede acceder a las secciones de administrar las reglas del sistema experto y de incluir nuevas variables a monitorizar.

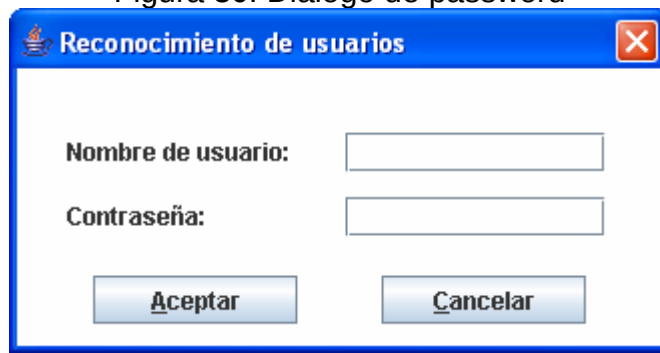
Figura 29. Formulario de Administrar el sistema



Los dos formularios poseen una imagen representativa de la opción a la que se puede ingresar por medio de ella, además tienen una descripción que aparece dentro del cuadro que tiene el título de descripción. La descripción es un breve comentario de lo que se puede hacer en la sección a donde se desea ingresar, la descripción aparece cuando se pasa el ratón sobre alguna de las imágenes.

Dialogo de password: Este es un dialogo (figura 30) sencillo donde la persona que desee ingresar a alguna de las secciones que se encuentran en el formulario de opciones del sistema, puede ingresar el usuario y el password para que el dialogo valide en la base de datos si el nombre de usuario y la contraseña son correctos; y pueda acceder a esta sección si tiene el permiso de entrar a ella.

Figura 30. Dialogo de password

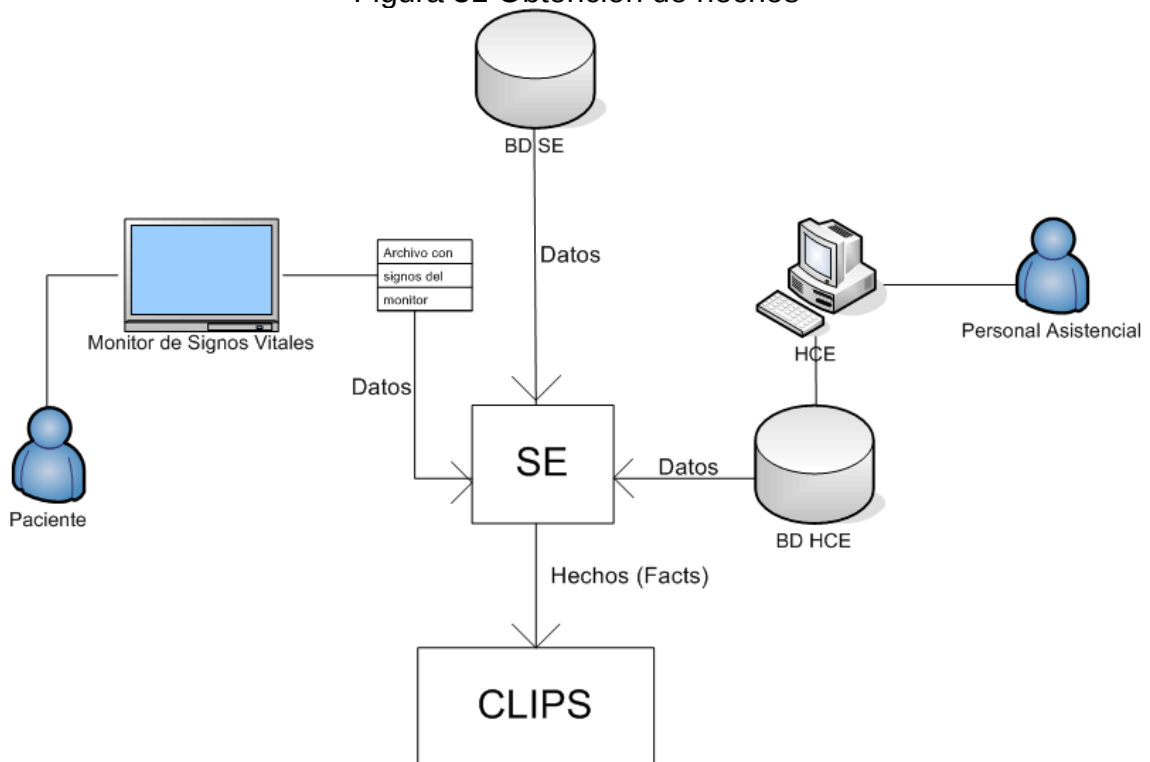


The image shows a Windows-style dialog box with a blue title bar. The title bar contains a small icon on the left and a close button (an 'X' in a red square) on the right. The title text is 'Reconocimiento de usuarios'. The main area of the dialog is white and contains two labels with corresponding text input fields: 'Nombre de usuario:' followed by a text box, and 'Contraseña:' followed by a text box. At the bottom of the dialog, there are two buttons: 'Aceptar' (Accept) on the left and 'Cancelar' (Cancel) on the right. Both buttons have a light blue gradient and a thin border.

4.5.6 Nuevos hechos

Con la creación de nuevas reglas y la inclusión de nuevas variables para monitorear, se hace necesaria la adquisición de los hechos que activan estas reglas y que representan estas nuevas variables. Para esto se realizó una nueva clase de java, la cual se encarga de conseguir los diferentes hechos que activan las reglas de la base de conocimiento. Esta clase (llamada CHechosReglasBD), se encarga de conseguir los valores de las variables que se han incluido en el sistema experto (en la sección creada para esto) y que son las que se utilizan para la creación de nuevas reglas; los valores de éstas se traen de los diferentes lugares donde se encuentran (ver figura 31), si la variable proviene del monitor de signos vitales, esta se lee del archivo, si viene de la base de datos, se trae de esta.

Figura 31 Obtención de hechos



En la sección de ingresar variables las variables se pueden definir por rango, cualitativa y por estándar de tablas, debido a esto la clase que se encarga de conseguir los valores debe hacer diferentes procedimientos. Si la variable fue definida por rango, ésta aparece en un diálogo al cual se accede en el menú que se encuentra en el formulario que contiene la información del paciente (estos diálogos fueron modificados por otro practicante que hace parte del proyecto) para que sea modificado por el usuario. Los valores del rango (límite superior y límite inferior) que se encuentra en este diálogo se toman y se pasan al motor de inferencia como hechos. El valor que presente la variable es buscado en la base de datos o en el monitor de signos vitales.

Si la variable está definida como cualitativa, se pasa como hecho el valor cualitativo que ésta presente en el momento; pero si la variable está definida por tablas, primero se busca el valor que presentan las variables que son los criterios, con estos valores se busca en la base de datos del SE el rango o el valor cualitativo que debe tener la variable según el valor de los criterios y se pasa este (o estos) valores al motor de inferencia en forma de hechos, por último se busca el valor que tiene la variable y también se pasa en forma de hechos.

En este proceso de adquisición de hechos se presentaron algunos inconvenientes, debido a que la base de datos de la historia clínica electrónica está hecha de una forma muy particular, esta no es relacional, por consiguiente no existen llaves foráneas y la integridad referencial es mínima. En algunas ocasiones algunas consultas se modificaron debido a que existen atributos que a pesar de ser el mismo, tienen tipos de datos diferentes en las diferentes

tablas, lo que causaba errores al momento de conseguir los hechos necesarios para el funcionamiento del sistema experto.

En la Tabla 1, se puede apreciar mejor lo ultimo explicado, aquí se presentan ciertas variables con rangos y que tienen un criterio para escoger el rango normal (la variable edad), por ejemplo la variable peso, se busca el calor que tenga el criterio Edad, luego se busca en la tabla la posición que corresponde a la edad que tiene el paciente y el rango o el valor que se encuentra ahí (si es único) se pasa al motor de inferencia en forma de hechos, después se busca el valor del peso y se pasa en forma de hecho al motor de inferencia. Con estos hechos el motor de inferencia ya podrá activar y luego ejecutar todas aquellas reglas que dependan de estos hechos.

Tabla 1. Relación edad, peso y frecuencias respiratorias, cardiaca y presión arterial

Edad	Peso/Kg	F.C.: lpm	F. R.: rpm.	Tensión Arterial: mm. Hg. Sistólica Diastólica	
Pretérmino	1	140-160	40-60	39-59	16-36
RN	3-4	120-180	30-50	50-75	30-50
6 m	7	100-130	20-40	80-10	45-65
1 año	10		20-30	80-105	45-70
1-2 a	10-12		15-25	80-120	50-80
2-3 a	12-14	80-110	15-20	85-130	55-90
3-6 a	12-19				
6-8 a	19-26	70-100	13-15	90-140	60-95
8-10 a	26-32				
10-14 a	32-50				
> 14 a	> 50				

4.6 FASE DE TRANSICIÓN

En esta fase se realizan la integración de todos los componentes y se realizan las correspondientes pruebas.

4.6.1 Cambios

Como se elaboró un modulo que permite el análisis del electrocardiograma, ya no se hace necesario el dialogo de preguntas del EKG, es por esto que en la integración final este dialogo es eliminado del sistema.

4.6.2 Estándar de desarrollo

En esta etapa se complemento el estándar de programación, completando ciertos tópicos que habían quedado incompletos en el momento de su elaboración. [Anexo F]

4.6.3 Integración y pruebas

Después de haber implementado cada una de las secciones (módulos) del sistema experto, se procedió a realizar cada una de las correspondientes pruebas.

A medida que se iba programando cada sección se iban realizando las llamadas pruebas de escritorio, en las cuales cada vez que se iba agregando código, se hacían pequeñas ejecuciones en busca de errores y al detectar un error, se recorría el código en busca de la línea o la sección de código que presentaba la falla y se corregía.

Para realizar las pruebas integración se baso en la metodología de la Integración no incremental, en esta primero se prueba cada modulo por separado, luego se integran todos de una vez y se prueba el programa completo.

Lo primero que se hizo, fue realizar las pruebas de cada modulo por separado mirando ciertos casos de prueba, de estas pruebas se obtuvieron las siguientes tablas:

Prueba al formulario de administración de reglas:

Tabla 2. Pruebas del formulario de administración de reglas

Nombre del caso	Entrada	Salida Esperada	Salida Real
Seleccionar modulo y creación de reglas.	Nombre del modulo	Dialogo de creación de reglas listo para su uso.	OK
Seleccionar regla y seleccionar editarla.	Nombre de regla a editar.	Dialogo de editar reglas con todos los campos con la información de la regla, llenos y listos para su edición.	OK.
Seleccionar regla y seleccionar eliminarla.	Nombre de la regla a eliminar.	Confirmación de eliminación y eliminación de registros en la base de datos.	Eliminación de registros OK.
Generación del archivo CLP.	Datos las reglas que se encuentran en la base de datos.	Archivo con extensión .clp con las reglas en la sintaxis del motor de inferencia.	Generación del archivo OK.

Prueba al formulario de Creación de reglas:

Tabla 3. Pruebas del formulario de creación de reglas

Nombre del caso	Entrada	Salida Esperada	Salida Real
Creación de regla con lado derecho de mensaje con solo variables y encadenada.	Variables seleccionadas, reglas seleccionadas y valores introducidos por el usuario para creación de la regla.	Inserción de registros en la base de datos, según el tipo de regla.	Registros OK.
Creación de regla con lado derecho de cambio de protocolo con solo variables y encadenada.	Variables seleccionadas, reglas seleccionadas, protocolo y valores introducidos por el usuario para creación de la regla.	Inserción de registros en la base de datos.	Registros OK.
Creación de regla con lado derecho de medicamentos con solo variables y encadenada.	Variables seleccionadas, medicamento y valores introducidos por el usuario para creación de la regla.	Inserción de todos los registros en las tablas correspondientes de la base de datos.	Registros OK.
Edición de regla con lado derecho de mensaje y encadenada.	Valores modificados y nuevos valores agregados.	Inserción correcta de datos.	OK.
Edición de regla cambiando el tipo de lado derecho.	Valores modificados y nuevos valores agregados y nuevo tipo de lado derecho.	Datos correctos y completos en la base de datos.	OK.

Prueba al modulo de Perfiles de Usuario:

Tabla 4. Pruebas al modulo de Perfiles de Usuario

Caso de prueba	Entrada	Salida Esperada	Resultado
Creación de un nuevo usuario.	Datos y permisos del nuevo usuario.	Inserción correcta en la base de datos.	OK.
Modificación de un usuario.	Datos y permisos modificados.	Actualización de los datos de usuario.	OK.
Eliminación del Usuario.	Usuario a eliminar.	Borrado de la base de datos.	OK.

Pruebas al modulo de copias de seguridad:

Tabla 5. Pruebas al modulo de copias de seguridad

Caso de prueba	Entrada	Salida Esperada	Resultado
Creación de copia de la base de datos.	Base de datos del sistema experto.	Archivo con extensión .zip	OK.
Creación de copia de informes.	Informes creados durante la ejecución del sistema.	Archivo con extensión .zip	OK

El siguiente paso fue la integración de todos los módulos y la realización de las pruebas, con sus casos correspondientes. Las siguientes son las tablas resultantes de las pruebas con los módulos integrados:

Pruebas al sistema integrado:

Tabla 6. Pruebas al sistema integrado

Caso de prueba	Entrada	Salida Esperada	Resultado
Acceso al módulo administración.	Evento de usuario	Visualización del módulo de administración de variables y reglas del sistema.	OK.
Acceso al módulo de copias de respaldo.	Evento de usuario	Visualización del módulo de copias de respaldo.	OK.
Acceso al módulo Sistema experto.	Evento de usuario	Visualización de la pantalla principal de ejecución del prototipo.	OK.
Acceso al módulo de administración de usuarios.	Evento de usuario	Visualización del módulo de administración de usuarios.	OK.
Definición de protocolo ejecución del sistema.	Protocolo seleccionado.	Almacenamiento del protocolo de funcionamiento.	OK.

Por ultimo se le realizo las pruebas de rendimiento al sistema, estas pruebas tienen como finalidad probar el funcionamiento bajo las condiciones extremas a las que se enfrenta en la UCI Pediátrica. La siguiente es la tabla resultante de las correspondientes pruebas:

Pruebas de rendimiento:

Tabla 7. Pruebas de rendimiento

Caso de prueba	Entrada	Condición	Rendimiento
Generación de informes	String contenido en de TextArea monitoreo.	Activo	72 h.
Monitoreo de signos y variables	Signos vitales del MSV y Consulta del a variables de la BD.	Activo	72 h.
Generación de gráficas	Registro histórico de la variable elegida.	Activo	72 h.
Análisis de ECG	Atención del paciente.	Activo	72 h.

4.7 SEGUNDO CICLO

En los últimos días de la práctica se alcanzo a participar en actividades consideradas del segundo ciclo. Estas actividades corresponden a un prototipo web del sistema experto.

Para este nuevo prototipo se diseño una nueva interfaz grafica de comunicación con el usuario en un entorno mucho más amigable que la actual.

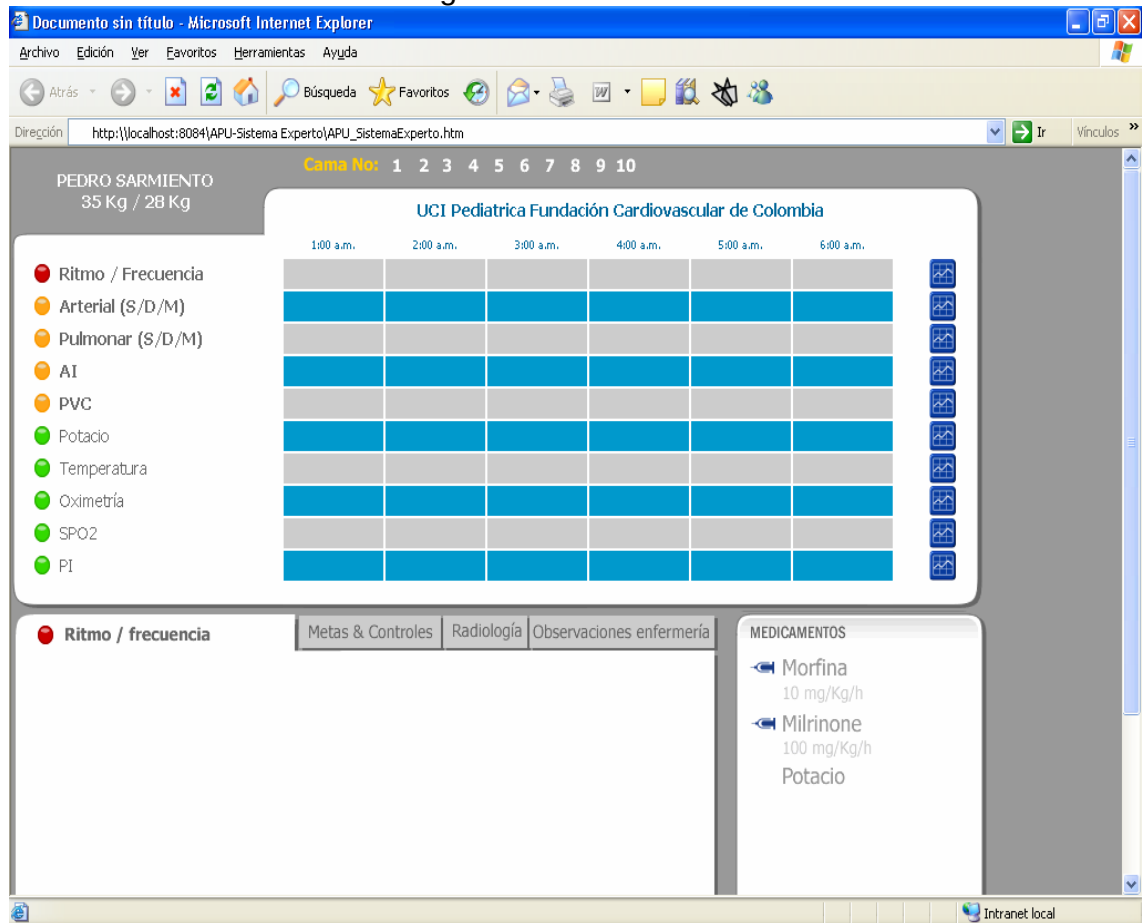
Esta interfaz web fue diseñada por un integrante nuevo del equipo de desarrollo del sistema experto y que es especialista en usabilidad, él diseñó esta nueva interfaz con el propósito de resolver algunos problemas de usabilidad, junto con otros requerimientos que los demás involucrados en el sistema le presentaron con respecto al sistema. Para el diseño de esta interfaz él se baso en las diferentes pantallas diseñadas e implementadas hasta el momento.

La nueva interfaz de comunicación con el usuario se eligió hacerla en una página web dinámica, como el lenguaje de programación en que se ha desarrollado el sistema es java, se eligió JSP (Java Server Pages, páginas de servidor de java) para implementarlas. Esto permite la reutilización de muchas de las clases creadas hasta el momento y facilita la transición entre el prototipo existente y la nueva versión.

La página web consta de 3 *frames* o 3 paginas (figura 32), el primero de estos *frames*, es el de la parte superior, donde se encuentran las variables a monitorizar, junto con sus valores de las últimas 6 horas y la posibilidad de graficar su comportamiento. Esta interfaz presenta ciertas modificaciones en cuanto a la forma de presentar los resultados generados por las reglas ejecutadas por el sistema. En la parte superior el usuario encuentra el nombre

de cada una de las variables que se están monitorizando variables que se están monitorizando. Al lado izquierdo de estas se encuentra un indicador (un círculo de color), este indicador cambia de color de acuerdo al estado de la variable, cuando se dispara alguna regla el color cambia a rojo en las variables que sean mas criticas (las reglas de mayor prioridad) y a amarillo las restantes. Cuando no se ha disparado la regla correspondiente a la variable, el indicador de esta aparecerá de color verde.

Figura 32. Interfaz web



Los otros dos *frames* se que se encuentran en la parte inferior, el de la izquierda contiene cuatro secciones a las que se se puede ingresar por medio de los cuatro hipervínculos que se encuentran en la parte superior. En la primera sección, se muestra todos los posibles diagnósticos relacionados con la variable que se haya seleccionado en el *frame* superior, además de que muestra las demás variables que corresponden a cada diagnostico, junto con los rangos definidos y el valor actual de cada una. En la segunda sección que se llama metas y controles, se encuentran todas las variables con sus correspondientes rangos para ser modificados por el usuario. La tercera sección es la correspondiente a radiología, el usuario puede visualizar las diferentes radiografías que se la han tomado al paciente junto con el análisis del radiólogo. En la última sección (observaciones de enfermería) se pueden ver las sugerencias que han sido enviadas por las reglas que se han activado.

4.7.1 Implementación

Para esta interfaz se implementó la página de la parte inferior izquierda (figura 33). Al momento que el usuario selecciona alguna de las variables que se encuentra en el *frame* superior, se envía hacia la pagina cual variable fue seleccionada en el *frame* superior, junto con la identificación del usuario y un indicador que es a la primera sección que se quiere acceder. Una vez cargada la página muestra lo que se puede apreciar en la figura 33, para que esto sea posible, la pagina jsp consulta en la base de datos todos los diagnósticos relacionados con esa variable, también busca los rangos que se han establecido para cada una las variables de cada diagnostico y los muestra; por último consulta la base de datos de la Historia Clínica Electrónica (HCE) y busca el valor actual de estas variables y si no es encontrado, indica al usuario que no existe el registro.

Una vez el usuario decida cual es el diagnóstico del paciente, podrá seleccionarlo haciendo click sobre éste y se le presenta una ventana preguntándole si ese es el diagnostico que desea guardar, si el usuario acepta se guarda el diagnostico junto con la hora y la fecha en que éste fue seleccionado.

Figura 33. Diagnósticos asociados a una variable

PH				
Metas & Controles				
Radiología				
Observaciones enfermería				
Diagnostico	Variables Asociadas			Valor actual
	Limite inferior	Variable	Limite superior	
enfermedad pulmonar crónica	12	PAP	15	No registra
	7	PAI	10	No registra
	6	pH	7	7.462
	-4	BE	4	9.4
ictericia por hiperbilirubinemia	7	PAI	10	No registra
	6	pH	7	7.462
	1	PO2	2	107.2
	90	sO2C	95	98.7
Sepsis neonatal	35	Temperatura	36	No registra
	6	pH	7	7.462

Cuando el usuario hace clic en el hipervínculo de Metas Y Controles, se buscan los rangos que tienen definidas las variables y se le presentan al usuario en cajas de texto para su modificación. Una vez el usuario termine de modificar los rangos, este debe oprimir el botón aceptar, el cual actualizara los rangos establecidos para las variables y el sistema pueda enviar al motor de inferencia estos datos en forma de hechos y se puedan evaluar correctamente las reglas.

Figura 34. Metas y controles

Temperatura			Metas & Controles	Radiología	Observaciones enfermería
Limite inferior	Variable	Limite superior			
9	PAM	12			
6	PVC (mmHg)	7			
12	PAP	15			
7	PAI	10			
35	Temperatura	36			
12	P.A.S. (mmHg)	14			
6	pH	7			
35	PCO2	38			
-4	BE	4			
1	PO2	2			
0.5	Ca++(Ionico)	3			
90	sO2C	95			

En el hipervínculo de Radiología (Figura 35) el usuario podrá observar las diferentes radiografías que se le han tomado al paciente en una versión reducida, si el desea ver alguna de estas radiografías en su tamaño natural, el puede hacer click encima de la radiografía que desea apreciar con mas claridad, esto hace que una ventana emergente aparezca mostrando la imagen de la radiografía en el tamaño real que esta tenga (Figura 36).

Figura 35. Radiología

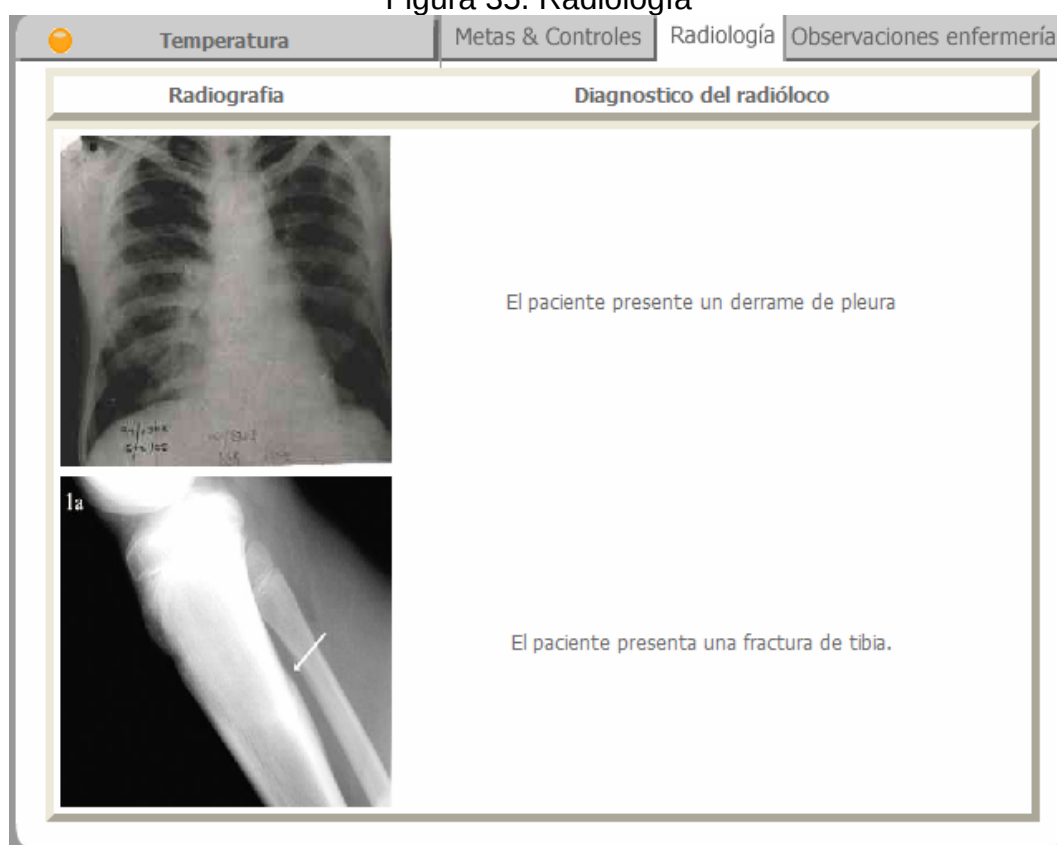
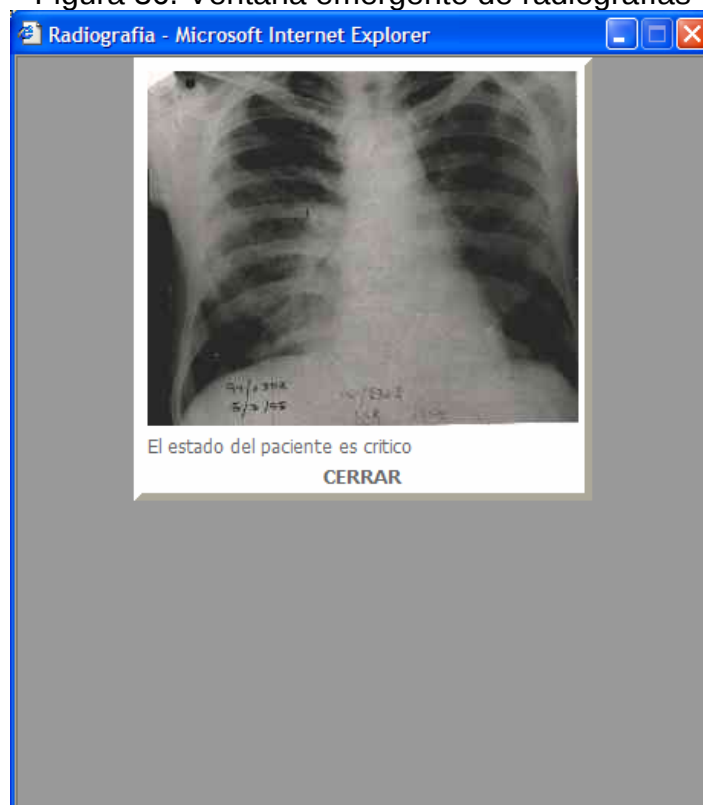


Figura 36. Ventana emergente de radiografías



También se implemento parte de la pagina del *frame* que se encuentra en la parte inferior derecha, en esta pagina el usuario podrá ver los medicamentos sugeridos (que cambian según el tiempo que lleve el paciente en la UCI), junto con el suministro del medicamento. Éste suministro es obtenido mediante la formula: $(\text{peso} * \text{dosis}) / \text{concentración}$, para esto la pagina jsp realiza una consulta a la base de datos de la HCE para obtener el peso actual del paciente, luego busca el valor de la concentración del medicamento en la base de datos del sistema y con una dosis preestablecida por protocolo (la pagina web solo esta implementada para el protocolo de tga) se realiza el correspondiente calculo.

El usuario podrá indicar en pantalla cuales de los medicamentos le va a suministrar al paciente, simplemente señala cuales de los medicamentos sugeridos desea aplicar y da clic sobre la imagen que se encuentra en la parte inferior de la figura 37. Los medicamentos sugeridos aparecen por debajo de la linea punteada y los suministrados sobre ésta.

Figura 37. Página de medicamentos

The screenshot shows a mobile application interface titled 'MEDICAMENTOS'. It features a list of four medications, each with a checkbox, a syringe icon, the medication name, and its dosage. A horizontal dashed line separates the medications above it from those below it. At the bottom, there are two icons: a blue hand icon and a green checkmark icon, followed by a 'Suministrar' button.

Medicamento	Dosis
☐ MORFINA	0.128mm/Kg/h
☒ MILRINONE	0.24mm/Kg/h
☐ GLICERINA	0.16mm/Kg/h
☐ ADRENALINA	0.06mm/Kg/h

4.7.2 Diseño de páginas web para la administración del sistema

En los últimos días de práctica también se logro participar en el diseño de algunas de las interfaces web para la administración del sistema. Estas interfaces son: administración de reglas, administración de variables y opciones del sistema.

Figura 38. Opciones del sistema



En la figura 38 se aprecia la interfaz de opciones del sistema versión web, en se observan las diferentes opciones del sistema en la parte derecha de la pantalla. Esta página presenta un entorno muy amigable para el usuario además de que la forma de entrar a cada una de las diferentes opciones del sistema es muy llamativa.

Figura 39. Administración de reglas



El diseño de la interfaz de administración de reglas versión web, lo podemos apreciar en la figura 39. Aquí se puede observar en la parte izquierda cada uno de los protocolos, allí al señalar alguno de ellos se muestran cada una de las diferentes reglas que tiene dicho protocolo. En la parte superior de la página se encuentran las variables de la regla y las reglas asociadas a esta, también se pueden apreciar los valores disponibles para cada una de las variables para

que se seleccione cual valor es el que se debe cumplir para que se ejecute la regla. También se aprecian en la parte inferior los tipos de lado derecho que pueden tener las reglas.

Figura 40 Administración de variables

Presión venosa central		42 - 80		80 - 120		120 - 150	
Edad	Talla	Límite inf.	Límite sup.	Límite inf.	Límite sup.	Límite inf.	Límite sup.
0 - 1							
1 - 2							
2 - 3							
3 - 4							
4 - 5							
5 - 6							
6 - 7							
7 - 8							
8 - 9							
9 - 10							
10 - 11							

La administración de variables que se aprecia en la figura 40, muestra una manera de configurar una variable para el sistema, al lado izquierdo se encuentran las variables agrupadas según el sistema al cual pertenecen, para seleccionar y configurar.

4.8 OTRAS ACTIVIDADES

Durante la realización de la práctica se realizaron otras actividades como complemento. La primera de estas actividades fue la búsqueda de un IDE para java, esto con el fin de encontrar una herramienta que agilizara la elaboración de las diferentes clases, en especial las que eran graficas, como los formularios y los diálogos.

Una actividad que se realizó con el fin de la publicación de un artículo, fue la revisión y búsqueda de revistas indexadas donde se pudiera realizar su publicación.

Otra actividad fue la participación en una capacitación acerca de puntos de función, que se dictó en la Fundación Cardiovascular a todo el personal relacionado con el área de software. [Anexo G]

Debido a que el proyecto de sistema experto se desarrolla dentro de un grupo de investigación y al estar haciendo parte de él, a los practicantes se les asignó algunas labores típicas del grupo de investigación. Una de estas actividades es la generación de ideas de proyectos que generen conocimiento útil para el grupo o que agilicen procesos internos del grupo. Por esto los practicantes plantearon un proyecto acerca de los CRF's [Anexo H]

La Fundación Cardiovascular de Colombia internamente declaró el 2006 como el año de la investigación. Con el fin de incentivar a las personas a que realicen investigación, se realizó un convenio con el SENA para ofrecer la posibilidad de asistir a un diplomado sobre investigación en el cual se participó con la intención de aprender nuevas cosas útiles para el desarrollo como persona y como profesional.

El área de investigación de pediatría dictó un curso de bioestadística, al curso se asistió con el ánimo de reafirmar conocimientos adquiridos en el transcurso de la carrera en el área de estadística, para ver como en otras disciplinas le dan utilidad a la estadística y para ver el manejo de ciertas herramientas para el análisis estadístico que se deseaban conocer para poder presentar la propuesta de proyecto de CRF's.

Finalmente se colaboró en la elaboración de un artículo acerca del sistema experto que se ha venido desarrollando. Este artículo fue aceptado en el evento CMPI 2006 Campus multidisciplinar en percepción e inteligencia 2006 en Albacete España, para ser presentado. [Anexo I]

5 CONCLUSIONES.

Las áreas del conocimiento en donde resulta mas conveniente aplicar el uso de la tecnología de los sistemas expertos, es en aquellas en donde se tiene un profundo conocimiento o se encuentra muy bien estructurado, de manera que se facilita su representación por las técnicas usadas por la IA. Es por esto que resulta muy útil para resolver problemas en el área médica.

Con las actividades realizadas a través del desarrollo del prototipo de sistema experto, se logró afianzar los conocimientos adquiridos a lo largo de los niveles cursados en la carrera de ingeniería de sistemas y además adquirir nuevos conocimientos, en especial en el campo de la inteligencia artificial conocido como sistemas expertos.

La metodología de Proceso Unificado utilizada para el desarrollo de la práctica, es una metodología muy útil para el desarrollo de proyectos ya que al tener una temprana retroalimentación conlleva a construir un sistema que se ajusta mas a las necesidades del personal involucrado.

Los sistemas expertos son herramientas muy útiles para asistir procesos en los cuales el error no es admisible, en especial en áreas tan críticas como la médica, donde el más mínimo error puede causar la perdida de una vida; debido a que poseen una capacidad para manejar mayores volúmenes de información que un humano, lo que hace posible que tenga en cuenta todas las variables a manejar, cosa que la mente humana no esta en capacidad de realizar.

6 PERSPECTIVAS HACIA EL FUTURO

El desarrollo del primer prototipo sentó las bases para que la Fundación Cardiovascular de Colombia prosiga con el proyecto hasta generar un producto que aprenda de la experiencia adquirida (hechos en la base de datos) y pueda ayudar mucho más al experto en el área médica.

Si esto se quiere lograr, aun falta camino por recorrer, primero hay que implementar otros módulos importantes, como el modulo de aprendizaje. Para que el sistema llegue a aprender, es necesario llevar el sistema experto de uno basado únicamente en reglas a uno basado en probabilidad. Existen varias técnicas para que un sistema experto aprenda, entre estas se encuentran, las cadenas de Harkov, las redes neuronales y las maquinas de soporte vectorial.

Un proyecto como este, puede llegar a ser extendido y abarcar no solo la UCI Pediátrica sino también, otras áreas críticas como la UCI de adultos y Urgencias, sitios donde los sistemas basados en IA también pueden lograr reducir el error humano, ayudando en la toma de decisiones que pueden tener un impacto positivo en calidad de la asistencia medica.

Este proyecto también en un futuro llegara a ser un aporte a la educación, al favorecer la interacción de diferentes disciplinas, ya que este proyecto requiere de mucho trabajo interdisciplinar; adicionalmente la oportunidad de reunir personal con diferente nivel de formación (pregrado, prepostgrado y maestría) constituirá una gran fortaleza pues se podrá contribuir tanto en los procesos de enseñanza de medicina como de ingeniería.

BIBLIOGRAFÍA

SÁNCHEZ, Aitor San Juan. Sistema CLIPS (Apuntes).

ROLSTON, David. Principios de Inteligencia Artificial y Sistemas Expertos. Sunnydale: McGraw-Hill

M. Ortín, J. García Molina, B. Moros, J. Nicolás. El Modelo del Negocio como base del Modelo de Requisitos.

FACULTAD DE CIENCIAS, Universidad de Ámsterdam [en línea].
<http://www.cs.vu.nl/~mrmenken/jclips/>.

WIKIPEDIA. <http://es.wikipedia.org/wiki/Portada>.

UNIVERSIDAD DE VIRGINIA, Health System
http://www.healthsystem.virginia.edu/UVAHealth/peds_cardiac_sp/tga.cfm.

CORREDOR, Martha Vitalia. Princípios de Inteligência Artificial y Sistemas Expertos. Bucaramanga, Ediciones UIS. 1992.

J C, Giarratano, Ph.D. CLIPS Version 6.20 User's Guide. March 31st 2002

CLIPS Reference Manual Volume I Basic Programming Guide Version 6.2. June 1st 2005.

CLIPS Reference Manual Volume II Advanced Programming Guide Version 6.23. January 31st 2005.

Anatomia Humana Sistemica Aplicada. [en línea] Aparato Circulatorio. Disponible en http://www.uc.cl/sw_educ/anatnorm/acirculat/glosario.htm. Acceso el 10 de abril 2006

Tutorial de uml.[en línea] Disponible en <http://www.dcc.uchile.cl/~psalinas/uml/introduccion.html>. Acceso el 10 de mayo 2006

TORO, Amador Metodología para la Elicitación de Requisitos de Sistemas Software

PIATTINI, Mario y otros. Análisis y diseño detallado de Aplicaciones Informáticas de Gestión.

PINILLA, Lucenith y PARADA, Yasmid. Sistema basado en el conocimiento para la reanimacion de pacientes politraumatizados "ABC 1.0"

[Tesis].Bucaramanga: Universidad Industrial de Santander. Escuela de Ingeniería de Sistemas 2002 87p

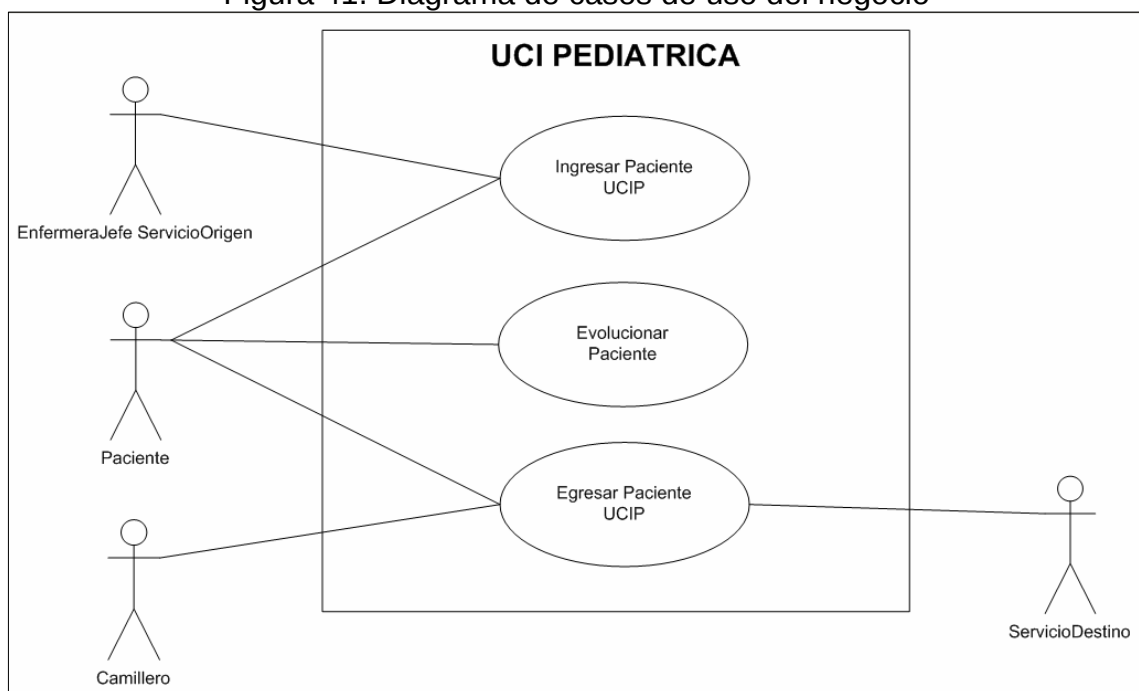
J: Mira. DIAVAL, sistema experto bayesiano para ecocardiografía.

ANEXOS

ANEXO A. DIAGRAMAS UML DEL MODELADO DEL NEGOCIO

En la figura 41 se puede observar el diagrama de casos de uso al que se llegó en el modelado del negocio, aquí se aprecian los tres casos de uso que representan los procesos del negocio que en este caso es la UCI pediátrica. Estos casos de uso son: el caso de uso de ingreso a la UCI pediátrica, el caso de uso de la evolución del paciente y el caso de uso de egreso de la UCI pediátrica.

Figura 41. Diagrama de casos de uso del negocio



Cada caso de uso tiene una descripción detallada de los actores que participan, las precondiciones, poscondiciones, sus principales escenarios y sus flujos alternativos. Esta descripción de cada uno de los casos de usos es la presentada a continuación:

CASO DE USO - INGRESAR PACIENTE UCIP

Nombre del Caso de uso: Ingresar paciente UCIP.

Actor Principal: Médico de la UCI Pediátrica.

Personal Involucrado e Intereses:

- Médico de la UCI Pediátrica: Desea recibir información oportuna del estado del paciente, conocer su procedencia y antecedentes para emitir un plan de manejo adecuado.
- Personal Asistencial: Quieren conocer el estado del paciente y sus necesidades, para la disposición del cubículo.
- Servicio Externo Remitente: Quiere remitir el paciente a la UCI Pediátrica y confirmar la remisión del paciente a la UCI Pediátrica.
- Clínica: Quiere brindar la mejor atención al paciente, satisfaciendo sus necesidades. Quiere asegurarse de que se registre el estado de ingreso del paciente a la UCI Pediátrica.

Precondiciones: La auxiliar de admisiones o el servicio de urgencias confirma ingreso del paciente indicando procedencia.

Garantía de éxito (Postcondiciones): El paciente es ingresado a UCI Pediátrica y registrado su estado en HC (Historia Clínica) y HCE (Historia Clínica Electrónica), el cubículo es dispuesto y se emite un plan de manejo acorde a las necesidades, por parte del médico.

Escenario Principal de Éxito:

1. El médico de turno recibe telefónica mente o vía fax la información del estado del paciente.
2. Se informa de las condiciones generales del paciente a la enfermera y auxiliar encargadas de recibirlo.
3. La auxiliar de enfermería, dota el cubículo asignado, con los elementos y equipos necesarios, mediante la guía de dotación del mismo.
4. A la llegada del paciente, la enfermera y la auxiliar reciben al paciente y lo monitorizan. La enfermera registra su ingreso en el proceso de enfermería de la HCE y la auxiliar de enfermería realiza el registro de parámetros hemodinámicos, ventilación, y control de líquidos administrados y eliminados, en los registros establecidos dentro de la historia clínica
5. El médico de turno realiza el ingreso teniendo en cuenta el interrogatorio, revisión de historia clínica, y examen físico.
6. El médico de turno registra en HC el diagnóstico, la categoría, objetivo, y el concepto en el cual estipula el criterio de entrada, y el plan de manejo.
7. El médico de turno realiza órdenes de medicamentos, hemoderivados, nutrición, procedimientos, controles y actividades.

Extensiones (o flujos alternativos):

- 2^a. El paciente es procedente de otro servicio de la institución.
 1. La enfermera del servicio remitente, informa a la enfermera encargada de la unidad, el estado del paciente.
- 2^b. El paciente proviene de otra institución.
 1. El médico de turno, informa las condiciones generales del estado del paciente a la enfermera.

CASO DE USO - EVOLUCIONAR PACIENTE UCIP

Nombre del caso de uso: Evolucionar paciente.

Actor Principal: Médico de la UCI Pediátrica, Personal Asistencial.

Personal Involucrado e Intereses:

- Médico de la UCI Pediátrica: Quiere recibir de modo oportuno información acerca de los cambios súbitos en el estado del paciente. Quiere contar con una monitoría constante de las variables relevantes para un paciente, de acuerdo a la patología que presenta el mismo. Quiere tener a su disposición información histórica que le permita evaluar el comportamiento del estado de salud del paciente. Quiere poder determinar un plan de manejo adecuado para cada situación particular que se presente.
- Personal Asistencial: Quieren contar con instrumentos que le suministren información acerca del comportamiento de las variables a monitorizar en un paciente. Quieren hacer un seguimiento a la evolución de cada paciente. Quieren suministrar los medicamentos ordenados por el médico en el momento oportuno. Quiere registrar los cambios que se presenten en el paciente.
- Clínica: Quiere brindar la mejor atención al paciente, satisfaciendo sus necesidades. Quiere que el personal médico y asistencial, cuente con las herramientas necesarias para cumplir su labor.
- Secretaría de Salud: Quiere ejercer control sobre los medicamentos que son suministrados a los pacientes

Precondiciones: El paciente ha sido ingresado a la UCI Pediátrica.

Garantía de éxito (Postcondiciones): El paciente es monitorizado continuamente permitiéndole al médico tomar decisiones oportunas y en muchos casos anticipadas a las complicaciones que puedan presentarse.

Escenario Principal de Éxito:

1. La Revista Médica se realiza en cada uno de los turnos con la participación de todo el grupo interdisciplinario.
2. El personal interdisciplinario visita uno a uno los pacientes de la unidad, valorando su estado de salud y emitiendo juicios sobre el manejo,
3. El médico diligencia los registros de evolución y ordenes médicas de la historia clínica, teniendo en cuenta la valoración y recomendaciones de la revista.
4. Las enfermeras, terapeutas respiratorias, auxiliares de enfermería y nutricionistas ejecutan los procedimientos descritos relacionados a la atención del paciente hospitalizado en la unidad, que conlleven a la satisfacción de necesidades de cada paciente.
5. El personal asistencial de la UCI Pediátrica, notifica al médico de turno todo tipo de cambio que presente el paciente para que él a su vez, valore al paciente, ordene y se realicen procedimientos que contribuyan a la estabilización del mismo.

6. El personal de enfermería debe notificar y registrar los cambios, así como las acciones tomadas, en los registros de la historia clínica establecidos, con hora y fecha.

CASO DE USO - EGRESAR PACIENTE UCIP

Nombre caso de uso: Egresar Paciente UCIP.

Actor Principal: Médico UCIP, Personal Asistencial UCIP

Personal Involucrado e Intereses:

- Médico UCIP: Quiere evaluar correctamente al paciente, para determinar si puede abandonar la UCIP.
- Personal Asistencial: Quiere realizar los trámites administrativos y clínicos del caso, de manera eficiente.
- Servicio Receptor: Quiere recibir el paciente y contar con la información oportuna del estado del mismo.
- Clínica: Quiere trasladar de manera eficiente al paciente, al destino particular sea este un servicio interno de la misma u otra Institución.
- Medicina Legal: Quiere corroborar las causas del fallecimiento del paciente, en caso de que las circunstancias así lo requieran.

Precondiciones: Orden de traslado por parte del médico de turno en historia clínica.

Garantía de éxito (Postcondiciones): Se realizan los registros pertinentes en historia clínica, trámites administrativos y procedimientos clínicos y el paciente es remitido al destino.

Escenario Principal de Éxito:

1. La enfermera o auxiliar de enfermería solicita a la secretaria del servicio la asignación de cama en el servicio de hospitalización.
2. La auxiliar de enfermería se encarga de la preparación del cubículo.
3. La enfermera encargada del paciente procederá a la preparación del traslado del paciente realizando procedimientos clínicos y administrativos.
4. La enfermera hace entrega del paciente telefónicamente a la enfermera encargada del servicio receptor y traslado de la historia clínica electrónica al servicio de destino.
5. La enfermera entrega verbalmente al paciente y envía epicrisis.
6. La auxiliar de enfermería prepara los elementos necesarios para el traslado junto con las pertenencias del paciente y la enfermera los verifica.
7. Se entregará al familiar reportes de laboratorio y de exámenes realizados durante su estancia en la Unidad.

Extensiones (o flujos alternativos):

- 1^a. El paciente ha fallecido.
 1. El médico de turno confirmará el fallecimiento mediante su valoración.
 2. Informa a la familia o responsable el deceso.

3. Diligencia el certificado de defunción y la epicrisis en la Historia clínica electrónica.
4. La enfermera realiza el trámite administrativo de entrega del cadáver.
 - 1^a. Se desconoce la causa del fallecimiento y el paciente no procede de otra PIS.
 1. El medico debe contactar a medicina legal
 - 2^a. El paciente se traslada al servicio interno de hospitalización.
 - 2^b. El paciente se traslada a otra Institución.
 1. La enfermera coordina el traslado del paciente.
 2. La secretaria solicita el servicio de ambulancia.

Para el modelado del negocio se crearon algunos escenarios para mostrar el aspecto de comportamiento de la colaboración e interacción entre los actores que hacen parte de cada caso de uso. Para esto se realizaron los diagramas de secuencia donde los objetos denotan las instancias de los roles que intervienen en la interacción. En cada proceso se puede hacer un diagrama de secuencias del escenario principal y del flujo alternativo (donde sea necesario) para mejorar el entendimiento de los diferentes procesos.

Para nuestro negocio se hicieron diagramas de secuencia y de actividades, apreciando en ellos los diferentes procesos para el ingreso a la UCI Pediátrica, para la evolución del paciente, para el egreso de la UCI Pediátrica y para el flujo alternativo de este que es cuando el paciente fallece. Los diagramas de secuencia y actividades que se obtuvieron son los que se aprecian en las siguientes figuras.

Figura 42. Diagrama de secuencias para el Ingreso a la UCIP

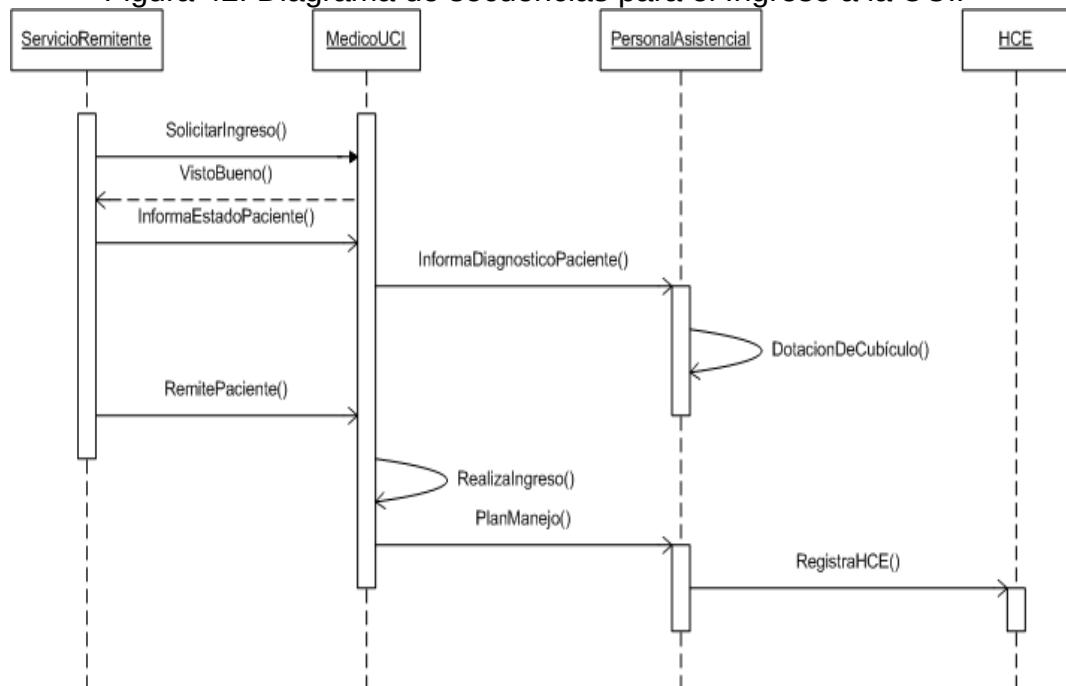


Figura 43. Diagrama actividades de ingreso a la UCIP

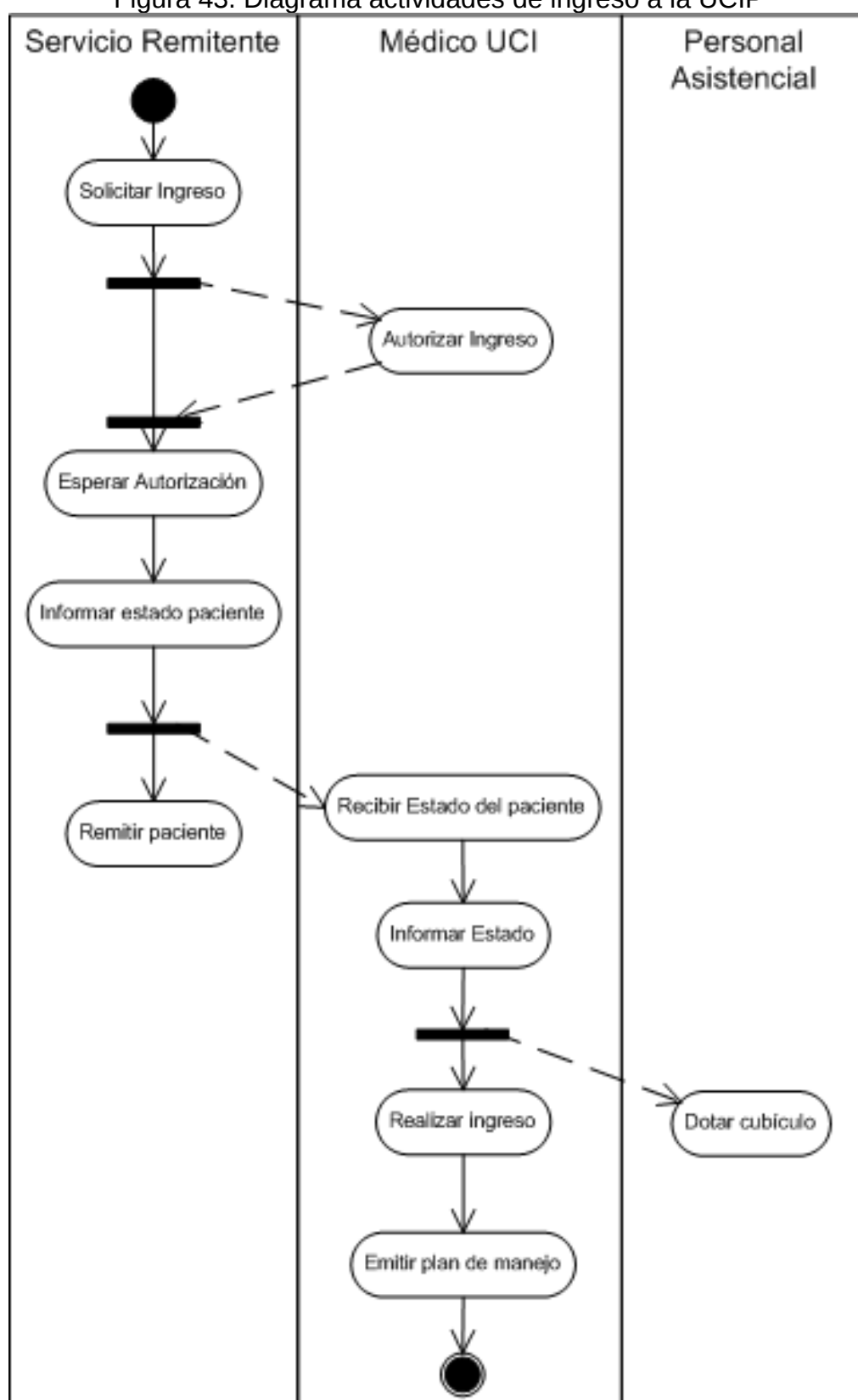


Figura 44. Diagrama de actividades evolucionar paciente

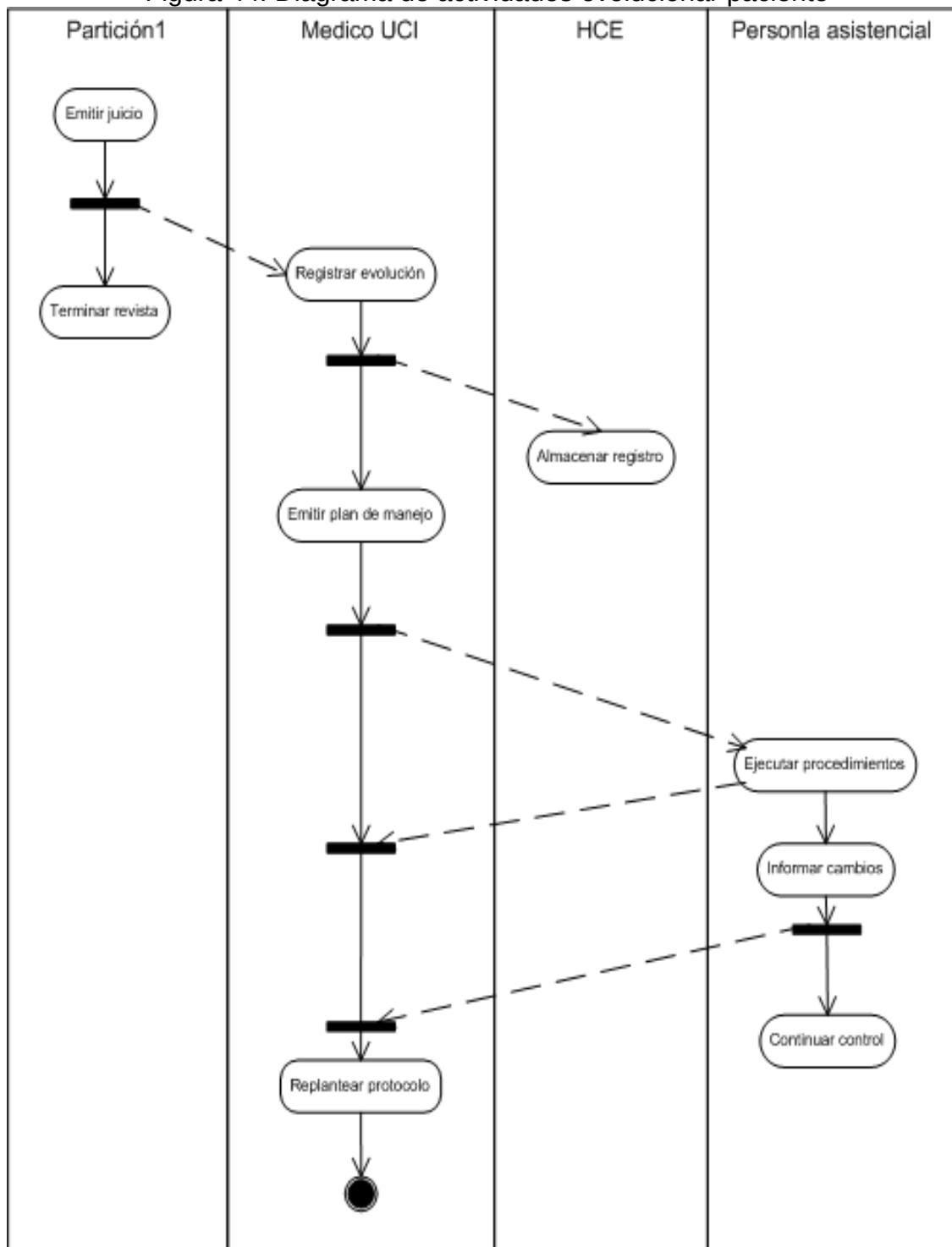
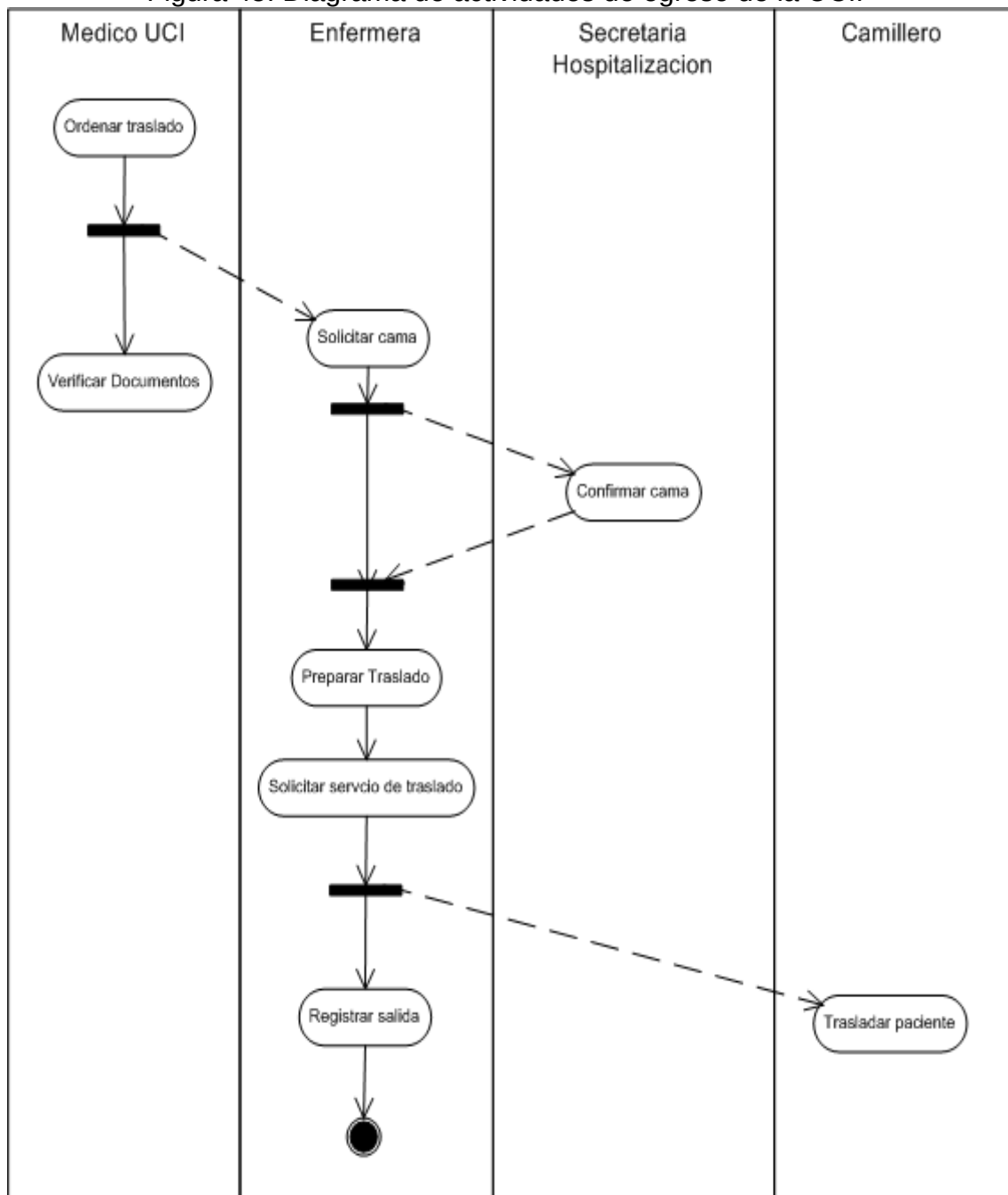


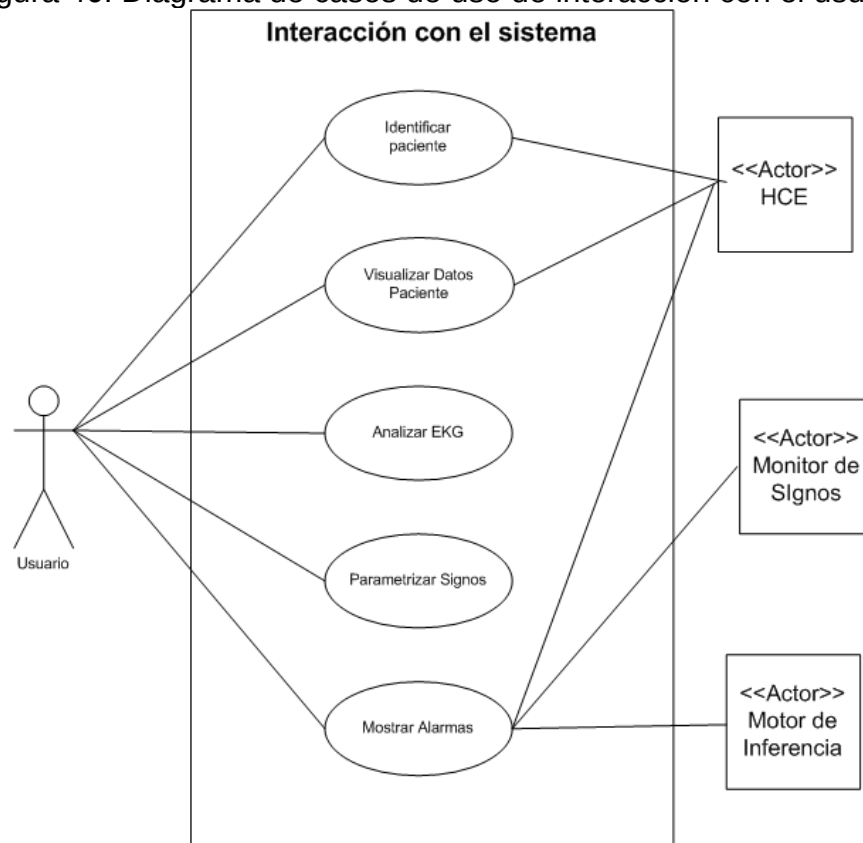
Figura 45. Diagrama de actividades de egreso de la UCIP



ANEXO B. DIAGRAMAS UML DE INTERACCIÓN CON EL USUARIO

El diagrama de casos de usos resultante en la interacción con el usuario fue:

Figura 46. Diagrama de casos de uso de interacción con el usuario



Las descripciones de cada uno de los casos de uso son las siguientes:

CASO DE USO – IDENTIFICAR PACIENTE.

Nombre del Caso de uso: Identificar Paciente.

Actor Principal: Medico UCIP (usuario).

Personal Involucrado e Intereses:

- Medico de la UCIP: Desea ingresar al sistema la atención del paciente para iniciar el funcionamiento del sistema experto.
- FCV: Quiere brindar la mejor atención al paciente, satisfaciendo sus necesidades.

Precondiciones: El usuario necesita haber ejecutado la clase en java que da inicio al sistema experto

Garantía de éxito (Postcondiciones): El paciente es identificado para poder conocer su identidad, cirugías a las que se ha sometido y obtener los hechos provenientes de la historia clínica electrónica.

Escenario Principal de Éxito:

8. El usuario busca el número de atención al paciente.
9. El usuario digita la atención del paciente.
10. El sistema busca en la base de datos de la historia clínica electrónica al paciente.
11. Es identificado el paciente y se guardan sus datos y la atención en el sistema.

Extensiones (o flujos alternativos):

3ª. La atención del paciente no existe

1. El sistema vuelve a pedir la atención hasta que se digite una atención válida.

CASO DE USO – ANALIZAR EKG.

Nombre del Caso de uso: Analizar EKG.

Actor Principal: Medico UCIP (usuario).

Personal Involucrado e Intereses:

- Medico UCIP: Desea indicarle al sistema cuando será su fin de turno y responder las preguntas referentes al electrocardiograma.
- FCV: Quiere brindar la mejor atención al paciente, satisfaciendo sus necesidades.

Precondiciones: Haber iniciado el sistema y haber digitado una atención de un paciente que sea válida.

Garantía de éxito (Postcondiciones): El responde las preguntas correspondientes y el sistema espera hasta que se acabe el turno para nuevamente pedir la fecha de fin de turno y preguntar.

Escenario Principal de Éxito:

1. El sistema muestra un dialogo donde introducir la fecha y hora del fin de turno.
2. Es introducida la hora y la fecha de fin de turno por el usuario
3. Aparece el dialogo con las preguntas correspondientes al electrocardiograma.
4. El usuario responde las preguntas correspondientes.
5. El sistema guarda las respuestas en variables.

Extensiones (o flujos alternativos):

2ª. El usuario digita una fecha no válida.

1. El sistema le dice al usuario que la fecha no es válida
2. El sistema pide nuevamente la fecha y hora del fin de turno hasta que esta sea válida.

CASO DE USO – VISUALIZAR DATOS PACIENTE.

Nombre del Caso de uso: Visualizar Datos Paciente.

Actor Principal: Medico UCIP (usuario).

Personal Involucrado e Intereses:

- Medico de la UCIP: Desea conocer el nombre del paciente, a que cirugías ha sido sometido y modificar los valores considerados como normales de las presiones y la temperatura.
- FCV: Quiere brindar la mejor atención al paciente, satisfaciendo sus necesidades.

Precondiciones: Haber introducido una atención del paciente que sea valida.

Garantía de éxito (Postcondiciones): El usuario puede identificar claramente al paciente y conocer las intervenciones quirúrgicas hechas al paciente.

Escenario Principal de Éxito:

1. El sistema muestra al usuario el nombre del paciente y las intervenciones quirúrgicas hechas.
2. El usuario identifica al paciente y sus intervenciones quirúrgicas.

Extensiones (o flujos alternativos):

- 2ª. El paciente no es el correcto.
1. El usuario rectifica la atención.
 2. El usuario reinicia el sistema.

CASO DE USO – PARAMETRIZAR SIGNOS.

Nombre del Caso de uso: Parametrizar Signos.

Actor Principal: Medico UCIP (usuario).

Personal Involucrado e Intereses:

- Medico de la UCIP: Desea modificar los valores considerados como normales de las presiones y la temperatura.
- FCV: Quiere brindar la mejor atención al paciente, satisfaciendo sus necesidades.

Precondiciones: Aparecer el dialogo de visualizar datos del paciente.

Garantía de éxito (Postcondiciones): El usuario puede identificar claramente al paciente, conocer las intervenciones quirúrgicas hechas y parametriza las presiones y la temperatura.

Escenario Principal de Éxito:

1. El usuario busca las opciones del menú.
2. El usuario elige que parametrizar, temperatura o presiones.
3. El usuario modifica los valores.
4. El sistema cambia los valores antiguos por los introducidos por el usuario.

Extensiones (o flujos alternativos):

- 2^a. El usuario escoge modificar las presiones.
 3. El usuario modifica las diferentes presiones.
 4. Acepta los cambios.
- 2^b. El usuario escoge modificar la temperatura.
 1. El usuario modifica la temperatura.
 2. Acepta los cambios.

CASO DE USO – MOSTRAR ALARMAS.

Nombre del Caso de uso: Mostrar Alarmas.

Actor Principal: Medico UCIP (usuario).

Personal Involucrado e Intereses:

- Medico de la UCIP: Desea conocer todas las alarmas generadas por los signos medidos por el monitor y los datos registrados en la historia clínica electrónica. También desea conocer los medicamentos sugeridos.
- FCV: Quiere brindar la mejor atención al paciente, satisfaciendo sus necesidades.

Precondiciones: Haber introducido una atención del paciente que sea valida y contestado las preguntas del EKG.

Garantía de éxito (Postcondiciones): El usuario puede ver las alarmas generadas y así darle una mejor atención al paciente.

Escenario Principal de Éxito:

1. El sistema envía los signos de la base de datos en forma de hechos al motor de inferencia.
2. El motor de inferencia ejecuta las reglas correspondientes.
3. El motor de inferencia envía las respuestas al sistema
4. El sistema envía las respuestas al usuario en un lenguaje comprensible, mostrándolo en la parte media del formulario.
5. El sistema envía los signos obtenidos del monitor de signos vitales al monitor en forma de hechos al motor de inferencia.
6. El sistema envía los valores de los rangos de las presiones y de la temperatura en forma de hechos al motor de inferencia.
7. El motor de inferencia ejecuta las reglas activadas con los hechos correspondientes.
8. El motor de inferencia devuelve las respuestas al sistema.
9. El sistema envía las respuestas obtenidas del motor de inferencia al usuario en un lenguaje comprensible para el, en la parte superior del formulario.
10. El sistema envía las variables con las respuestas de las preguntas del EKG en forma de hechos al motor de inferencia.
11. El motor de inferencia ejecuta las reglas que se activaron con los hechos enviados.

12. EL motor de inferencia devuelve los resultados obtenidos de la ejecución de las reglas.
13. El sistema transforma las respuestas a un lenguaje entendible por el usuario en la parte central del formulario.
14. El sistema envía los resultados al usuario.
15. El sistema le envía al usuario los diferentes medicamentos sugeridos para usar con el paciente y los muestra en la parte inferior del formulario.

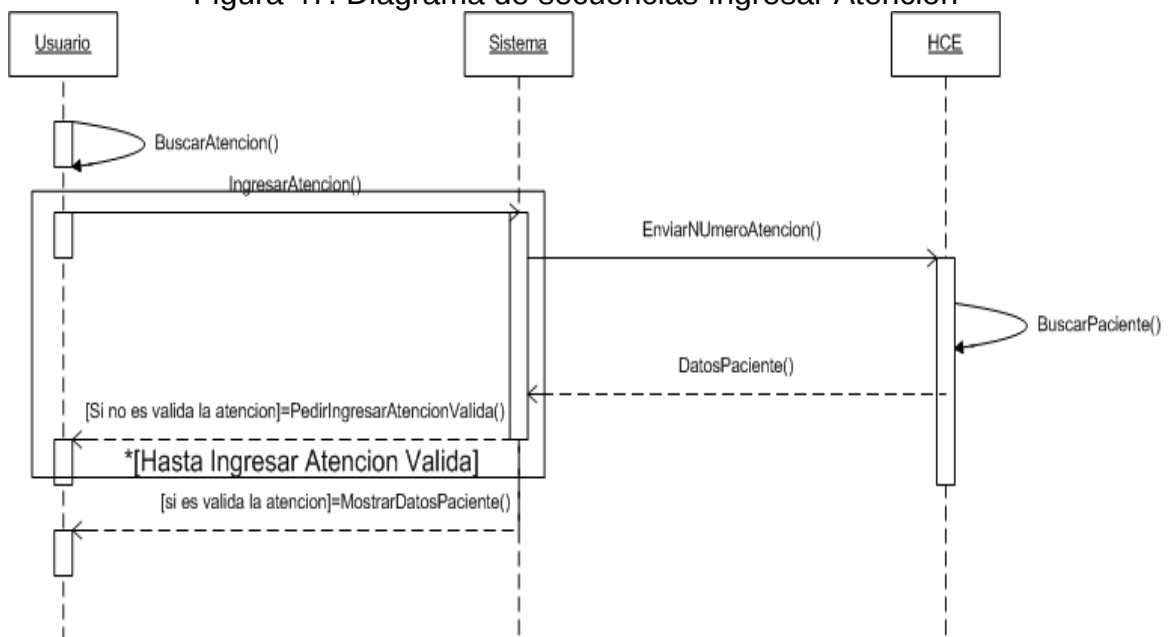
Extensiones (o flujos alternativos):

- 2ª. El sistema indica algún signo faltante en la base de datos.
 1. El usuario identifica que signo falta.
 2. Se introduce el valor faltante.
 3. El sistema genera las alarmas faltantes.

Los diagramas de secuencias de la interacción con el usuario resultantes fueron los siguientes:

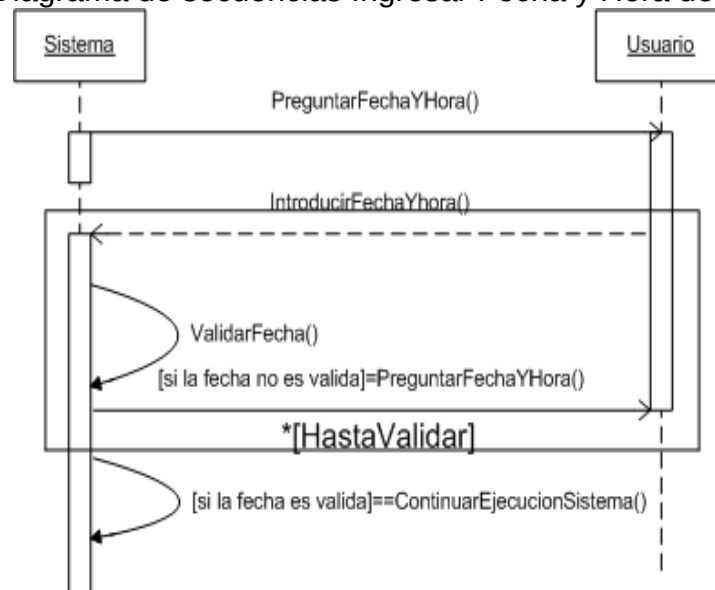
El diagrama de secuencias Ingresar Atención muestra la interacción del usuario con el sistema cuando este ingresa la atención.

Figura 47. Diagrama de secuencias Ingresar Atención



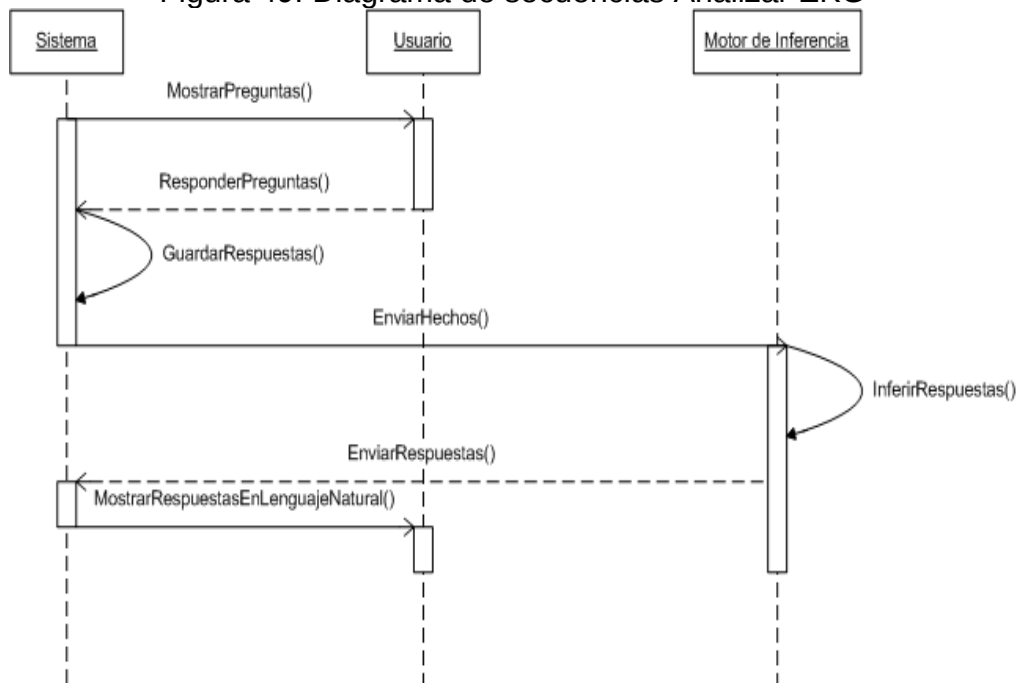
En el diagrama de secuencias Ingresar Fecha y Hora de Fin de Turno, se puede observar el funcionamiento del sistema en el momento que el usuario ingresa la hora y la fecha (hasta que sea valida) en el sistema.

Figura 48. Diagrama de secuencias Ingresar Fecha y Hora de Fin de Turno



El diagrama de secuencias Analizar EKG, deja ver como interactúa el usuario con el sistema al momento en que este le pide que conteste unas preguntas sobre el electrocardiograma (EKG).

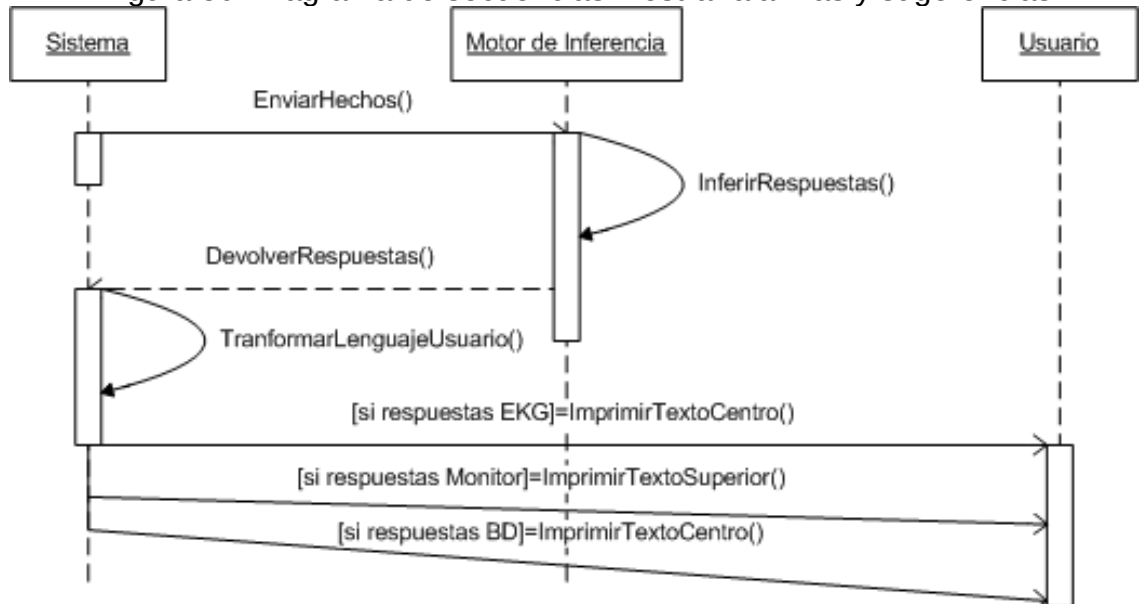
Figura 49. Diagrama de secuencias Analizar EKG



En el siguiente diagrama se puede apreciar la secuencia cuando el sistema muestra las diferentes alarmas y sugerencias al usuario. Esta secuencia actúa igual en cualquier tipo de alarmas, la única diferencia es que según donde

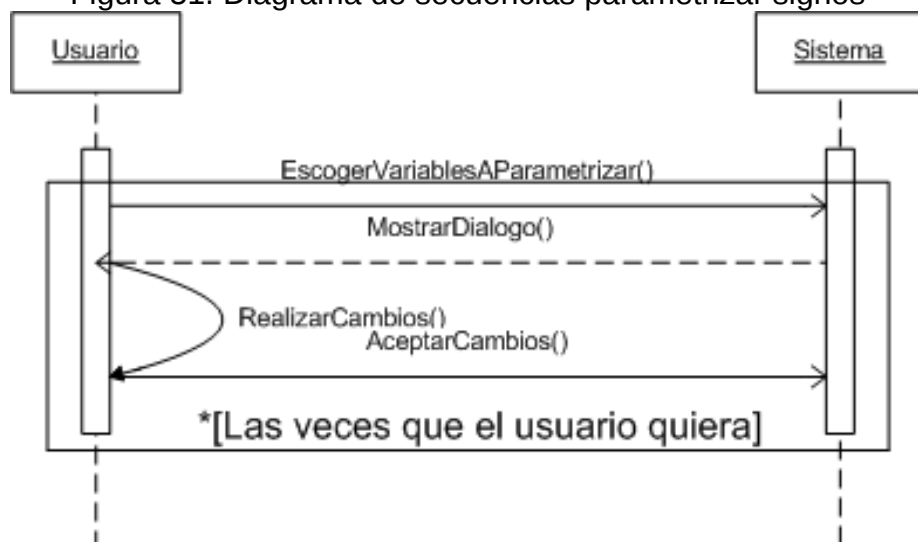
provenzan los hechos el sistema (HCE, BD, usuario) mostrara los datos en un lugar diferente de la pantalla.

Figura 50. Diagrama de secuencias mostrar alarmas y sugerencias



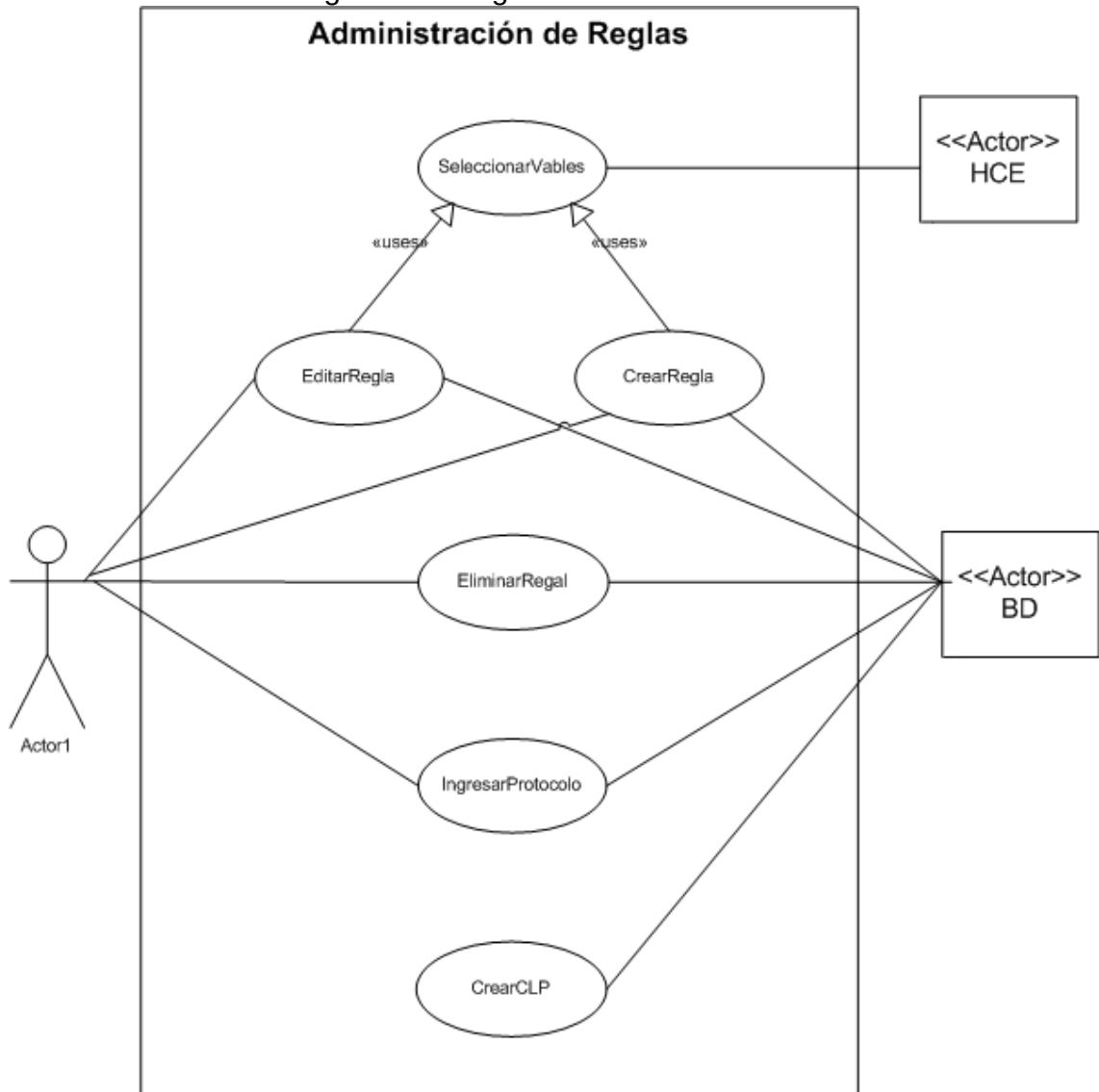
Este último diagrama muestra la secuencia de interacción de usuario con el sistema en el momento en el que el usuario desea cambiar los valores por defecto de las presiones y de la temperatura.

Figura 51. Diagrama de secuencias parametrizar signos



ANEXO C. DIAGRAMAS UML DE ADMINISTRACIÓN DE REGLAS

Figura 52. Diagrama de casos de uso



A continuación se presenta la descripción de cada uno de los casos de uso de acuerdo al formato existente para esto.

CASO DE USO – SELECCIONAR VARIABLES.

Nombre caso de uso: Seleccionar Variables.

Actor Principal: Usuario autorizado para la creación de reglas.

Personal Involucrado e Intereses:

- Medico : Quiere que el sistema llegue a un punto donde lo ayude a prestar una mejor atención al paciente
- Clínica: Quiere obtener un sistema más eficiente y que funcione para los diferentes protocolos de manejo que hay en la UCIP.
- Usuario: Quiere ingresar las variables necesarias para la elaboración de la regla.

Precondiciones: El sistema debe haber cargado las diferentes variables que se encuentran en la base de datos de la Historia Clínica Electrónica (HCE).

Garantía de éxito (Postcondiciones): Se seleccionan las variables necesarias y estas se usan para la elaboración de la regla.

Escenario Principal de Éxito:

1. El sistema muestra al usuario las diferentes variables que se encuentran en la base de datos de la HCE.
2. El usuario busca la variable que necesita para la regla.
3. Selecciona la variable que va a usar para hacer la regla.
4. El sistema carga en una variable temporal el nombre de esta variable seleccionada de la base de datos de la HCE, para ser guardada cuando el usuario acepte guardar la regla.

Extensiones (o flujos alternativos):

- 3ª. El deselecta la variable que ya no desea usar en la regla.
- 4ª. El sistema borra de la variable temporal el nombre de la variable que ha sido deseccionada.

CASO DE USO – CREAR REGLAS.

Nombre caso de uso: Crear Reglas.

Actor Principal: Usuario autorizado para la creación de reglas.

Personal Involucrado e Intereses:

- Medico : Quiere que el sistema llegue a un punto donde lo ayude a prestar una mejor atención al paciente
- Clínica: Quiere obtener un sistema más eficiente y que funcione para los diferentes protocolos de manejo que hay en la UCIP.
- Usuario: Quiere hacer una regla para la base de conocimiento del sistema experto.

Precondiciones: El usuario debe haber iniciado el sistema y entrado a la sección administración reglas.

Garantía de éxito (Postcondiciones): Se llenan los datos necesarios para la creación de reglas y son guardados en la base de datos.

Escenario Principal de Éxito:

1. El sistema muestra el formulario donde el usuario podrá seleccionar el modulo al cual pertenecerá la regla.
2. El usuario selecciona el modulo al cual pertenecerá la regla.

3. El usuario indicara al sistema que desea crear la regla.
4. El sistema carga las reglas existentes que pertenecen al modulo.
5. El sistema carga las variables que han sido establecidas para el sistema en la sección de administración de variables.
6. El sistema muestra el dialogo para la creación de reglas, con las variables y reglas cargadas.
7. El usuario selecciona las variables necesarias para la creación de la regla.
8. El usuario selecciona las reglas necesarias para hacer encadenamiento de reglas.
9. El usuario le da un nombre a la regla.
10. Se asigna una prioridad a la regla por parte del usuario.
11. Selecciona el operador con el cual se va a comparar la variable.
12. Elige el parámetro de comparación.
13. El usuario selecciona el tipo de lado derecho.
14. Se ingresa el mensaje para ser mostrado como alarma o sugerencia.
15. El usuario revisa la regla como ha quedado en un lenguaje natural y entendible para él.
16. El usuario acepta la regla que ha creado.
17. El sistema guarda la regla en la base de datos.

Extensiones (o flujos alternativos):

- 7a. Selecciona las variables: Incluye Seleccionar Variables.
- 12a. Cuando la variable esta dentro de un rango, el usuario seleccionara entre alguno de los limites.
- 12b. Cuando la variable tiene un valor cualitativo, el usuario seleccionara entre alguno de los valores posibles.
- 14a. El usuario elige el medicamento a sugerir en el tratamiento.
- 14b. El usuario selecciona el protocolo al cual se remite en caso de activarse esa regla.

CASO DE USO – EDITAR REGLAS.

Nombre caso de uso: Editar Reglas.

Actor Principal: Usuario autorizado para la creación de reglas.

Personal Involucrado e Intereses:

- Medico : Quiere que el sistema llegue a un punto donde lo ayude a prestar una mejor atención al paciente
- Clínica: Quiere obtener un sistema más eficiente y que funcione para los deferentes protocolos de manejo que hay en la UCIP.
- Usuario: Quiere modificar una regla de las existentes para la base de conocimiento del sistema experto.

Precondiciones: El usuario debe haber iniciado el sistema, entrado a la parte de creación de reglas e indicado que desea modificar una regla existente.

Garantía de éxito (Postcondiciones): La regla se modifica y se actualiza en la base de datos.

Escenario Principal de Éxito:

1. El sistema muestra el formulario donde el usuario podrá elegir que regla desea modificar, dentro de las existentes en cada modulo.
2. El sistema carga los datos correspondientes
3. El usuario modifica las variables necesarias para la creación de la regla.
4. Modifica las reglas para hacer encadenamiento de reglas.
5. El usuario le modifica el nombre a la regla.
6. Es modificada la prioridad a la regla por parte del usuario.
7. Modifica el operador con el cual se va a comparar la variable.
8. Modifica el parámetro de comparación.
9. El usuario modifica el tipo de lado derecho.
10. Se ingresa el mensaje para ser mostrado como alarma o sugerencia.
11. El usuario revisa la regla como ha quedado en un lenguaje natural y entendible para él.
12. El usuario acepta la regla que ha creado.
13. El sistema actualiza la regla en la base de datos.

Extensiones (o flujos alternativos):

- 6a. Modifica las variables: Incluye Seleccionar Variables.
- 8a. Cuando la variable esta dentro de un rango, el usuario seleccionara entre alguno de los limites.
- 8b. Cuando la variable tiene un valor cualitativo, el usuario seleccionara entre alguno de los valores posibles.
- 9a El usuario elige el medicamento a sugerir en el tratamiento.
- 9b El usuario selecciona el protocolo al cual se remite en caso de activarse esa regla.

CASO DE USO – ELIMINAR REGLAS

Nombre caso de uso: Eliminar Reglas.

Actor Principal: Usuario autorizado para la creación de reglas.

Personal Involucrado e Intereses:

- Medico : Quiere que el sistema llegue a un punto donde lo ayude a prestar una mejor atención al paciente
- Clínica: Quiere obtener un sistema más eficiente y que funcione para los deferentes protocolos de manejo que hay en la UCIP.
- Usuario: Quiere eliminar alguna de las reglas existentes.

Precondiciones: El usuario le indica al sistema que desea eliminar una de las reglas existentes.

Garantía de éxito (Postcondiciones): La regla es eliminada de la base de datos.

Escenario Principal de Éxito:

1. El sistema muestra el formulario donde el usuario podrá eliminar la regla.
2. El sistema muestra las reglas existentes.
3. El usuario selecciona la regla a eliminar.
4. El sistema muestra al usuario los datos correspondientes a la regla que selecciono para eliminar.

5. El usuario verifica que sea a regla que quiere eliminar.
6. El usuario acepta la regla que quiere eliminar.
7. El sistema elimina la regla de la base de datos.

CASO DE USO – INGRESAR PROTOCOLO.

Nombre caso de uso: Ingresar Protocolo.

Actor Principal: Usuario autorizado para la creación de reglas.

Personal Involucrado e Intereses:

- Medico : Quiere que el sistema llegue a un punto donde lo ayude a prestar una mejor atención al paciente
- Clínica: Quiere obtener un sistema más eficiente y que funcione para los deferentes protocolos de manejo que hay en la UCIP.
- Usuario: Quiere ingresar un nuevo protocolo para agregarle a este las diferentes reglas correspondientes.

Precondiciones: El usuario debe haber iniciado el sistema.

Garantía de éxito (Postcondiciones): Se crea un nuevo modulo correspondiente al protocolo que se desea crear.

Escenario Principal de Éxito:

1. El sistema trae de la base de datos los diferentes módulos existentes.
2. El sistema muestra al usuario los módulos encontrados.
3. El usuario indica que desea agregar un modulo nuevo.
4. El sistema pide al usuario el nombre del nuevo protocolo.
5. El usuario ingresa el nombre del nuevo protocolo.
6. El sistema valida el nuevo protocolo a crear.
7. El protocolo es aceptado.
8. Se guarda el protocolo en la base de datos.

Extensiones (o flujos alternativos):

- 7a. El protocolo no es aceptado.
 1. El sistema pide nuevamente el nombre de nuevo protocolo.
 2. El usuario ingresa el nombre.
 3. El sistema valida el nuevo protocolo.

CASO DE USO – CREAR CLP.

Nombre caso de uso: Crear CLP.

Actor Principal: Sistema Experto.

Personal Involucrado e Intereses:

- Medico : Quiere que el sistema llegue a un punto donde lo ayude a prestar una mejor atención al paciente
- Clínica: Quiere obtener un sistema más eficiente y que funcione para los deferentes protocolos de manejo que hay en la UCIP.

Precondiciones: El usuario debe haber ingresado al menos una regla a la base de datos.

Garantía de éxito (Postcondiciones): Se crea un archivo con extensión .clp listo para ser cargado por el motor de inferencia.

Escenario Principal de Éxito:

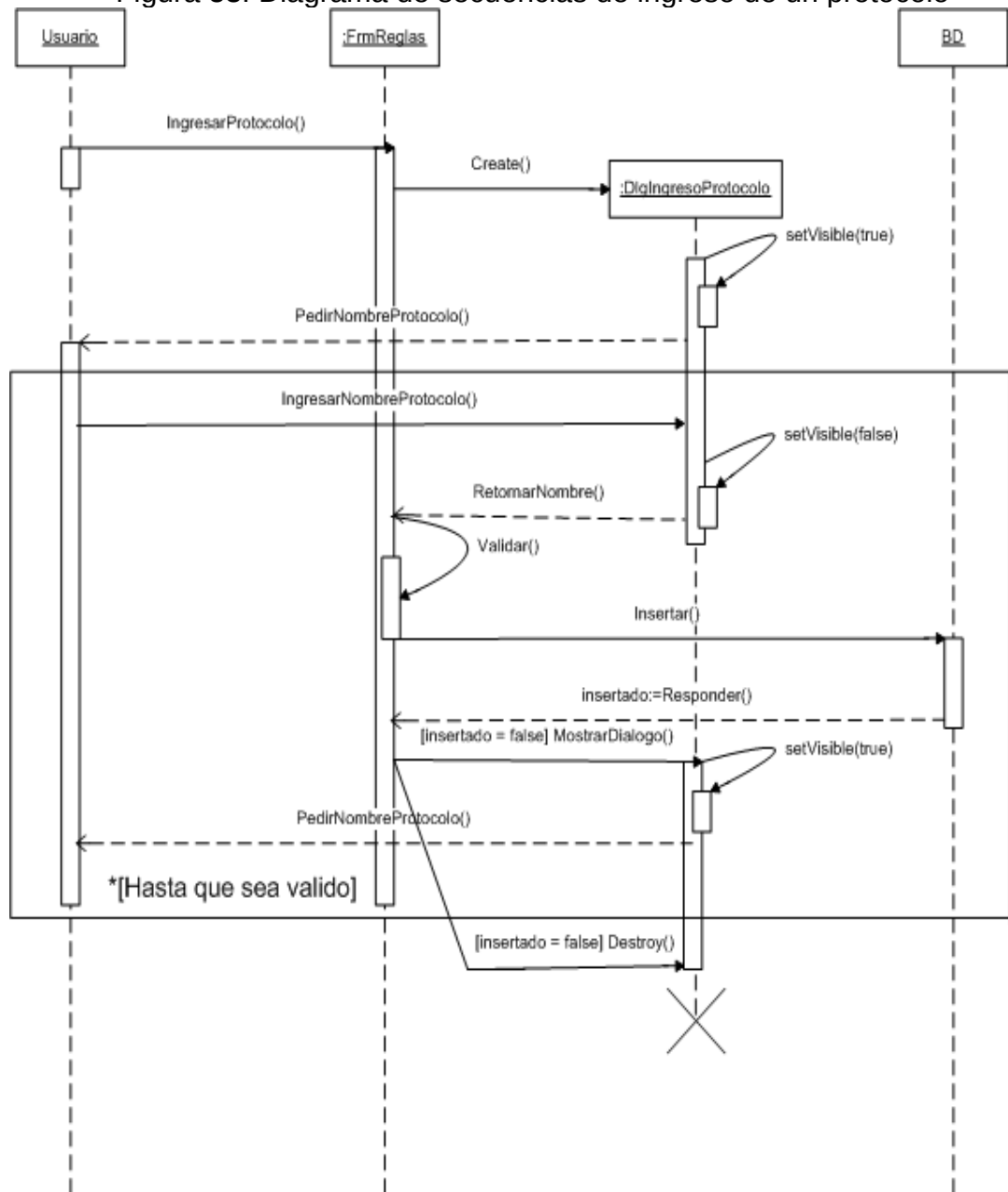
1. El sistema carga los datos de las reglas de la base de datos en variables.
2. El sistema construye las reglas con los diferentes datos obtenidos según la sintaxis del motor de inferencia.
3. El sistema crea un archivo con extensión .clp.
4. El sistema imprime en el archivo las diferentes reglas construidas.

Extensiones (o flujos alternativos):

- 1a. No existen reglas en la base de datos.
 1. El sistema termina el proceso ya que no hay reglas para crear el archivo.

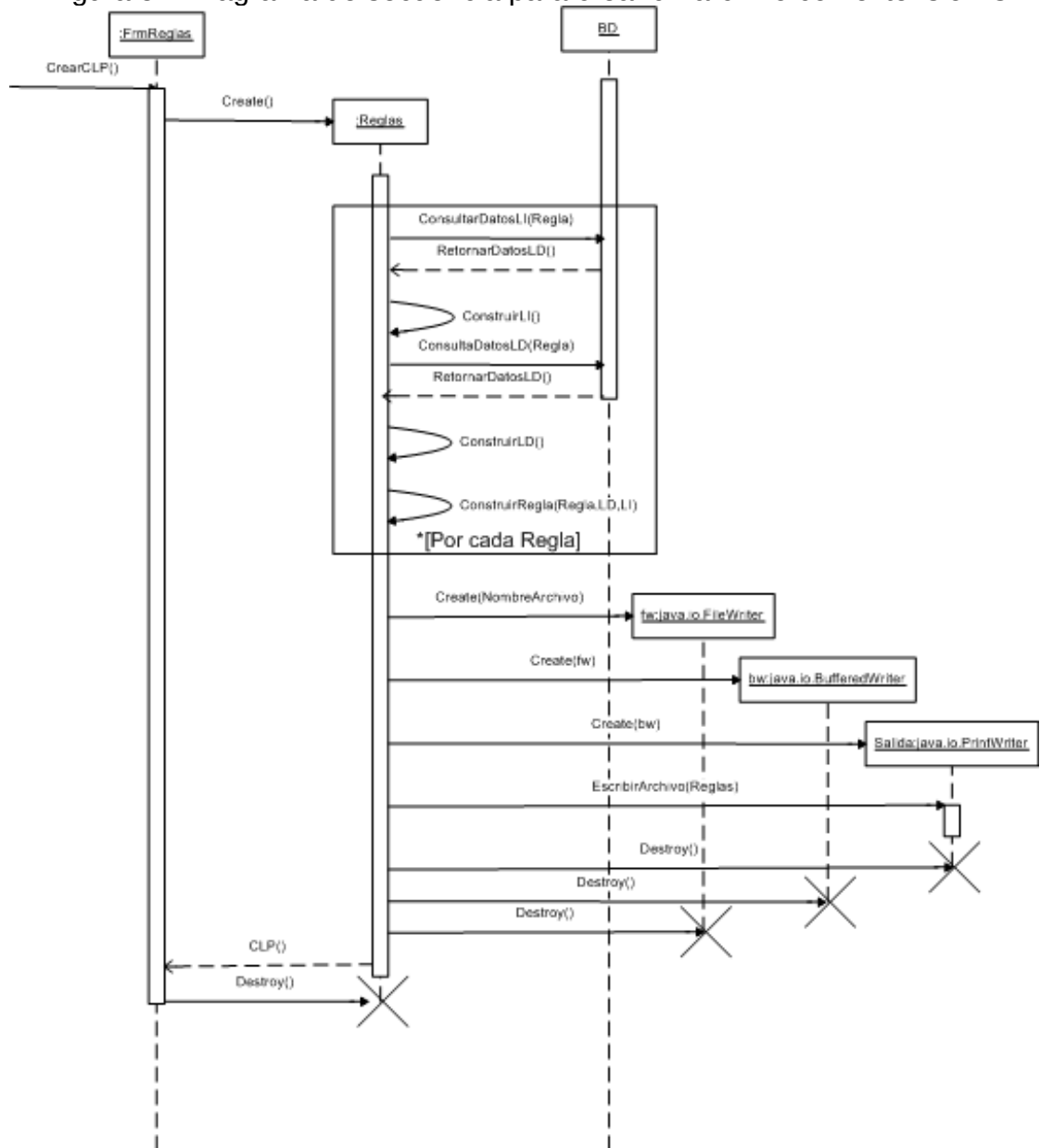
Los siguientes son los diagramas de secuencias correspondientes a ingresar un protocolo, ingresar una regla, editar una regla, crear el archivo con extensión .clp y eliminar una regla.

Figura 53. Diagrama de secuencias de ingreso de un protocolo



Este diagrama de secuencias muestra el proceso que se realiza para el ingreso de un nuevo protocolo. El usuario indica al sistema enviándole un mensaje al formulario de que desea crear un protocolo, el sistema crea una instancia de un dialogo por medio del cual el usuario introducirá el nombre del nuevo protocolo, el sistema pedirá el nombre hasta que este sea un protocolo valido y que se pueda agregar en la base de datos. Cuando el protocolo es insertado la instancia del dialogo es destruida.

Figura 54. Diagrama de secuencia para crear un archivo con extensión CLP



En este diagrama podemos apreciar como el sistema crea el archivo con las reglas, primero crea cada una de las reglas con la sintaxis del motor de inferencia con los datos guardados en la base de datos. Luego crea las instancias de las clases nativas en java que son necesarias para la creación de un archivo, después envía al archivo las reglas creadas. Cuando el archivo es creado se destruyen las instancias de las clases que lo crearon.

Figura 55. Diagrama de secuencias de creación de reglas

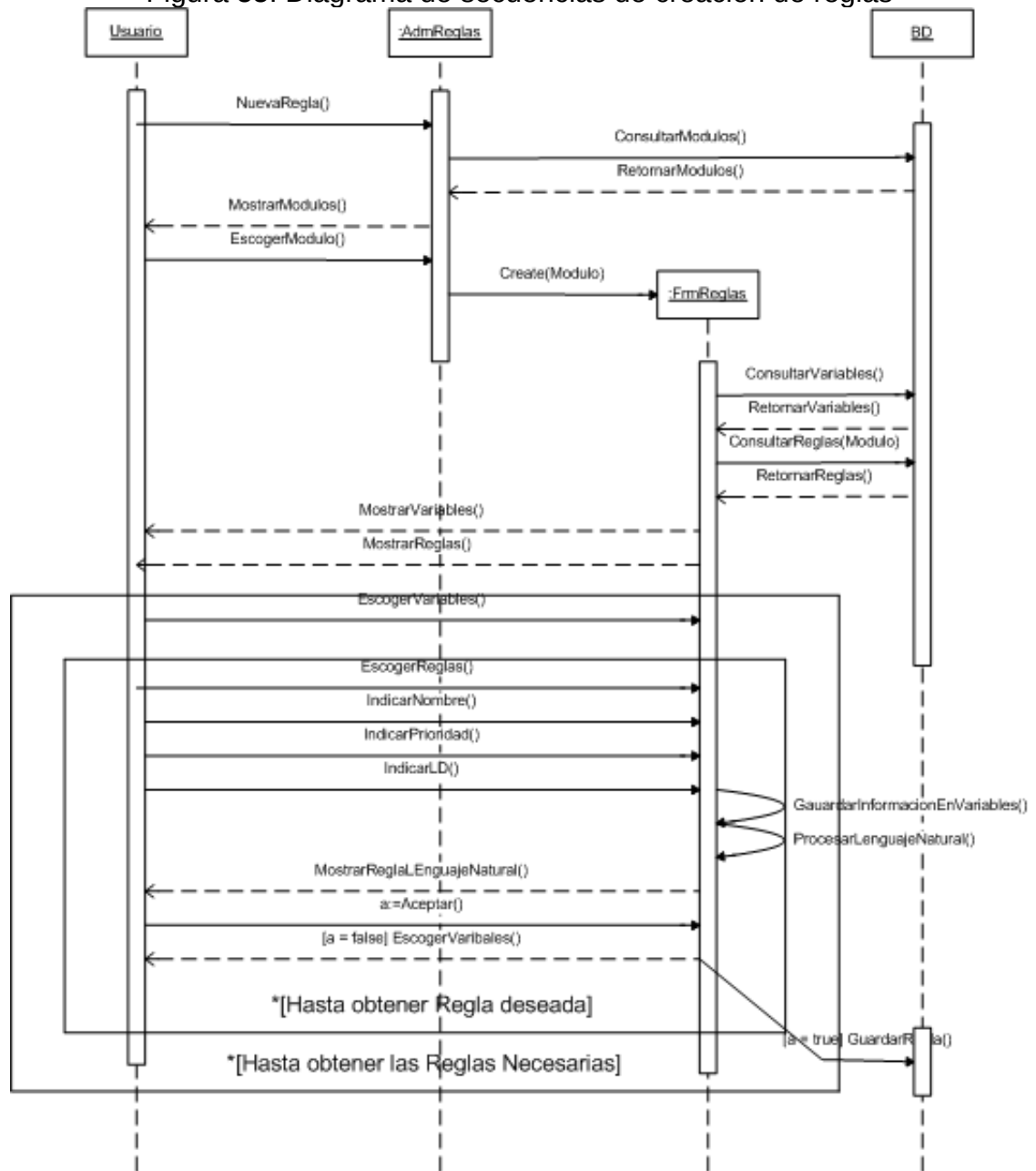
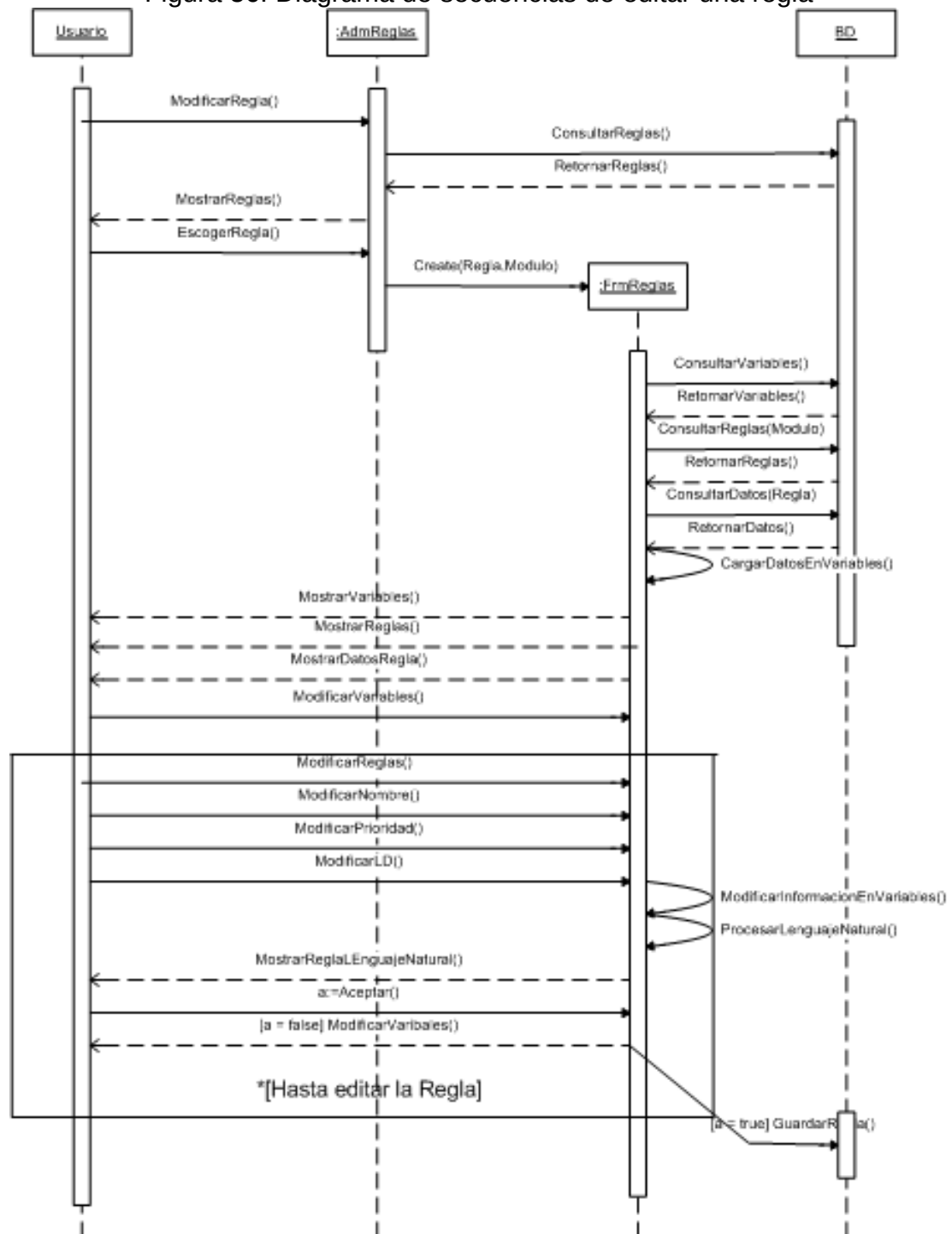


Diagrama de secuencias de creación de reglas (figura 55): Aquí se puede apreciar la secuencia que se realiza cuando un usuario desea crear una regla, primero el sistema pone a disposición del usuario los módulos existentes, después que selecciona el módulo al cual pertenecerá la regla, se mostrara un nuevo dialogo con las reglas de la base de datos, después el usuario elige y digita los datos necesarios para la creación de la regla hasta obtener la regla que el desea construir. Cuando la regla es aceptada, esta se guarda en la base de datos, todo esto se puede repetir hasta que el usuario haya creado todas las reglas que el necesite.

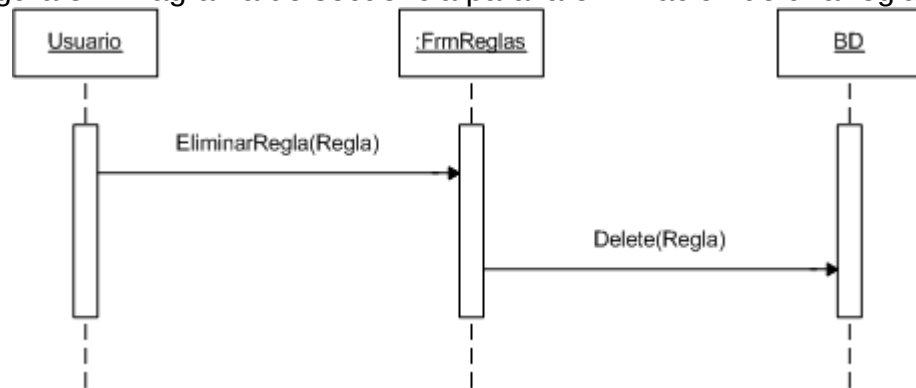
Figura 56. Diagrama de secuencias de editar una regla



La siguiente es la secuencia (figura 56) que ocurre cuando el usuario desea editar una regla existente, el sistema pondrá a disposición los módulos y las reglas que se encuentran en cada uno de ellos, de allí el usuario escogerá cual regla es la que desea editar, el sistema cargará las variables y las reglas tal como lo hace cuando se crea una regla, pero además carga los datos de la regla que se desea modificar para que el usuario pueda modificarlos

(agregar o quitar variables y reglas, cambiar nombre, etc.). Después de que la regla es modificada en el formulario, esta es modificada en la base de datos.

Figura 57. Diagrama de secuencia para la eliminación de una regla

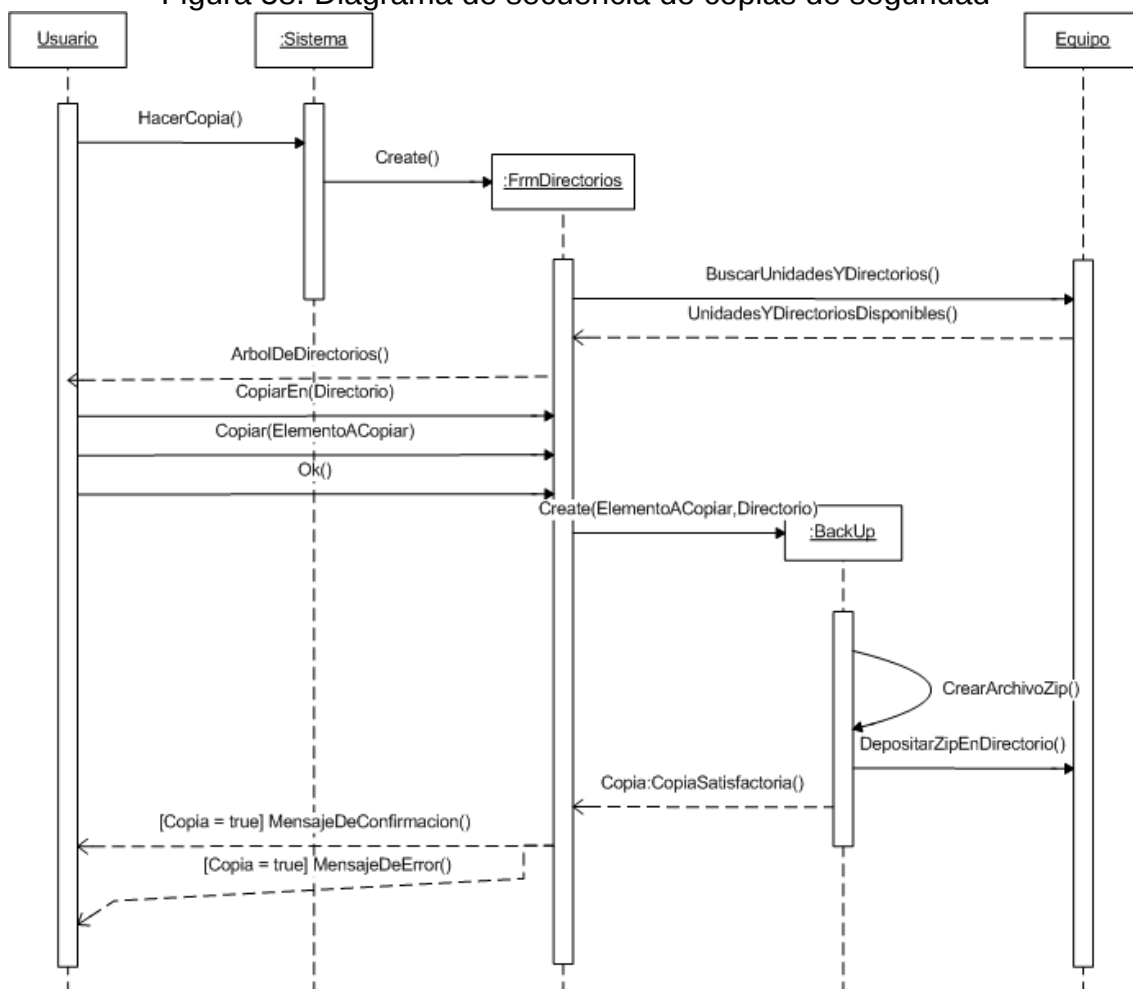


En la figura 57 se ve la secuencia al momento de eliminar una regla, el usuario indica cual regla es la que desea eliminar y el sistema la elimina de la de la base de datos.

ANEXO D. DIAGRAMA DE SECUENCIA PARA LAS COPIAS DE SEGURIDAD

Para la creación del modulo de copias de seguridad se realizo el siguiente diagrama de secuencias:

Figura 58. Diagrama de secuencia de copias de seguridad



En el anterior diagrama se aprecia los pasos a seguir por el usuario, y lo que debe hacer el sistema para realizar las copias de seguridad. El usuario debe indicar al sistema que desea hacer una copia de seguridad, el sistema debe mostrar los diferentes directorios y unidades donde el usuario puede guardar su copia; luego de que el usuario elija que desea copiar (si la base de datos o los informes) y acepte, el sistema debe indicarle al usuario si la copia fue exitosa o no.

ANEXO E. GLOSARIO

Aorta: Arteria que lleva la sangre arterial (oxigenada) desde el ventrículo izquierdo al resto del cuerpo.

Arritmia: Frecuencia ritmo anormal de la contracción auricular o ventricular.

Arteria pulmonar: Lleva la sangre venosa (desoxigenada), desde el ventrículo derecho a los pulmones para que recupere su nivel de oxígeno óptimo.

Balance hídrico: Relación que hay entre los líquidos se le suministran al paciente y los que son evacuados por el paciente. Su valor ideal es de cero (que lo que entre sea igual a lo que sale).

Bomba de Infusión: Máquina que controla el ritmo y cantidad de fluido intravenoso administrado.

Cardiovascular: Relacionado con el corazón y los vasos sanguíneos (arterias, venas y capilares); sistema circulatorio.

Catéter: tubo plástico que se inserta en el cuerpo; incluye aquellos catéteres que se usan para administrar alimentos, succionar líquidos, administrar fluidos intravenosos y extraer muestras de orina.

Derrame pleural: Acumulación patológica de líquido en el espacio pleural; es el resultado de un desequilibrio entre la formación y la reabsorción de líquido a este nivel.

Destete ventilatorio: Es la adaptación progresiva de un paciente intubado y ventilado mecánicamente a la ventilación espontánea.

Diálisis peritoneal: El proceso de diálisis se basa en la ósmosis (filtración y difusión de soluciones a través de las membranas semipermeables). La solución de diálisis se perfunde en la cavidad peritoneal a través de un catéter. La membrana peritoneal sirve como membrana semipermeable entre los vasos sanguíneos y el líquido de diálisis. Cuando se extrae el líquido de diálisis, éste contiene los productos de desecho que normalmente serían expulsados por los riñones. Además ayuda en la redistribución de líquidos corporales y optimiza las funciones orgánicas (especialmente la cardio-pulmonar).

Diástole: Fase del ciclo cardíaco en la cual el músculo cardíaco se relaja y dilata, y el corazón se llena de sangre.

ECG/EKG: Un electrocardiograma (ECG o EKG), es un gráfico en el que se inscriben las variaciones del voltaje del corazón, captadas por electrodos en la superficie de nuestro cuerpo, en relación con el tiempo.

Ecocardiograma: Es una prueba que utiliza ondas de ultrasonido para observar el corazón. Sirve para determinar el tamaño del corazón, fuerza del bombeo, problemas valvulares, fluidos alrededor del corazón,, coágulos de sangre, tumores dentro del corazón u orificios anormales entre las cámaras.

Endovenoso: método que se utiliza para administrar medicamentos o líquidos directamente en la vena.

Electrolitos: sodio, potasio y cloruro, elementos esenciales para el cuerpo que pueden medirse en la sangre u orina, o agregarse al fluido intravenoso.

Epicrisis: Es el resumen de la historia clínica del paciente que ha recibido servicios de urgencia con observación o de hospitalización.

Extrasístoles: Contracción prematura de la aurícula o el ventrículo, (o de ambos) independiente del ritmo normal.

Frecuencia cardiaca: Cantidad de latidos del corazón en un determinado tiempo.

Gasto cardiaco: Cantidad de sangre que el corazón expulsa por minuto. En reposo es alrededor de cinco litros y en situaciones de esfuerzo puede llegar a multiplicarse por cuatro.

Hematocrito: Porcentaje del volumen de la sangre que ocupa la fracción de los glóbulos rojos. Las cifras normales de hematocrito en humanos oscilan entre 37% y 54% dependiendo de diversos factores fisiológicos, como la edad, el sexo y la condición física del sujeto.

Hemoglobina: Es la sustancia química de la sangre contenida en los glóbulos rojos y encargada del transporte del oxígeno desde los pulmones a todos los tejidos del organismo.

Hipotermia: Estado en el que la temperatura del cuerpo es inferior a 35°C, causada generalmente por una exposición prolongada al frío. Se caracteriza por una disminución de las frecuencias respiratoria y cardiaca, palidez generalizada y escalofríos.

Historia clínica: Es un documento en el que se registran los datos e información pertinentes a la salud y a la enfermedad de una persona, se archivan en el área de Estadística y puede ser de papel, en película, soporte óptico o magnético. Debe ser única o sea que una persona tendrá a lo largo de su vida un solo documento.

Homeostasis: Cuando el organismo funciona como debe porque todos los sistemas están equilibrados.

Inotropico: Agente que afecta la fuerza de las contracciones musculares.

Intervalo P-R: Comprende la onda P más el segmento P-R del EKG, representa el tiempo que media entre el comienzo de la despolarización auricular y el comienzo de la ventricular.

Intubación: colocación de un tubo en la tráquea, que se conecta a un ventilador para ayudar al bebé a respirar.

Isquemia miocárdica: Falta de oxigenación del tejido miocárdico por disminución del aporte sanguíneo al mismo. Puede variar desde una isquemia reversible, con sintomatología muy variable, desde la falta de síntomas reconocibles por el paciente, pasando por la angina de pecho, hasta un infarto de miocardio, caracterizado por necrosis del tejido miocárdico.

Leucocitos: También llamados glóbulos blancos, hay entre 9.000 y 10.000 unidades por mm^3 de sangre. Actúan defendiendo al organismo de los agentes nocivos.

Monitoreo hemodinámico: Técnica de medición de signos fisiológicos útil para la identificación precoz de situaciones que amenazan la supervivencia, para la evaluación de la respuesta inmediata del paciente al tratamiento y/o intervenciones realizadas, así como para el aporte de datos que orienten diagnósticos médicos y de enfermería.

Monitoreo invasivo: monitoreo que se le realiza al paciente por medio de instrumentos que son introducidos al cuerpo del paciente.

Monitoreo no invasivo: monitoreo realizado al paciente con instrumentos externos.

Neonato: Es el tiempo de vida desde el momento del nacimiento hasta los 40 días de vida.

Neumotórax: escape de aire de los pulmones hacia el espacio pleural (entre los pulmones y las costillas).

Pleura: La pleura es una membrana serosa de origen mesodérmico que recubre ambos pulmones, el mediastino, el diafragma y la parte interna de la caja torácica. La pleura parietal es la parte externa, en contacto con la caja torácica mientras que la pleura visceral es la parte interna, en contacto con los pulmones.

PVC: La presión venosa central (PVC) se corresponde con la presión sanguínea a nivel de la aurícula derecha y la vena cava, estando determinada

por el volumen de sangre, volemia, estado de la bomba muscular cardiaca y el tono muscular. Es una presión invasiva.

Radiografía: imagen bidimensional de una estructura del cuerpo, con diferentes tonos del negro al blanco separando estructuras y delimitando tejidos. Generalmente se tiene un modelo normal y con este se pueden comparar las variaciones que aparezcan para extraer datos necesarios para el diagnóstico de diferentes patologías

Sístole: Es la contracción del tejido muscular cardiaco. Esta contracción produce un aumento de la presión en la cavidad cardiaca, con la consiguiente eyección del volumen sanguíneo contenido en esa cavidad.

Switch Arterial: En este procedimiento la aorta es conectada al ventrículo izquierdo de tal forma que ésta es bombeada al resto del cuerpo. Por otro lado la arteria pulmonar es reconectada al ventrículo derecho de tal forma que la sangre venosa se dirija a los pulmones. Este switch arterial debe realizarse en la primera semana después del nacimiento o dependiendo de varios factores, poco tiempo después.

TGA: En la transposición de grandes arterias (TGA), la arteria pulmonar y la aorta están invertidas, de tal forma que la aorta está conectada al ventrículo derecho llevando la sangre venosa a todo el cuerpo y por otro lado la arteria pulmonar se conecta con el ventrículo izquierdo permitiendo que la sangre oxigenada vuelva a entrar a los pulmones.

UCIP: La unidad de cuidados intensivos pediátrica (UCIP) es un servicio destinado a la hospitalización de pacientes menores de 18 años, en postoperatorio de cirugía cardiovascular o con alteraciones multisistémicas causadas por otras patologías, que se encuentren en estado crítico que exigen monitoreo invasivo y no invasivo, asistencia médica y de enfermería permanente con la ayuda de equipos especializados.

Vasodilatador: Agente que causa la dilatación de los vasos sanguíneos.

Vasoconstrictor: Agente causante de la constricción o disminución del volumen de los vasos sanguíneos.

Ventilación mecánica: Medio terapéutico al que se recurre cuando la mecánica ventilatoria del paciente no le permite el intercambio gaseoso suficiente para cubrir las necesidades mínimas de oxígeno y eliminación de dióxido de carbono, la cual necesita ser ajustada a la homeostasis del paciente, entendiendo como tal los parámetros ventilatorios y ácido- base que le eran usuales a esa persona.

ANEXO F. ESTÁNDAR DE DESARROLLO

1. PROPÓSITO

Establecer la normatividad estándar que se debe tener en cuenta al momento de implementar un aplicativo.

2. ALCANCE

Este estándar es aplicable al área de desarrollo tecnológico del grupo de Investigación en pediatría, por todo el personal que tenga dentro de sus actividades el desarrollo de software.

3. DEFINICIONES

No aplica

4. DESCRIPCIÓN DE ACTIVIDADES

Símbolos y Convenciones:

Sis: Sigla de 3 letras o del proyecto software. Ejemplo: Software para IPS: FCV

Sistema: Nombre descriptivo abreviado del aplicativo, de máximo 15 letras sin espacios en blanco. Ejemplo: FCV.Soft

Mod: Sigla de 3 letras del nombre del módulo. Ejemplo: Hce (Historia clínica electrónica)

Módulo: Nombre descriptivo abreviado de los módulos componentes de un sistema, si los hay, de máximo 15 letras. Ejemplo: ViaClinica, Módulo de Vías clínicas componente de Gerencial.

Nombre Abreviado: Nombre descriptivo abreviado de la referencia, de máximo 15 letras sin espacios en blanco.

Clase: Abreviado de los tipos de imágenes utilizados en el proyecto software.

Nombre Documento: Nombre de los documentos soporte del proyecto software.

Identificación: Número de identificación de las personas. Aplica para imágenes y video.

Consecutivo: Número de secuencia, para el caso de tener varios archivos del mismo tipo o clase. Aplica para imágenes y video.

4.1. Árbol de Directorios y Nomenclatura de Archivos

La organización de los directorios de un proyecto software, dependerá de una carpeta con el nombre del mismo. Se utilizara el nombre de proyecto *ExampleV1.0* como ejemplo ilustrativo. En dicha carpeta principal los subdirectorios son:

	ExampleV1.0: Folder principal del proyecto dentro de c:\InvesPed\	
	Bin: (Binary File) Archivos Ejecutables(.exe) y de procesamiento (.bat) que hayan sido probados	
	Bint: (Binary File) Archivos Ejecutables(.exe) y de procesamiento (.bat) que no hayan sido probados	
	Dat: (Data) Archivos de Base de datos Local	
	Dll: (Dynamic Link Library) Librerías dinámicas compiladas	
	Doc: (Document) Documentacion General del producto	
	Dev: (Developer) Controles de cambio del software	
	Pmg: (Project Managment) Archivos de administracion del Proyecto	
	Usr: (User Document) Documentacion de Usuario	
	Hlp: (Help) Manual de Usuario en línea	
	Img: (Image) Imágenes comunes a la documentacion de usuario	
	Tch: (Technical) Documentacion Técnica del Software	
	Trn: (Training) Manuales y videos de entrenamiento	
	Flt: (Flat File) Archivos Planos generados por el Software	
	Img: (Image) Imágenes a cargar en tiempo de ejecucion	
	Ins: (Installer) Instaladores del producto	
	Ole: (Object Linking Embedding) Objetos enlazados desde el software	
	Source: Archivos fuentes del software	
	Lybrary: Cursores, imágenes, videos e iconos usados en tiempo de diseño	
	Ico: (Icon) Archivos de íconos	
	Img: (Image) Archivos de Imágenes	
	Snd: (Sound) Archivos de Sonido	
	Vid: (Video) Archivos de Video	

Figura 59 Árbol de Directorios y Nomenclatura.

4.2. Programación de Procesos

4.2.1 Nomenclatura de Objetos en Programación

4.2.1.1 Clases

Prefijo: C $\overline{F}$ rm para las clases que extienden de JFrame, C $\overline{D}$ lg para las clases que extienden de Dialog, C $\overline{T}$ hr para las clases que extienden de Thread. Para aquellas clases que no heredan de ninguna clase, se usara el prefijo C.

Nombre: *NombreAbreviado*

Ejemplos: CdlgPresiones, Cprincipal.

4.2.1.2 Objetos en Formularios y Diálogos

Prefijo xxx: Éste se define para cada objeto en las Tabla de Componentes, Tabla de Contenedores y Tabla de Menús.

Nombre: NombreAbreviado que identifique la variable que representan.

Ejemplos: cmdSalir, lblEmpresa.

Tabla 8. Componentes swing

COMPONENTES SWING	
Objeto	Abreviatura
Jbutton	jbtn
JcheckBox	jchk
JradioButton	jrads
Jlabel	jlbl
JtextField	jtxf
JpasswordField	jpas
JtextArea	jtxa
JscrollBar	jscr
Jlist	jlst
JcomboBox	jcmb

Tabla 9. Contenedores swing

CONTENEDORES SWING	
Contenedor	Abreviatura
JFrame	jfrm
JPanel	jpnl
JDialog	jdlg

Tabla 10. Swing menus

SWING MENUS	
Menu	Abreviatura
JMenuBar	jmnb
JMenu	jmnu
JPopupMenu	jpum

4.2.1.3 Variables

Prefijo xyyy

x : global por clase, o locales de un método.

yyy : identifican el tipo de dato según Tabla de Tipos de Datos.

Nombre: Nombreabreviado

Ejemplos: gstrUsuario, lstrCadena.

Tabla 11. Tipos de Datos

VARIABLES	
Tipo de Dato	Abreviatura
Connection	con
Date	dte
double	dbl
float	flt
int	int
Integer	itg
long	lng
String	str
Statement	sta
ResultSet	rst

4.2.1.4 Procedimientos y Funciones

Prefijo xy

x : publica, privada, protegida...

Nombre: NombreAbreviado

Ejemplos: pubguardar , preimprimir

4.2.1.5 Reportes

Prefijo *rptXXX*Y

Donde XXX, hace referencia a la abreviatura de la patología y Y es un consecutivo si es requerido.

Nombre: NombreAbreviado.

Ejemplos: rptTga32Historico.

4.3. Base de Datos

4.3.1 Nomenclatura del Nombre de la Base de Datos

Los nombres de los archivos de la base datos empezaran por el prefijo BD.

Nombre NombreAbreviado

Excepcion Ninguna

Ejemplos BDVariables, BDReglas

4.3.2 Nomenclatura de Objetos en la Base de Datos

4.3.2.1 Campos

Prefijo de tres (3) caracteres en minúscula que indiquen su contenido.

Nombre *NombreAbreviado*

Excepción Ninguna

Ejemplos numFactura, codProducto, codCliente, fecNacimiento

4.3.2.2 Tablas

Prefijo con los caracteres tbl

Nombre *NobreAbreviado*

Excepción Si la tabla es creada directamente por la aplicación se separarán el prefijo, nombre y demas identificadores por el carácter _

Ejemplos tblVAvariables, tbl_OndaQ_Profundidad

ANEXO G. CAPACITACIÓN SOBRE PUNTOS DE FUNCIÓN

Lo más destacado de esta capacitación es lo descrito a continuación:

Las métricas orientadas a la función fueron propuestas por Allan **Albrecht**, de IBM, en 1979 quien sugirió un acercamiento a la medida de la productividad denominado método de puntos de función. Los puntos de función miden la aplicación desde una perspectiva del usuario, dejando de lado los detalles de codificación.

Los puntos de función evalúan con fiabilidad el valor comercial de un sistema para el usuario, tamaño del proyecto, coste y tiempo de desarrollo, calidad y productividad del programador y el esfuerzo de adaptación, modificación y mantenimiento.

Un programa que tenga una determinada cantidad de puntos de función, entrega al usuario final el mismo número de funciones comerciales para el usuario final.

El proceso de los puntos de función es básicamente el siguiente: Se identifican las funciones que se van a entregar al usuario y se ordenan dentro de una de las cinco categorías, después se clasifican según su nivel de complejidad y por ultimo se ajustan estos de función.

Las categorías son:

Entradas de usuario.

Se cuenta cada entrada única que el usuario proporcione al software diferentes datos orientados a la aplicación. Una entrada se considera única cuando tiene un formato diferente al de otra entrada, y si este es igual requiere una lógica diferente de procesamiento o modifica un archivo lógico diferente. Ejemplos de entradas son, el ratón, el teclado, transacciones en cintas, lectores de códigos de barras, etc.

Salidas al usuario.

Se cuentan las diferentes salidas únicas que proporcionan al usuario la información orientada a la aplicación. De esta manera las salidas pueden ser informes en pantalla, mensajes de error, copias de seguridad, etc.

Consultas o peticiones al usuario.

Una consulta en puntos de función se refiere a una entrada (salida) interactiva que resulta de la generación de algún tipo de respuesta en forma de salida (entrada) interactiva. Una consulta es única cuando posee un formato diferente

en su salida o en su entrada o es igual pero requiere una lógica de procesamiento diferente.

Archivos.

Se cuenta cada agrupación lógica de datos (archivo maestro lógico), que puede ser una parte de una gran base de datos o un archivo independiente.

Interfaces.

Se cuentan todas las interfaces legibles por la maquina por ejemplo los archivos de datos que pueden estar en cualquier medio (cintas, discos, etc.) y que son utilizados para transmitir información a otro sistema.

Después de haber clasificado las funciones dentro de estas categorías se procede a otorgarles un nivel de dificultad, este nivel de dificultad puede ser, simple, medio y complejo. Un criterio para determinar el nivel de dificultad consiste en analizar cuantos datos son necesarios para realizar la función y cuantos archivos se ven afectados en este proceso. Cada nivel de dificultad tiene una ponderación específica de acuerdo a la categoría de la que estemos hablando.

Figura 60. Tabla para calcular los Puntos de Función

Parámetro de medición	FACTOR DE PONDERACIÓN				
	Cuenta	Simple	Medio	Complejo	
Numero de entradas de usuario		X 3	4	6	=
Numero de salidas de usuario		X 4	5	7	=
Numero de peticiones de usuario		X 3	4	6	=
Numero de archivos		X 7	10	15	=
Numero de interfaces externas		X 5	7	10	=
Cuenta = Total					→

Una vez realizado lo anterior se llena una tabla como la que se muestra en la figura 60

El siguiente paso es ajustar estos puntos de función, esto se hace mediante la fórmula:

$PF = Total * [0.65 + 0.01 * Fi]$, donde Total es el resultado obtenido al llenar la tabla de la Figura 5.1 y Fi un valor de ajuste obtenido al responder ciertas preguntas planteadas (Están incluidos en el diseño la conversión y la instalación?, se ha diseñado el código para ser reutilizable?, etc.).

Una vez se calcula esto, este valor puede ser utilizado como medida de productividad (PF/persona-mes), calidad (Errores/PF), costo (\$/PF), etc.

ANEXO H. FORMATO DE IDEA DE PROYECTO DE CRF

Fecha de inscripción:		Cod.Interno	
		Cod.Externo	
Unidad empresarial	Área:	Tipo de Proyecto:	
IDEA DE PROYECTO			
Título: Diseño e implementación de una herramienta software para el desarrollo de bases de datos aplicables a procesos de investigación, con base en el diseño de formularios de			
Objetivos (General): Desarrollar una herramienta software que estructure y facilite la recolección de información asociada a procesos de investigación en salud.			
Descripción (Problema y Justificación): El objetivo de una prueba clínica es determinar la veracidad de una hipótesis que surge a partir de estudios no controlados observacionales, descriptivos o retrospectivos, o de estudios preclínicos, que pueden conducir a la generación de nuevo conocimiento. Dado que los datos son el principal indicador de la calidad de una prueba clínica, es necesario contar con instrumentos idóneos para su recolección. De éstos, los CRF (<i>Case Report Forms</i>) son las herramientas más importantes, diseñadas para recolectar de manera efectiva dentro de una prueba clínica, información completa y precisa, minimizando cualquier posibilidad de malinterpretarla o de cometer errores de transcripción. La recolección y manejo de la información, en el marco de cualquier investigación, requiere personal entrenado en el desarrollo de las inquietudes generadas durante la práctica clínica y que sea cualificado tanto en el acceso a las diversas fuentes de información como en el uso de las diferentes herramientas informáticas disponibles que permiten almacenar de manera organizada los datos recolectados. A partir de la experiencia adquirida dentro de los diferentes grupos de investigación de la Fundación Cardiovascular, el cuerpo médico concluyó, que las actividades en las que mas dificultades tienen los usuarios durante la implementación del método en el desarrollo de sus propuestas de investigación, son la creación de bases de datos con características adecuadas para ser analizadas con paquetes estadísticos estándar, la creación de formatos de recolección de datos y la doble digitación controlada.			
Responsable y/o investigador principal: Javier Martínez Gómez Ing. MSc.			

	Cod.Interno	
Resultados esperados: Una herramienta software diseñada para dar soporte a procesos de investigación en salud, que cumpla con las especificaciones técnicas, científicas y legales requeridas y que permita manejar bases de datos generadas a partir del diseño y publicación de formatos de recolección de datos CRF.		
Proponentes: Andrés Felipe Villamizar Vecino. Gregorio José Mauricio Rincón Blanco.		
Entidades involucradas: Fundación Cardiovascular de Colombia Universidades de la Región		
Para presentar a: Comité Técnico-Científico de la Fundación Cardiovascular de Colombia.		

ANEXO I. ARTICULO SOBRE EL SISTEMA EXPERTO

SISTEMA EXPERTO PARA SOPORTE DIAGNÓSTICO EN EL POSTOPERATORIO DE TRANSPOSICIÓN DE GRANDES ARTERIAS

Víctor Raúl Castillo¹, Xiomara Patricia Blanco Valencia², Álvaro Eduardo Durán¹, Gregorio José Mauricio Rincón Blanco³, Andrés Felipe Villamizar Vecino³,

¹MD, Fundación Cardiovascular de Colombia, inv_pediatria@fcv.org

² Ing. de Sistemas, Fundación Cardiovascular de Colombia, xblanco@fcvsoft.com

³ Estudiantes Universidad Industrial de Santander

Abstract.

En los mejores centros del mundo la mortalidad acumulada de cirugía cardiovascular pediátrica es cercana al 2%, en la Fundación Cardiovascular de Colombia, la incidencia acumulada de muerte en los últimos 8 años es del 5.8%.

El gran volumen de información, la síntesis e interpretación se suele hacer manualmente en la mayoría de las Unidades de Cuidado Intensivo, haciéndolas un lugar susceptible al error médico y sitio ideal para la aplicación de una serie de herramientas informáticas, que permitan una mayor eficiencia en el cuidado de los pacientes.

Uno de los campos de aplicación de la Inteligencia Artificial son los sistemas expertos, que solucionan problemas utilizando conocimientos basados en hechos y capacidad de razonamiento. Este tipo de tecnologías, realiza diferentes aportes a la práctica clínica: soporte confiable para la toma de decisiones mostrando sugerencias, estandarizando actitudes, métodos y tratamientos, respetando la autonomía del profesional de la medicina.

1. Planteamiento de un problema clínico modelo

La Transposición de Grandes Arterias (TGA) se refiere al origen invertido de las grandes arterias del corazón. Hoy en día es posible corregir esta anomalía en niños de 15 días de vida con un porcentaje de éxito que en los mejores centros del mundo es cercano al 100% [1]. En nuestro medio debido a las dificultades del recurso humano y en algunos casos técnicos el porcentaje de sobrevida puede ser inferior al 40%.

La mortalidad de esta patología en las unidades de cuidado intensivo (UCIs) es un indicador de calidad, debido a la alta complejidad del manejo postoperatorio, a la baja tolerancia al error, requiriéndose una estructura organizacional sofisticada, coordinación de esfuerzos de múltiples personas trabajando en equipo y altos niveles de conocimiento y desempeño técnico [1]. Se ha descrito que bajos índices de mortalidad asociados a esta patología, se asocian con mejores resultados en otros postoperatorios. Al comparar las UCIs de hospitales como los de Suecia, los cuales cuentan con los mismos recursos técnicos que las de los hospitales Colombianos, se encuentra que sus índices de mortalidad son menores. Esto último se ha atribuido al alto nivel de entrenamiento de su personal, con lo cual han logrado reducir el error humano.

2. Sistemas expertos

La disciplina conocida como Inteligencia Artificial (IA) ha generado en las últimas décadas un gran número de sistemas con supuestos rasgos de inteligencia¹⁶ en los que interactúan unos datos de entrada con los de una base de información que contiene el conocimiento de un área específica. Un tipo bien conocido de este tipo de sistemas es el de los sistemas expertos (SE).

Una de las aplicaciones más importantes de los SE ha tenido lugar en el campo médico, donde éstos han sido utilizados para recoger, organizar, almacenar, poner al día y recuperar información médica de una forma eficiente y rápida, permitiendo aprender de la experiencia [2].

Dentro de los trabajos relacionados con el tema se pueden citar: INTERNIST-1/CADUCEUS, sistema de fácil uso sobre medicina interna. MYCIN, construido en Stanford que diagnostica enfermedades infecciosas de la sangre y receta los antibióticos apropiados y PUFF que diagnostica enfermedades pulmonares. En la Universidad de Edimburgo se está realizando una tesis de doctorado para evaluar condiciones de monitoreo en neonatos en una UCI. Dicho sistema detecta falsas alarmas causadas por diferentes factores y puede inferir el comportamiento de una variable [3].

Desde hace 2 años, la Fundación Cardiovascular de Colombia (FCV) ha ido desarrollando una herramienta de ayuda diagnóstica y terapéutica orientada a la reducción del error humano y a la mejora de procesos de atención en salud gracias a la toma oportuna de conductas para preservar la estabilidad clínica y predecir la complicación de niños atendidos en la UCI postoperatoria cardiovascular pediátrica. Dicha herramienta, podría en un futuro ser extendida

¹⁶ En el área de la IA, el término inteligencia se utiliza de manera débil, haciendo referencia a cualquier comportamiento de apariencia inteligente, que puede ir del reconocimiento de un patrón perceptual determinado a la toma de decisiones basada en reglas preconstruidas. La mayoría de los sistemas no presupone una cierta teoría global de la inteligencia humana, ni pretende simular la función del cerebro, más bien, se limita a aprovechar técnicas algorítmicas existentes para resolver una situación concreta.

a otras disciplinas y centros de atención de alta complejidad, con enormes beneficios tanto en el sector público como en el privado.

3. Aplicación del proyecto al modelo clínico de interés

La TGA uno de los diagnósticos congénitos más complejos por su manejo postoperatorio. Si adicionalmente se considera la gran cantidad de variables que se requiere integrar al realizar el ejercicio médico de su atención; encontramos explicación a situaciones cotidianas en servicios asistenciales como las UCIs donde son comunes los errores de comunicación, la pérdida e incorrecta transcripción de datos, y la demora en la obtención de registros que alerten sobre anomalías y permitan anticiparse a posibles complicaciones en los pacientes [1]. Todo esto hace estos servicios particularmente susceptibles al error médico y el ambiente ideal para la aplicación de una serie de herramientas informáticas tradicionales o basadas en técnicas de IA, que permitan una mayor eficiencia en el cuidado de los pacientes.

4. Metodología y descripción de las etapas del proyecto

El desarrollo del sistema se planeó realizar por ciclos que constituyeran cada uno, una versión del producto. Cada ciclo se dividió en cuatro fases (inicio, elaboración, construcción y transición) que a su vez se dividieron en iteraciones que se planearon desarrollar a lo largo de 5 flujos de trabajo fundamentales: requisitos, análisis, diseño, implementación y pruebas.

Fase de elaboración:

En esta fase la labor primordial fue la selección de una arquitectura estable con el fin de planificar adecuadamente las labores de construcción. Se definieron aspectos como motor de inferencia, lenguaje de programación y plataforma.

Motor de inferencia: se analizaron catorce diferentes shells de motores de inferencia en aspectos como plataforma, lenguaje, estrategia de búsqueda, forma en que elige el conocimiento y posibilidad de incorporar metaconocimiento.

Al finalizar nuestro análisis se escogieron Jess y Clips como posibles shells¹⁷ ya que cumplían con características como trabajo en plataformas Linux y Windows, integración con java¹⁸ como lenguaje de programación, sistema de producción por encadenamiento hacia delante (de las mejores estrategias encontradas para diagnóstico médico)[4] y tenían implementado Rete como algoritmo de búsqueda.

¹⁷ El intérprete de comandos usado para interactuar con el núcleo de un sistema operativo.

¹⁸ Java es un lenguaje de programación libre, independiente de la plataforma, muy extendido y con mucha importancia en el ámbito de Internet. [15]

Algoritmo Rete: del tiempo de ejecución el 90% se consume en el proceso de emparejamiento. El algoritmo Rete se basa en dos observaciones (suposiciones):

La memoria de trabajo es muy grande y cambia poco entre cada ciclo. El esfuerzo de emparejamiento depende de la razón de cambio de la memoria de trabajo en lugar del tamaño de ésta. Las condiciones de muchas reglas son similares. Rete procesa (compila) las reglas antes de ser usadas, localizando condiciones comunes y eliminando todas menos una [5].

Fue diseñado para facilitar el análisis temporal estático de los programas que lo usan, así propiedades como la no duplicación, la novedad, la especificidad y la prioridad de operación, se preservan. [6][7]

Fase de construcción:

Se construyó el primer prototipo de TGA.

Construcción de la base de conocimiento: se diseñaron e implementaron las reglas del protocolo de manejo postoperatorio de TGA que incluyen aspectos como:

Alarmas generadas por alteración de una variable.

El especialista tiene la posibilidad de mirar la alteración de una variable en especial y el sistema le muestra un listado de los posibles diagnósticos asociados con la alteración de esta variable, teniendo en cuenta prioridades en las complicaciones posibles. Cuando se escoge uno de la lista, el sistema muestra un paralelo entre los signos asociados con los signos y síntomas que presenta el paciente para que el especialista pueda decidir si acepta o no el diagnóstico. De esta manera el especialista va construyendo reglas en la práctica clínica y se está generando un histórico para alimentar el sistema de aprendizaje a realizar en una etapa posterior.

Otro tipo de reglas muestran consideraciones a tener en cuenta por posibles complicaciones dependiendo de la edad y peso del paciente.

Un conjunto de reglas sugieren tratamientos a seguir dependiendo del tiempo del postoperatorio, edad, peso y signos del paciente.

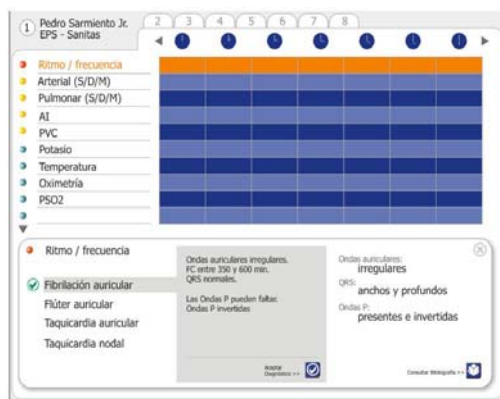


Fig. 1. Interfaz donde se pueden ver las alertas generadas por las variables, los diagnósticos asociados y los datos que el paciente presenta.

Hasta el momento se han realizado pruebas experimentales en dos niños en este tipo de postoperatorio obteniendo resultados favorables y observando que estas mismas reglas se pueden generalizar a casi todos los postoperatorios cardiovasculares de la UCI.

El sistema esta en capacidad de conectarse al monitor de signos vitales diseñado por el grupo de bioingeniería de la FCV [8], ventaja que le permite al SE disparar alarmas en tiempo real. Adicionalmente esta conectado a la base de datos de la HCE¹⁹ de donde obtienen los valores de los signos no monitoreados en tiempo real.

Vale la pena resaltar que el diseño de este prototipo se dio gracias a un trabajo interdisciplinario de médicos e ingenieros, que inicialmente acordaron las características principales del sistema y visualizaron la necesidad de realizar un estudio de usabilidad²⁰ dada la complejidad de los sitios donde va a ser empleado.

Se considera que las fases de inicio, elaboración y construcción deben repetirse hasta que exista un sistema lo suficientemente robusto para volver a realizar pruebas. El paso siguiente fue el desarrollo de una interfaz de usuario para el ingreso de reglas y visualización de resultados teniendo en cuenta el estudio de usabilidad. Paralelamente a este proceso se probaran las reglas implementadas para el manejo de pacientes operados de TGA.

Sistema de aprendizaje: será desarrollado bajo el marco teórico que existe acerca de las Máquinas de Soporte Vectorial (MSV). Los progresos importantes en teoría de aprendizaje basados en estadística han introducido nuevos paradigmas tales como MSV cuya principal característica es la estructura de un algoritmo de aprendizaje el cual consiste en la solución de un simple problema cuadrático. [9]. Las MSV se pueden ver también como un nuevo método para el

¹⁹ Sistema software de Historia Clínica Electrónica.

²⁰ **Usabilidad** es la efectividad, eficiencia y satisfacción con la que un producto permite alcanzar objetivos específicos a usuarios específicos en un contexto de uso específico" ISO/IEC 9241.

entrenamiento de modelos polinómicos, redes neuronales, modelos borrosos, entre otros [9].

5. Discusión de los resultados preliminares

Al revisar la historia de los sistemas basados en el conocimiento, la mayoría de ellos son extremadamente pequeños en cuanto al volumen de sus bases de conocimiento y base de hechos. Es decir, el universo de estos sistemas es simple y limitado, y en la mayoría de los casos se centran exclusivamente en resolver una situación o un problema determinado. Partiendo del reconocimiento de esa situación de estrechez en el ámbito de los sistemas basados en el conocimiento, el presente proyecto pretende construir una base de conocimiento de un volumen radicalmente mayor en diagnóstico y tratamiento de enfermedades cardiovasculares. Adicionalmente puede llegar a tener un gran impacto clínico si se asocia con una disminución significativa de indicadores como la mortalidad en postoperatorios cardiovasculares. La complejidad de plasmar el conocimiento médico requiere de mucho tiempo y dedicación, el presente proyecto le permite al especialista construir las reglas durante la práctica clínica. El presente proyecto, podría en un futuro ser extendido a otras disciplinas y centros de atención de alta complejidad, lo cual tendría un gran impacto en el campo asistencial.

Referencias

- [1] Leval Marc, MD. Carthey Jane, PhD. Wright David, PhD. And all United Kingdom pediatric cardiac centers. Human Factors and Cardiac Surgery: a multicenter study. Toracic Cardiovasc. Surg. -01-APR-2000;119(4 pt) 661-72 from NIV MEDLINE.
- [2] E. Castillo, J.M. Gutiérrez, and A.S. Hadi. Expert Systems and Probabilistic Network Models. Springer Verlag, New York, 1997. 600 pages.ISBN: 0- 387-94858-9.
- [3] Williams Christopher K. I., Quinn John, McIntosh Neil. Factorial Switching Kalman Filters for Condition Monitoring in Neonatal Intensive Care. November 2005.
- [4] Montagut Martha Vitalia. Principios de inteligencia artificial y sistemas expertos. Ediciones UIS 2000.
- [5] L. Martin, W. Taylor, S. Meadows and K. Freeman. CLIPS Application Abstracts. November 1st 1997.
- [6] Third Conference on CLIPS. September 12-14, 1994. Lyndon B. Johnson Space Center. 19 – 22.
- [7] CLIPS Version 5.1 User's Guide, NASA Lyndon B. Johnson Space Center, Software. Technology Branch, Houston, TX, 1991.
- [8] Proyecto aprobado por COLCIENCIAS. “Diseño y construcción de un prototipo para monitoreo presencial y remoto de signos vitales de pacientes en estado crítico”. Contrato No. 332-2004, Código No. 6566-14-172117. Fundación Cardiovascular de Colombia.
- [9] Vladimir N. Vapnik, The Nature of Statistical Learning Theory. Springer, 1998.