

Sistema para la gestión de información de los pacientes de la sala de Neonatología del Hospital Pediátrico Provincial de Holguín Octavio de la Concepción y la Pedraja (NeoGest).

Trabajo de diploma para optar por el título de Ingeniero Informático

Autor

José Luis Pérez Céspedes

Tutores:

M.Sc. María del Pilar Céspedes Gamboa

Ing. Yurima Isbet Pérez Calzadilla

Consultante.

Ing. Oscar Gabriel Reyes Pupo

“La inteligencia consiste no sólo en el conocimiento, sino también en la destreza de aplicar los conocimientos en la práctica”.

Aristóteles

Agradecimientos

A mis queridos padres y hermanas, por su apoyo y amor, solo ellos saben cuánto me he esforzado para llegar hasta aquí.

A mi novia por su apoyo y comprensión en los momentos difíciles.

A Dios por su bendición.

A Yurima y Oscar por su guía y apoyo incondicional.

A todos aquellos que hicieron posible este resultado.

DEDICATORIA

A mis padres, mis hermanas y familia en general por su apoyo, amor y confianza incondicional.

A mi Gretel por su amor, apoyo y por creer en mí en todo momento.

A Dios por ser Todo.

Resumen

En las últimas décadas, el desarrollo y uso generalizado de la computación y una creciente inserción de las comunicaciones en el ámbito del ciberespacio han provocado un aumento del uso de las aplicaciones Web, permitiendo acceder a la información de forma organizada y gestionar de una manera eficiente el flujo de trabajo en buena parte de una o varias áreas de una empresa o institución.

El servicio de Neonatología del Hospital Pediátrico provincial de Holguín Octavio de la Concepción y de la Pedraja, tiene deficiencias para gestionar la información de los pacientes ingresados en la misma y no cuenta con un sistema capaz de gestionar eficientemente esta información.

En este trabajo se describe el desarrollo de la investigación que se realiza en dicho servicio. Se desarrolló una aplicación Web con el objetivo de favorecer la gestión de la información de los pacientes ingresados, logrando un medio que la unifique, garantizando la integridad, confiabilidad y consistencia de los datos.

Para el desarrollo del sistema para la gestión de la información del servicio de Neonatología (NEOGEST), se empleó la metodología de desarrollo de software Rational Unified Process (RUP) y el lenguaje unificado de modelado (UML), El Entorno de Desarrollo Integrado (IDE, por sus siglas en Inglés) y el framework ZK para el diseño, implementación y desarrollo en general del sistema, el Sistema Gestor de Base de Datos PostgreSQL y como servidor Web el Apache Tomcat.

Abstract

In recent decades, the development and widespread use of computing and an increasing integration of communications in the field of cyberspace have led to increased use of Web applications, enabling access to information in an organized and efficiently manage workflow across much of one or more areas of a company or institution.

The neonatology ward in the Pediatric Hospital of Holguín “Octavio de la Concepción and the Pedraja”, has deficiencies in managing information of patients admitted to the same and does not have a system that can efficiently manage this information.

This paper describes the development of the research being done in the service. Web application was developed with the aim of promoting information management of hospitalized patients, achieving a unifying medium that ensures the integrity, reliability and consistency of the data.

To develop the system for information management on Neonatology service (NEOGEST) was used in the software development methodology, Rational Unified Process (RUP) and Unified Modeling Language (UML), the Integrated Development Environment (IDE) and the ZK framework for the design, implementation and overall development of the system, management system PostgreSQL database as the Apache Tomcat Web server.

Índice

INTRODUCCIÓN	1
CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA.....	7
Introducción	7
1.1 Descripción general del objeto de estudio.	7
Personal de la sala de neonatología	7
Organigrama del Servicio Neonatal del HPUH.....	8
1.2 Modelo del negocio.....	9
1.2.1 Descripción de los procesos del negocio.	9
1.2.2 Procesos del negocio	10
1.2.3 Reglas del negocio a considerar	11
1.2.4 Actores del negocio.....	12
1.2.5 Trabajadores del negocio.....	13
1.2.6 Diagrama de los casos de usos del negocio	13
1.2.7 Casos de uso a expandir con su descripción textual, diagrama de Actividades y de clases del modelo de objeto.....	14
1.3 Sistemas informáticos actuales vinculados al campo de acción.....	22
1.4 Tendencias y tecnologías actuales.....	23
1.4.1 Arquitectura cliente –servidor.....	23
1.4.2 Las aplicaciones Web	24
1.4.3- Software libre	25
1.4.4 Tecnologías y herramientas para el desarrollo de aplicaciones Web...	26
1.4.5- Servidores Web.....	29
1.4.6 Lenjuages de Programación para la Web.	30
1.4.7 Frameworks Web basados en JAVA para el desarrollo de aplicaciones Web.....	34
1.4.8 Sistema gestores de bases de datos.	38
1.4.9 Herramientas para Reportes.	41
1.5 Metodologías de ingeniería de software.	45
1.6- Conclusiones del capítulo.....	51

CAPÍTULO 2 Descripción y elaboración de la solución propuesta.....	52
2.1 Captura de Requerimientos.....	52
2.1.1 Requisitos funcionales	52
2.1.2 <i>Definición de los requisitos no funcionales</i>	54
2.2 Actores del sistema	56
2.3 Diagrama de los Casos de Uso del Sistema.....	57
2.3.1 Diagrama de Casos de Uso del Sistema del paquete de Seguridad.....	58
2.3.2 Diagrama de Casos de Uso del Sistema del paquete Paciente y Reporte	59
2.4 Diagrama de Generalización Especialización entre Actores del Sistema ...	59
2.5 Descripción Textual de los Principales Casos de Uso del Sistema en formato real y expandido	60
2.6 Valoración de Sostenibilidad del Producto informático.	68
2.7 Flujo de diseño	72
2.7.1 Organización de los casos de uso a desarrollar por paquetes.....	72
2.8 Diagrama de despliegue.....	74
2.9 Flujo de implementación	75
Seguridad del Sistema.....	78
2.10 Modelo de prueba	79
2.10.1 Casos de pruebas	80
2.10.2 Procedimientos de pruebas.....	80
2.11 Valoración de los resultados obtenidos	81
2.11.1 Encuestas	81
2.11.2 Entrevista	82
Conclusiones del capítulo.....	83
Conclusiones Generales	84
Recomendaciones.....	85
Glosario de términos	86
Bibliografía y Referencias Bibliográficas	90
ANEXOS	93
ANEXO 1 Entrevista preliminar realizada al cliente.....	93

Estudio del negocio. Captura de requisitos	93
<i>ANEXO 2 Descripciones textuales de los CUN. Diagrama de Actividades y del Modelo de Objeto</i>	94
<i>ANEXO 3 Descripciones textuales en formato real y expandido de los CUS</i>	110
<i>ANEXO 4 Tablas y Fórmulas del COCOMO. Resultados</i>	122
<i>ANEXO 5 Diagramas de Clases del Diseño Web.</i>	127
<i>ANEXO 6 Diagrama de Clases Persistentes del Sistema NeoGest</i>	131
<i>ANEXO 7 Diagrama de Componente del Sistema NeoGest (paquete paciente y reporte)</i>	132
Diagrama de Componentes Sistema NeoGest (paquete Seguridad)	133
<i>ANEXO 8 Tablas de los Casos de Pruebas y los Procedimientos de Pruebas</i> ...	134
<i>ANEXO 9 Encuesta de opinión de los usuarios del sistema NeoGest</i>	140
ANEXO 10 Procesamiento de la encuesta de opinión de los usuarios aplicando el método Delphi	142
<i>ANEXO 11 Entrevista realizada al cliente.</i>	147

Índice de figuras

Figura 1 Personal del servicio	8
Figura 2 Organigrama del servicio	8
Figura 3 Diagrama de casos de usos del negocio.....	14
Figura 4 Diagrama de Actividades del CUN Examinar paciente	17
Figura 5 Diagrama de clases del modelo de objeto del CUN (Examinar Paciente)	18
Figura 6 Descripción textual DEL C.U.N (Confeccionar reporte estadístico)	20
Figura 7 Diagrama de actividades del CUN Confeccionar reporte estadístico	21
Figura 8 Diagrama de clases del modelo de objeto del CUN (Confeccionar Reportes Estadísticos.	22
Figura 9 Esquema de flujo de creación de informes en BIRT	42
Figura 10 Esquema del mecanismo de generación de informes en JasperReports	43
Figura 11 Flujo de trabajo de RUP	46
Figura 12 Diagrama de paquetes del Sistema NeoGest	57
Figura 13 Diagrama de Casos de Uso del Sistema del paquete de Seguridad.....	58
Figura 14 Diagrama de Casos de Uso del Sistema del paquete Paciente y Reporte	59
Figura 15 Diagrama de Generalización Especialización entre Actores del Sistema	60
Figura 16 Diagrama de Clases del diseño del paquete del menú principal	73
Figura 17 Diagrama de despliegue	75
Figura 18 Diagrama de Actividades del CUN Verificar estado evolutivo	95
Figura 19 Diagrama de clases del modelo de objeto del CUN Verificar estado evolutivo.	95
Figura 20 Diagrama de Actividades del CUN Realizar análisis	97
Figura 21 Diagrama de clase del modelo de objeto del CUN Realizar análisis.....	97
Figura 22 Diagrama de actividades del CUN Registrar fallecimiento paciente	99
Figura 23 Diagrama de clase del modelo de objeto del CUN Registrar fallecimiento paciente	99

Figura 24 Diagrama de actividades del CUN Realizar traslado paciente	102
Figura 25 Diagrama de clase del modelo de objeto del CUN Realizar traslado paciente.....	102
Figura 26 Diagrama de Actividades del CUN Establecer alta paciente	104
Figura 27 Diagrama de clase del modelo de objeto del CUN Establecer alta paciente.....	104
Figura 28 Diagrama de Actividades del CUN Registrar infección nosocomial	106
Figura 29 Diagrama de clase del modelo de objeto del CUN Registrar infección nosocomial	107
Figura 30 Diagrama de Actividades del CUN Registrar Ventilación paciente	109
Figura 31 Diagrama de clase del modelo de objeto del CUN Registrar ventilación paciente.....	109
Figura 32 Diagrama de clases del diseño del paquete1 (Gestionar Ficha Paciente). Inserción y modificación	127
Figura 33 Diagrama de clases del diseño del paquete 2 Realizar Búsquedas por Criterios	128
Figura 34 Diagrama de clases del diseño del paquete 3 Visualizar Reportes.....	129
Figura 35 Diagrama de clases del diseño del paquete 4 Gestionar Usuarios	130
Figura 36 Diagrama de Clases Persistentes del Sistema NeoGest	131
Figura 37 Diagrama de Componente del Sistema NeoGest (paquete Paciente y Reporte)	132
Figura 38 Diagrama de Componentes Sistema NeoGest (paquete Seguridad) ..	133

Índice de tablas

Tabla 1 Actores del negocio	12
Tabla 2 Trabajadores del negocio	13
Tabla 3 Descripción textual del CUN Examinar paciente	16
Tabla 4 Comparación entre las metodologías de modelación RUP e Iconix	50
Tabla 5 Actores del sistema	57
Tabla 6 Descripción textual del CUS Actualizar Ficha Paciente.....	65
Tabla 7 Descripción textual del CUS Realizar Búsqueda por Criterios	67
Tabla 8 Estándar o estructura de codificación de los componentes ZK	77
Tabla 9 Descripción textual DEL C.U.N (Verificar estado evolutivo)	94
Tabla 10 Descripción textual DEL C.U.N (Realizar análisis)	96
Tabla 11 Descripción textual DEL C.U.N (Registrar fallecimiento paciente)	98
Tabla 12 Descripción textual DEL C.U.N (Realizar traslado paciente).....	101
Tabla 13 Descripción textual DEL C.U.N (Establecer alta paciente)	103
Tabla 14 Descripción textual DEL C.U.N (Registrar infección nosocomial)	106
Tabla 15 Descripción textual DEL C.U.N (Registrar ventilación paciente)	108
Tabla 16 Descripción Textual en formato real y expandido del Caso de uso del sistema Gestionar Usuarios del Sistema.....	115
Tabla 17 Descripción Textual en formato real y expandido del CUS Visualizar Reportes.....	117
Tabla 18 Descripción Textual en formato real y expandido del CUS Mostrar Gráficos de Datos Comparativos por Rangos de Fecha	119
Tabla 19 Descripción textual del CUS Validar Usuario	120
Tabla 20 Descripción textual del CUS Autenticar Usuario	121
Tabla 21 Descripción textual del CUS Realizar Backup automático y salva cruzada de la BD.....	121
Tabla 22 Salidas externas(EO)	122
Tabla 23 Entradas externas(EI).....	122
Tabla 24 Ficheros Lógicos Internos	123
Tabla 25 Puntos de función desajustados.....	123

Tabla 26 Características	124
Tabla 27 Tabla Multiplicadores de esfuerzo	124
Tabla 28 Tabla de Factores de escala	125
Tabla 29 Tabla de Valores calibrados	125
Tabla 30 Caso de prueba Realizar búsqueda por Criterios (caso1)	134
Tabla 31 Caso de prueba Realizar búsqueda por criterios (caso 2).....	134
Tabla 32 Caso de prueba Actualizar ficha del paciente (caso1).....	135
Tabla 33 Caso de prueba Actualizar ficha del paciente (caso2).....	135
Tabla 34 Caso de prueba Actualizar ficha del paciente (caso3).....	136
Tabla 35 Caso de prueba Actualizar ficha del paciente (caso 4).....	136
Tabla 36 Caso de prueba Actualizar ficha del paciente (caso 5).....	137
Tabla 37 Procedimiento de prueba Realizar búsqueda por criterios	138
Tabla 38 Procedimiento de prueba insertar ficha del paciente.....	139
Tabla 39 Procedimiento de prueba Actualizar Ficha del paciente (escenario modificar).....	139
Tabla 40 Aspectos a medir en encuesta de opinión de los usuarios del sistema NeoGest	141
Tabla 41 Tabla de frecuencia absoluta	142
Tabla 42 Tabla de frecuencia absoluta acumulada	143
Tabla 43 Tabla del inverso de la frecuencia absoluta acumulada	144
Tabla 44 Tabla de determinación de los puntos de cortes	145
Tabla 45 Tabla conclusiones generales (Método Delphi).....	146

Introducción

INTRODUCCIÓN

Los constantes avances tecnológicos en el campo de la Informática, han ido abarcando de manera general al mundo en cuanto a necesidades de uso de ordenadores y sistemas que realicen y permitan acciones que faciliten de forma efectiva el trabajo.

El uso de las Tecnologías de la Informática y las Comunicaciones (TIC), permite diversas funciones que hacen al hombre actual más profesional en su trabajo. Además posibilita estar en contacto globalmente mediante Internet. Llegando a convertirse dichas tecnologías en una fuente de información imprescindible para la vida. Lo anteriormente mencionado es un hecho que surge debido a los numerosos beneficios que brindan las TIC: interactividad, instantaneidad, elevados parámetros de calidad de imagen y sonido y automatización.

Casi es cuestión de principios que empresas u otras instituciones, que en sus actividades diarias requieran de organización, empleen tecnologías para obtener logros, ya sea en promoción, gestiones en sus actividades, correo electrónico, etcétera.

El desarrollo constante de sistemas se realiza cada vez con más destreza. Estos gestionan, solucionan y/o simulan muchos procesos de cualquier entidad para el que ha sido creado.

El Gobierno Cubano se ve inmerso en un proceso de informatización de los diferentes sectores de la sociedad. En este proceso se le ha dado prioridad a la salud, ya que en política de esta índole, Cuba muestra resultados comparables con los de países desarrollados de acuerdo a sus cifras de Mortalidad Infantil y es posible que en un futuro no lejano pueda ocupar el primer lugar del continente, según Fidel, quien apunta que las perspectivas de vida se han elevado en casi 18 años, a partir de los 60 posterior al triunfo de la revolución en enero de 1959. (Ramonet 2006)

Desde los primeros años de la revolución se tomó la decisión, por el máximo nivel del país, de conformar un Sistema Nacional de Salud que contribuyera a elevar el estado de salud de la población, lo que se materializó con la ejecución de un gran número de programas, entre los cuales se incluyó el Programa de Atención

Introducción

Materno Infantil (PAMI); confluyendo en el mismo los resultados de todas las acciones de éste y el resto de los diferentes programas para preservar la salud tanto de la madre como del niño.(Dotres 2003)

Luego de la creación de la especialidad de Neonatología en Cuba se incrementa el personal profesional que trabaja en estos servicios, teniéndose 53 servicios de esta especialidad ubicados a todo lo largo del país, de ellos 41 brindan atención de intensivismo neonatal y 12 brindan atención especializada al neonato no crítico. Los estudios de seguimiento del neonato o recién nacido (RN) de alto riesgo comenzaron desde la segunda mitad de la década del 70 y se han ido perfeccionando en los últimos 20 años. Se han dedicado fundamentalmente a los aspectos inherentes a problemas respiratorios, nutricionales y desviaciones del neurodesarrollo, lo que ha permitido obtener mejores resultados en la detección y tratamiento de los problemas del RN. (Dueñas 2009)

El Hospital Pediátrico Provincial Universitario de Holguín(HPUH) Octavio de la Concepción de la Pedraja, ubicado en Holguín avenida de los Libertadores No 91, perteneciente al Ministerio de Salud Pública MINSAP, con un área de atención de los 14 municipios de la provincia y centro de referencia para la provincia de Guantánamo, cuenta con una dotación de 540 camas, 127 dedicadas a la atención quirúrgica y 413 a la atención pediátrica con un promedio anual de 21000 ingresos y 6000 operaciones.

Tiene como misión el mejoramiento de la situación de salud de la población pediátrica a través de la mejor calidad de los servicios brindando atención integral e integrada a sus áreas de influencia propiciando la participación comunitaria y la autogestión, en correspondencia con criterios de prioridad del riesgo epidemiológico y de los diferentes grupos vulnerables, con utilización eficiente de los recursos humanos motivados y con capacitación actualizada, aplicando la tecnología más adecuada y promoviendo el grado de satisfacción de forma permanente.

El HPUH brinda oportunidades de atención en todos los servicios clínicos quirúrgicos en la edades pediátricas, dentro de estos servicios se encuentra el de Neonatología, en dicho servicio es de vital importancia almacenar la información

Introducción

de los pacientes que ingresan a la sala de la cual se extraen datos que posteriormente son utilizados en la confección de reportes estadísticos exigidos por la dirección de salud de la provincia.

En el quehacer diario del servicio se ingresan pacientes, a los mismos se le confeccionan historias clínicas, fichas de información, se le realizan análisis de distinta índole, se consultan con otras especialidades, constantemente se analiza el estado evolutivo de los pacientes y de acuerdo a este se le procede a dar alta a trasladar de entidad o servicio o a realizar la necropsia en caso de fallecimiento. A fin de mes se realizan reportes de distintos criterios y tipos que se sustentan de los documentos confeccionados por parte del personal encargado, con el objetivo de informar a la Dirección del Hospital del estado de los pacientes así como probar el grado de efectividad y desempeño del servicio.

En la actualidad el servicio no cuenta con un mecanismo que organice y gestione toda la información referente a los pacientes, esta queda registrada en papeles y el proceso de gestión de la misma se realiza de forma manual. Debido a la gran cantidad de pacientes que se atienden anualmente el volumen de información almacenada es sustancialmente grande. Todo esto provoca las deficiencias siguientes:

1. Casi en su totalidad la información referente a los pacientes se registra de forma manual lo que provoca el rápido deterioro de los documentos que almacenan dicha información.
2. La actualización y corrección de errores en la información recogida le dificulta la rapidez en el desempeño de los procesos que se llevan a cabo diariamente en el servicio.
3. El proceso de búsqueda de datos de un paciente se ralentiza por realizarse de forma manual.
4. Los datos recogidos por los especialistas para la confección de las historias clínicas y la ficha de los pacientes en ocasiones no son uniformes, algunas de estas cuentan con errores, así como un cierto grado de ilegibilidad y falta de organización.

Introducción

5. La elaboración de reportes estadísticos por parte del personal encargado se torna demasiado lento y trabajoso, además de la posibilidad de cometer errores a la hora de realizar los cálculos pertinentes.
6. La seguridad en lo concerniente a la información en su totalidad es deficiente ya que cualquiera puede acceder a los registros y alterar su estado.
7. No hay posibilidades de acceso a la información fuera de la entidad.

La situación antes planteada dio paso a formular el siguiente **problema científico**: ¿Cómo mejorar el proceso de gestión de información (PGI) del servicio de Neonatología del HPUH?

El **objeto** de la investigación es el proceso de gestión de información de los pacientes del servicio de neonatología del HPUH.

Con el propósito de darle solución a la problemática antes expuesta el **objetivo** de la presente investigación es: Desarrollar un producto informático (PI) para mejorar el PGI de los pacientes que ingresan en el servicio de Neonatología del HPUH Octavio de la Concepción y la Pedraja.

El **campo de acción** en que se enmarca la investigación es la Informatización del PGI de los pacientes del servicio de Neonatología del HPUH.

Para guiar la investigación se plantea la **hipótesis**:

La elaboración de un PI, que se caracterice por su facilidad de uso, que disminuya el tiempo empleado en la generación de informes o reportes así como de búsquedas de información, mejore la integridad, seguridad y disponibilidad de los datos, favorecerá el PGI de los pacientes del servicio de Neonatología del HPUH.

La investigación transcurrió a través de las siguientes **tareas**:

1. Diagnosticar el estado del proceso de gestión de información de los pacientes del servicio de neonatología del HPUH.
2. Elaborar las bases teóricas del proceso de gestión de información de la sala y de las tendencias y tecnologías actuales disponibles para el desarrollo de los sistemas de gestión de información.
3. Capturar los requisitos del sistema.
4. Diseñar el producto informático

Introducción

5. Valorar la sostenibilidad del producto informático.
6. Implementar, probar e implantar el producto informático.
7. Valorar el grado de efectividad y aceptación por parte del cliente del sistema.

Para cumplimentar estas tareas se emplearon los métodos teóricos y empíricos de la investigación científica. Los **métodos teóricos** empleados fueron:

- Histórico-lógico el cual permitió recopilar toda la información relacionada con los procesos de gestión de información. Además creó las bases para la confección del módulo informático que se propuso como solución a la situación problemática.
- Análisis y síntesis: se empleó para el estudio y diagnóstico de la situación existente referente a la gestión de los informes de la sala, permitiendo determinar las características fundamentales del mismo y profundizar en el proceso que es el objeto de estudio de la investigación.
- Hipotético-deductivo: se utilizó en la elaboración de la hipótesis y su posterior evaluación, la modelación para la construcción abstracta de la realidad, la cual es de vital importancia en esta investigación, ya que permitió ir confeccionando la solución del problema.

Entre los **métodos empíricos** que se emplearon se encuentran:

- Revisión de documentos, que sirvió para realizar un estudio detallado de los procesos de gestión de información, con el propósito de conocer como se manifiesta el mismo realmente.
- Entrevista: fue utilizada en la recopilación de información mediante una conversación profesional con los futuros usuarios y para la captura de requisitos.
- Encuesta: permitió valorar el grado de aceptación del cliente, con la propuesta de solución planteada.

Se empleó el **método estadístico** Delphi para el procesamiento de la encuesta en vista a la validación de la hipótesis.

Introducción

El documento está estructurado en dos capítulos. En el capítulo 1 se exponen las bases teóricas que sustentan la investigación, conceptos relacionados con el objeto de estudio, luego se describen las principales herramientas que se utilizan para la elaboración de la propuesta y la metodología de desarrollo de software utilizada en la investigación.

El capítulo 2 resume la elaboración de la solución propuesta, modelando el análisis, diseño e implementación, además de la valoración de sostenibilidad del sistema informático resultante según la dimensión administrativa, socio-humanista, medio ambiental y tecnológica entre otros resultados.

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

Introducción

En este capítulo se realiza un análisis del objeto de estudio de la investigación para alcanzar un mayor conocimiento acerca de los procesos del negocio sobre los cuales se sustenta la gestión de información de los pacientes en la sala de neonatología. Luego se describen las principales características del mismo, las tendencias y tecnologías actuales y las herramientas que se utilizaron para la elaboración de la propuesta.

1.1 Descripción general del objeto de estudio.

El servicio de neonatología se encuentra dentro del hospital pediátrico provincial Octavio de la Concepción y de la Pedraja ubicado en Holguín avenida de los Libertadores No 91 perteneciente al Ministerio de Salud Pública MINSAP, en este servicio ingresan todos los niños menores de 28 días nacidos en la provincia de Holguín y todos los quirúrgicos de las provincias orientales que cumplan con el tiempo establecido, estos deben ser remitidos del hospital de origen con previo diagnóstico de patología quirúrgica.

La misión de esta sala es brindar servicios médicos especializados que les permitan a los neonatos tener un estado de salud satisfactorio y libre de riesgos contando con un equipo de especialistas de suma experiencia.

En cuanto a la visión de la misma, aspira a convertirse en centro de referencia de servicios al recién nacido en el país, así como elevar el nivel o categoría científica de todo el cuerpo de médicos, enfermeras y personal de laboratorio en vista a brindar servicios con un mayor profesionalismo calidad y eficiencia.

En el servicio de Neonatología del HPUH es de vital importancia almacenar la información de los pacientes que ingresan a la sala de la cual se extraen datos que posteriormente son utilizados en la confección de reportes estadísticos exigidos por la dirección de salud de la provincia.

Personal de la sala de neonatología

El servicio de neonatología cuenta con un personal como se muestra a continuación:

Fundamentación Teórica

- 12 médicos especialistas (2 hombres y 10 mujeres).
- 40 enfermeras (distribuidas en 4 equipos, todas mujeres).
- 4 técnicos de laboratorio (mujeres).
- 1 recepcionista (mujer).

A continuación gráfico del personal del servicio:

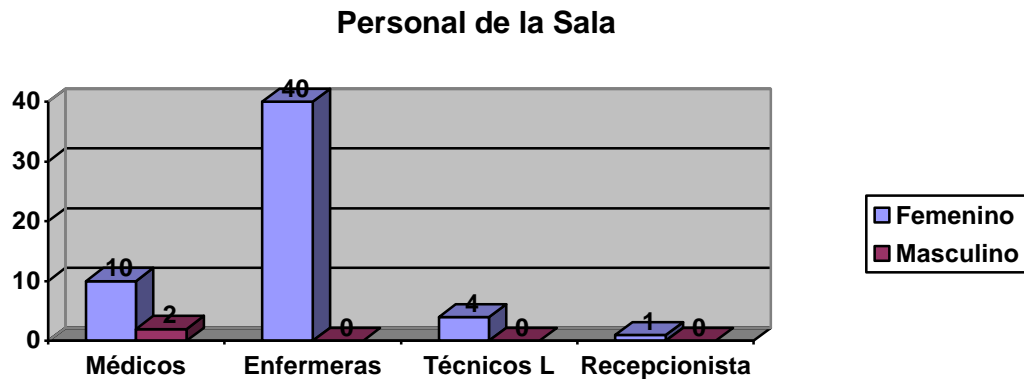


Figura 1 Personal del servicio

El servicio de neonatología posee una distribución en relación a las funciones que desempeñan los trabajadores como se muestra en la siguiente figura 2:

Organigrama del Servicio Neonatal del HPUH.

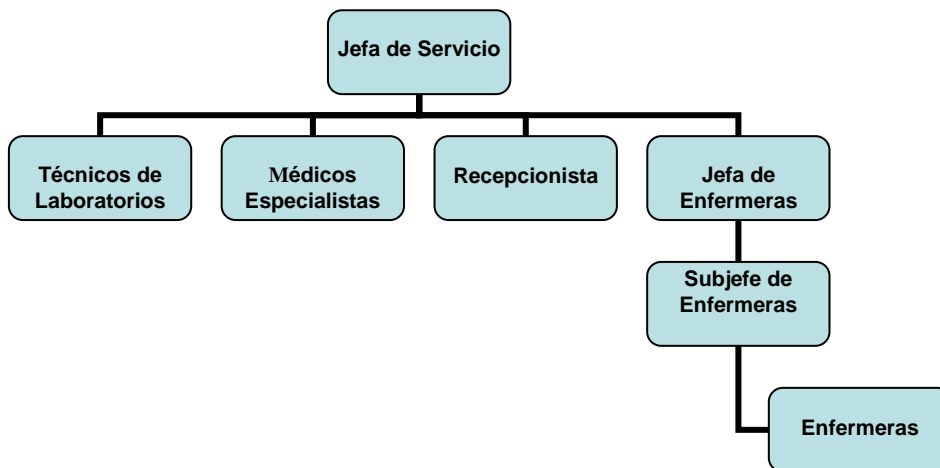


Figura 2 Organigrama del servicio

1.2 Modelo del negocio

1.2.1 Descripción de los procesos del negocio.

En la sala de neonatología del Hospital Pediátrico Octavio de la Concepción y de la Pedraja de Holguín se ingresan los neonatos enfermos (menores de 28 días) nacidos en la provincia de Holguín y todos los quirúrgicos de las cinco provincias orientales que cumplan con el tiempo establecido, estos deben ser remitidos del hospital de origen con previo diagnóstico de patología quirúrgica.

Una vez que llegan al servicio, el médico de guardia lo examina y determina si se debe ingresar o no. Si determina ingresarlo se le confecciona una historia clínica al paciente, en donde queda registrado nombre, edad, procedencia, peso, sexo entre otros datos, y una ficha del paciente en donde queda registrado (nombre, apellidos, número de historia clínica, fecha de nacimiento, fecha de ingreso, edad, patología, edad gestacional, estado de ingreso, Apgar (puntaje establecido al momento del nacimiento del niño, el cual es proporcional al estado satisfactorio del mismo), entre otros datos. Luego de ingresar se le crea al paciente una tarjeta de identificación la cual consta de datos generales como nombre y apellidos procedencia, peso al nacer, fecha nacimiento, fecha de ingreso, diagnóstico, cama y un estado de acuerdo a las condiciones del paciente (regular, grave, crítico).

Para determinar con exactitud el diagnóstico se le hace un examen físico minucioso, análisis complementarios, se interconsulta con otras especialidades de ser necesario, se estabiliza al paciente con todas las medidas encaminadas a ello, luego de lo cual se procede a intervenir quirúrgicamente de ser necesario por parte de cirujanos.

En caso de requerir ventilación estos pacientes se lleva un registro donde se plasma, además de los datos generales a los que se hizo referencia en la ficha del paciente, se registra fecha de inicio de la ventilación, edad en este momento, modalidad ventilatoria, fecha de retiro de la misma, complicaciones, tiempo de ventilación, diagnóstico y contrarreferencia.

Fundamentación Teórica

En ocasiones pueden aparecer infecciones adquiridas en el hospital, también llamadas nosocomiales, las cuales son recogidas en otro registro, llevándose los siguientes datos: fecha de comienzo de la infección, fecha de fin de la infección, factores de riesgo, tipo de infección y diagnóstico base.

De acuerdo a su evolución, el paciente puede irse de alta, trasladarse a otra sala o servicio, o fallecer, a este último es muy importante realizarle la necropsia que confirmaría las causas de muerte y este dato también debe aparecer en la ficha del paciente.

Los médicos diariamente verifican el estado evolutivo del paciente y se procede a darle el alta si este es satisfactorio y seguimiento por consulta de ser necesario.

Al terminar cada mes la jefa de servicio, realiza un reporte relacionando el total de pacientes ingresados (clínicos y quirúrgicos), el total de fallecidos tanto clínicos como quirúrgicos, cantidad de graves y ventilados además de una comparación teniendo en cuenta el año anterior. El manejo de esta información relacionada con los reportes estadísticos solo podrá ser manipulado y actualizado por la Jefa de Servicio. En cuanto a los datos estadísticos solo podrán ser accesibles a la Dirección del Hospital, la Jefa de Servicio de la sala y los médicos de la misma.

1.2.2 Procesos del negocio

“Los procesos de negocio son un grupo de actividades lógicamente relacionadas que se llevan a cabo para dar soporte al funcionamiento del mismo, empleando sus recursos para obtener resultados en correspondencia con su misión”(Jacobson 2000).

Los principales procesos que dan soporte al funcionamiento de la sala de neonatología son:

- Examinar paciente
- Realizar análisis
- Verificar estado evolutivo
- Establecer alta paciente
- Realizar traslado paciente
- Registrar fallecimiento paciente
- Registrar ventilación paciente

- Registrar infección nosocomial
- Elaborar reporte estadístico

1.2.3 Reglas del negocio a considerar

Las reglas del negocio son aquellas normas que rigen y velan por el correcto funcionamiento de un negocio dado (Jacobson 2000). A continuación se relacionan las reglas del negocio que caracterizan al proceso de gestión de información de los pacientes de la sala de neonatología del HPUH.

- Todos los pacientes que ingresan tienen que ser menores de 28 días de nacido.
- Los pacientes tienen que ser nacidos en la provincia de Holguín o quirúrgicos provenientes de las cinco provincias orientales.
- Los pacientes tienen que ser remitidos del Hospital de origen con previo diagnóstico.
- Para ingresar un paciente se le confecciona una historia clínica.
- La historia clínica debe contener nombre, edad, procedencia, peso, sexo, entre otros datos.
- Para ingresar un paciente se le confecciona una ficha del paciente.
- La ficha del paciente debe contener (nombre, apellidos, número de historia clínica, fecha de nacimiento, fecha de ingreso, edad, Diagnóstico inicial y final, edad gestacional, reporte, procedencia, contrareferencia, Apgar entre otros datos).
- Cada paciente tiene una tarjeta de identificación.
- La tarjeta de identificación posee los siguientes datos: nombre y apellidos procedencia, peso al nacer, fecha nacimiento, fecha de ingreso, diagnóstico, cama y un estado de acuerdo a las condiciones del paciente.
- Las condiciones de un paciente o reporte pueden ser (regular, grave, crítico).
- En caso de no ingresar el médico confecciona un método con el tratamiento a seguir por el paciente en casa.

Fundamentación Teórica

- Los datos estadísticos no son públicos, a esta información solo tienen acceso la Dirección del Hospital, la Jefa de Servicio y los médicos de la sala.

1.2.4 Actores del negocio

En los procesos de negocio intervienen partes, que se benefician de los resultados obtenidos de los procesos, denominados actores del negocio (Jacobson 2000). A continuación se describe los actores del negocio que se relaciona con los procesos antes mencionados.

Actor del negocio	Justificación
Paciente	Todos aquellos recién nacidos menores de 28 días que presentan anomalías. Inicia el caso de uso del negocio Examinar paciente , interviene en los procesos del negocio Realizar análisis, Realizar traslado paciente, Registrar fallecimiento paciente, Registrar infección nosocomial, Registrar Ventilación, Verificar estado evolutivo y Establecer alta paciente.
Dirección del Hospital	Es un grupo de personas y departamentos interesados en la obtención de los datos estadísticos provenientes de los distintos servicios del HPUH así como del correcto funcionamiento de los mismos. Inicia el caso de uso del negocio Confeccionar reporte estadístico y Realizar Búsquedas.

Tabla 1 Actores del negocio

Fundamentación Teórica

1.2.5 Trabajadores del negocio

También en los procesos de negocio intervienen trabajadores y otras partes de la Empresa que participan directamente en la realización de estos procesos, denominados trabajadores del negocio (Jacobson 2000). Los trabajadores que tienen relación con los procesos antes mencionados se encuentran relacionados en la siguiente tabla.

Trabajador del negocio	Justificación
Médico	Es el encargado de que se lleven a cabo los procesos del negocio: Examinar paciente, Verificar estado evolutivo, Realizar análisis, Registrar infección nosocomial, Registrar Ventilación, Registrar fallecimiento paciente, Realizar traslado paciente y Establecer alta paciente.
Jefa de servicio	Es la encargada del correcto funcionamiento del servicio de Neonatología, interviene en el proceso del negocio Confeccionar reportes estadísticos y Realizar Búsquedas para brindar esta información a la Dirección del hospital.

Tabla 2 Trabajadores del negocio

1.2.6 Diagrama de los casos de usos del negocio

“Un diagrama de casos de uso del negocio es un diagrama que representa gráficamente a los procesos del negocio y su interacción con los actores del negocio.” (Jacobson 2000).

A continuación se muestra el diagrama de los casos de usos del negocio en la figura 3:

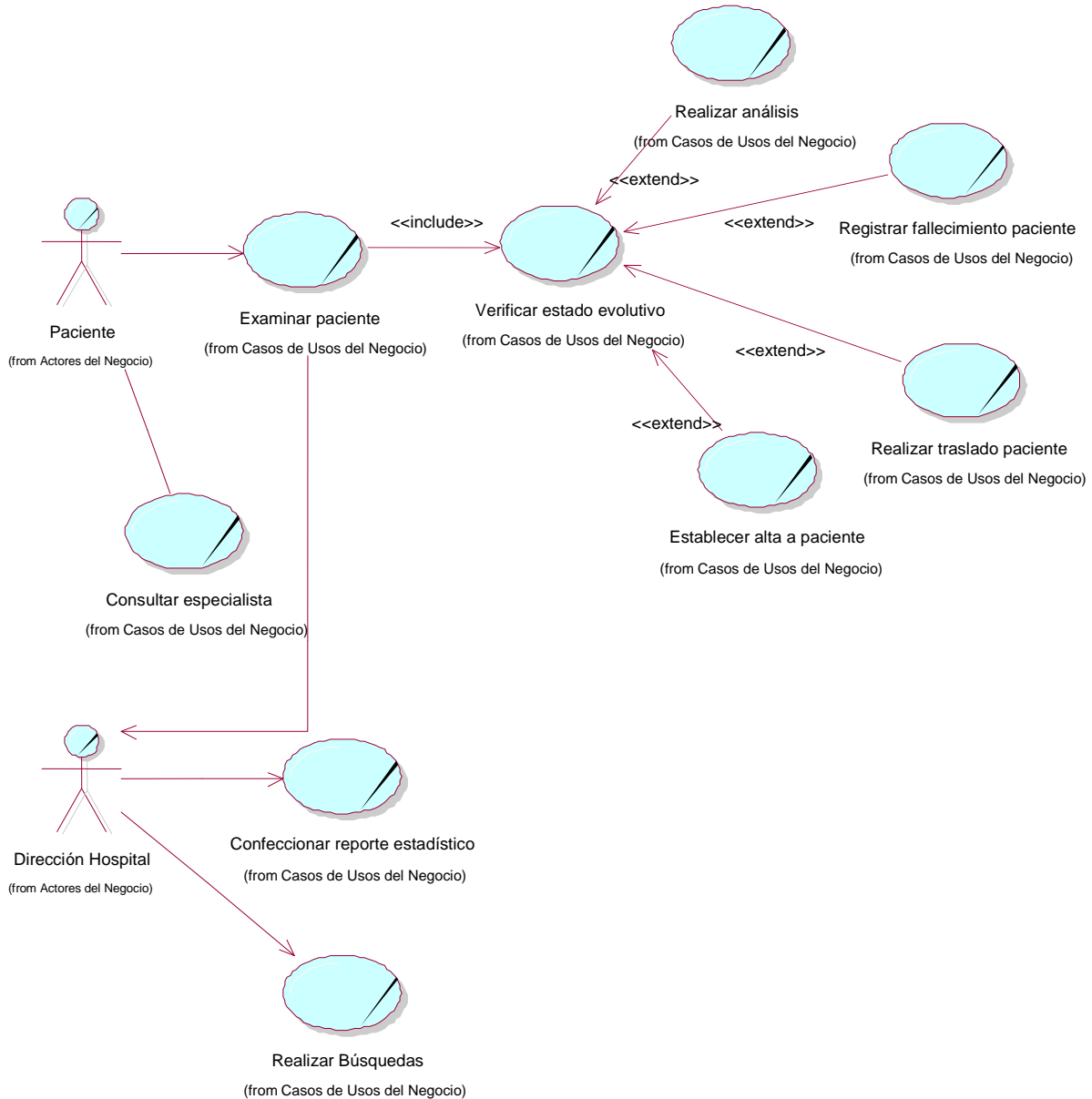


Figura 3 Diagrama de casos de usos del negocio

1.2.7 Casos de uso a expandir con su descripción textual, diagrama de Actividades y de clases del modelo de objeto

Los diagramas de actividades describen el flujo de trabajo asociado a los procesos de negocio. Muestran a través de las calles las responsabilidades de los trabajadores del negocio y a través del flujo de objetos cómo se utilizan las entidades del negocio.

Las entidades del negocio representan los objetos que los trabajadores del negocio utilizan o generan durante la realización de los procesos de negocio.

Fundamentación Teórica

Los diagramas de clases del modelo de objetos describen el modelo de objetos del negocio, muestran la participación de los trabajadores y entidades del negocio y la relación que se establece entre ellos.

A continuación se detallan 2 de los casos de usos del negocio más importantes, mediante su descripción textual, diagramas de actividades y de clase del modelo de objeto. El resto de las descripciones textuales así como los diagramas de actividades y del modelo de objeto puede ser consultado en el [Anexo 2](#).

Descripción textual del C.U.N (Examinar paciente)

Nombre Caso de Uso	Examinar paciente
Actores	Paciente(Inicia el caso de uso)
Propósito	Examinar los pacientes en busca de anomalías que determinen si es necesario ingresarlos o no.
Resumen	El caso de uso inicia cuando el paciente llega al servicio de neonatología del hospital pediátrico ya sea de la provincia Holguín o remitido de otra provincia Oriental con previo diagnóstico. El médico lo examina y determina si debe ser ingresado o no en la sala de neonatología. De no ser ingresado el médico confecciona un método con un tratamiento a seguir y envía el paciente a la casa y termina así el caso de uso. De ser ingresado el médico le confecciona una historia clínica, una ficha del paciente y una tarjeta de identificación, finalizando así el caso de uso.
FLUJO NORMAL DE EVENTOS	

Fundamentación Teórica

Acción del Actor	Respuesta del negocio
<p>1- El paciente llega al servicio de neonatología para ser examinado.</p> <p>4- El paciente regresa a su casa y así termina el caso de uso.</p>	<p>2- El médico examina al paciente en búsqueda de anomalías que justifiquen un ingreso o no.</p> <p>2.1- Si el paciente no necesita ser ingresado pasa al paso 3.</p> <p>2.2- Si el paciente necesita ser ingresado pasa al paso 5.</p> <p>3- El médico confecciona un método con el tratamiento a seguir por el paciente en la casa.</p> <p>5- El médico confecciona la historia clínica, la ficha del paciente y su tarjeta de identificación.</p> <p>6- El paciente es ingresado y así termina el caso de uso.</p>
Mejoras	La automatización de este proceso agilizará la confección de la ficha del paciente, el registro de ventilados y el registro de infecciones nosocomiales.
Prioridad	CUN Crítico

Tabla 3 Descripción textual del CUN Examinar paciente

Fundamentación Teórica

Diagrama de actividades del caso de uso examinar paciente
Las actividades de color azul serán las que se automatizarán.

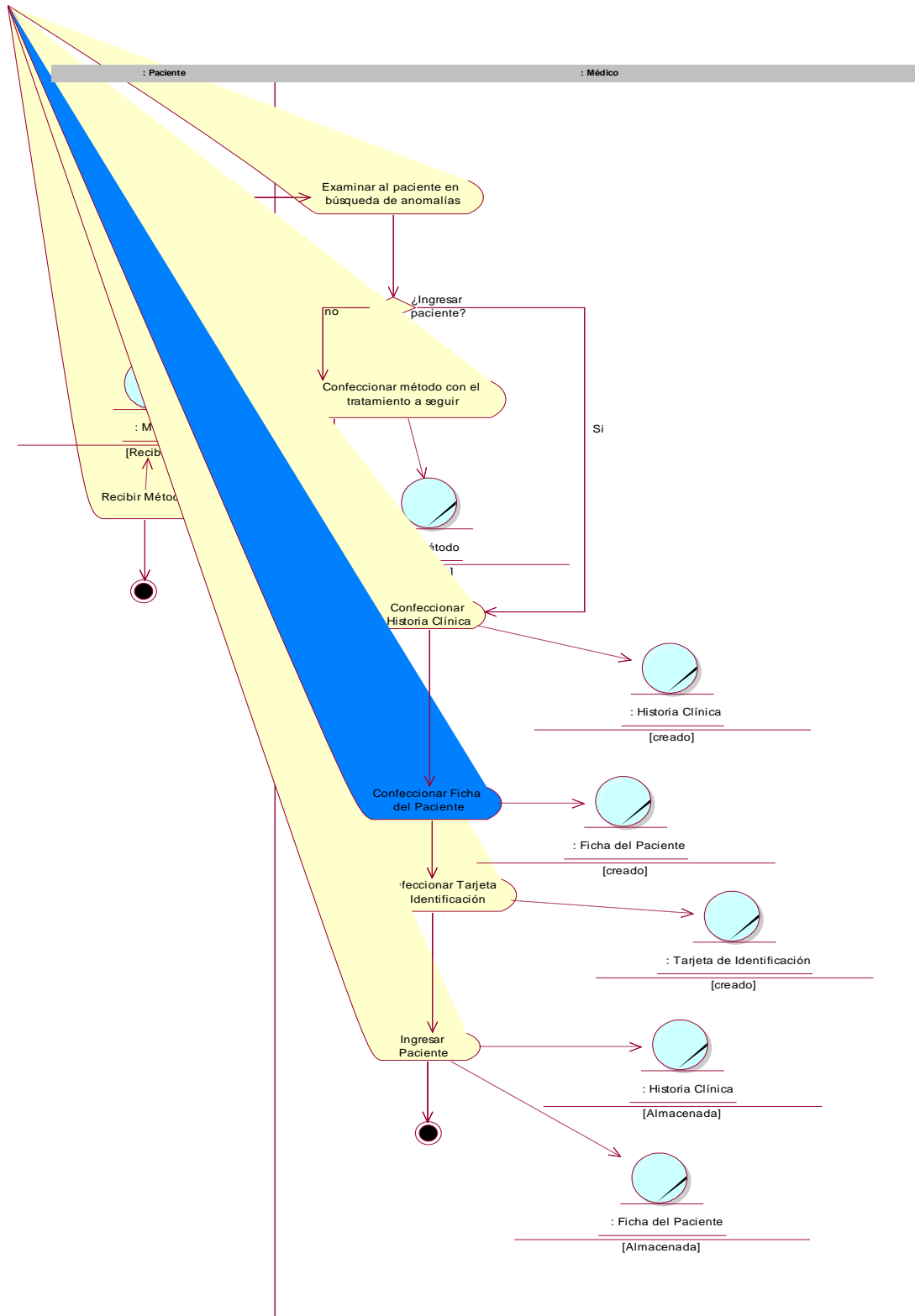


Figura 4 Diagrama de Actividades del CUN Examinar paciente

Diagrama de clases del modelo de objeto del CUN (Examinar Paciente)

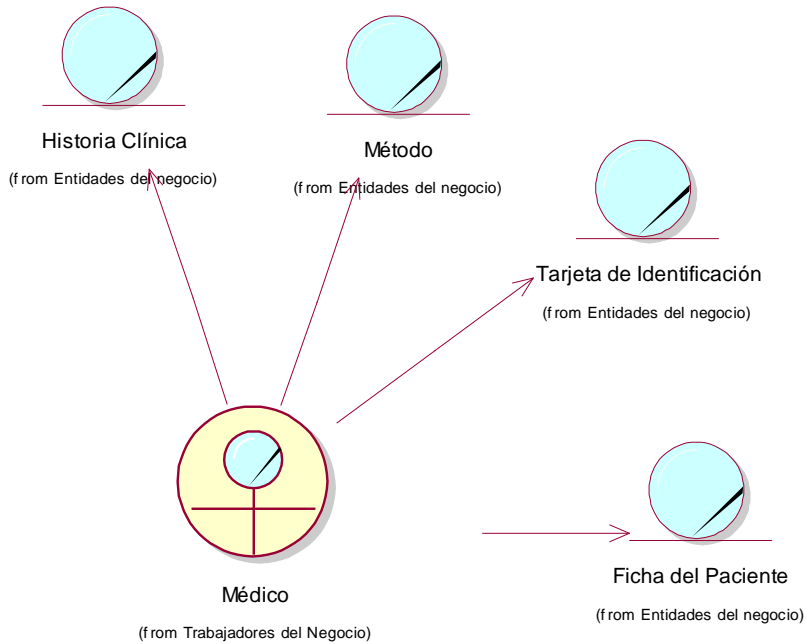


Figura 5 Diagrama de clases del modelo de objeto del CUN (Examinar Paciente)

Descripción textual DEL C.U.N (Confeccionar reporte estadístico)

Nombre Caso de Uso	Elaborar reporte estadístico
Actores	Dirección del Hospital(Inicia el caso de uso)
Propósito	Elaborar reportes estadísticos teniendo determinados criterios como son: total de pacientes ingresados (clínicos y quirúrgicos), el total de fallecidos tanto clínicos como

Fundamentación Teórica

	quirúrgicos, cantidad de graves al ingreso y ventilados además de una comparación teniendo en cuenta el año anterior.
Resumen	El caso de uso inicia cuando el Director del Hospital solicita a la Jefa de Servicio de la sala de neonatología confeccionar un reporte con algunos datos estadísticos. La Jefa de Servicio analiza el rango de tiempo pedido y cuáles son los datos estadísticos que se requieren en el informe. La Jefa de Servicio elabora el reporte estadístico y se lo entrega al Director del Hospital, así finaliza el caso de uso.
FLUJO NORMAL DE EVENTOS	
Acción del Actor	Respuesta del negocio
1- El Director del Hospital solicita a la Jefa de Servicio un reporte estadístico con datos específicos en un rango de tiempo.	2- La Jefa de Servicio recibe la solicitud del Director del Hospital. 3- La Jefa de Servicio analiza el rango de tiempo y los datos a tener en cuenta en la confección del reporte. 4- La Jefa de Servicio hace una búsqueda en la Ficha del paciente de todos aquellos datos comprendidos en el rango de tiempo exigido que le permita llevar la estadística. 4.1- La Jefa de Servicio encuentra los datos que le permiten confeccionar el reporte estadístico y el flujo de eventos continua en el paso 5.

Fundamentación Teórica

<p>7-El Director del Hospital recibe el reporte estadístico y así finaliza el caso de uso.</p> <p>8- El Director del Hospital es informado de la no posibilidad de confección del reporte pedido, terminando así el caso de uso.</p>	<p>4.2- La Jefa de Servicio no encuentra los datos que le permiten confeccionar el reporte estadístico, y el flujo de eventos continua en el paso 8.</p> <p>5- La Jefa de Servicio confecciona el reporte estadístico.</p> <p>6- La Jefa de Servicio entrega el reporte al Director del Hospital.</p>
Mejoras	La automatización del proceso de búsqueda de los datos así como la confección del reporte estadístico se hará con una mayor eficiencia.
Prioridad	CUN Secundario.

Figura 6 Descripción textual DEL C.U.N (Confeccionar reporte estadístico)

Diagrama de actividades del CUN Confeccionar reporte estadístico *Las actividades de color azul serán las que se automatizarán.*

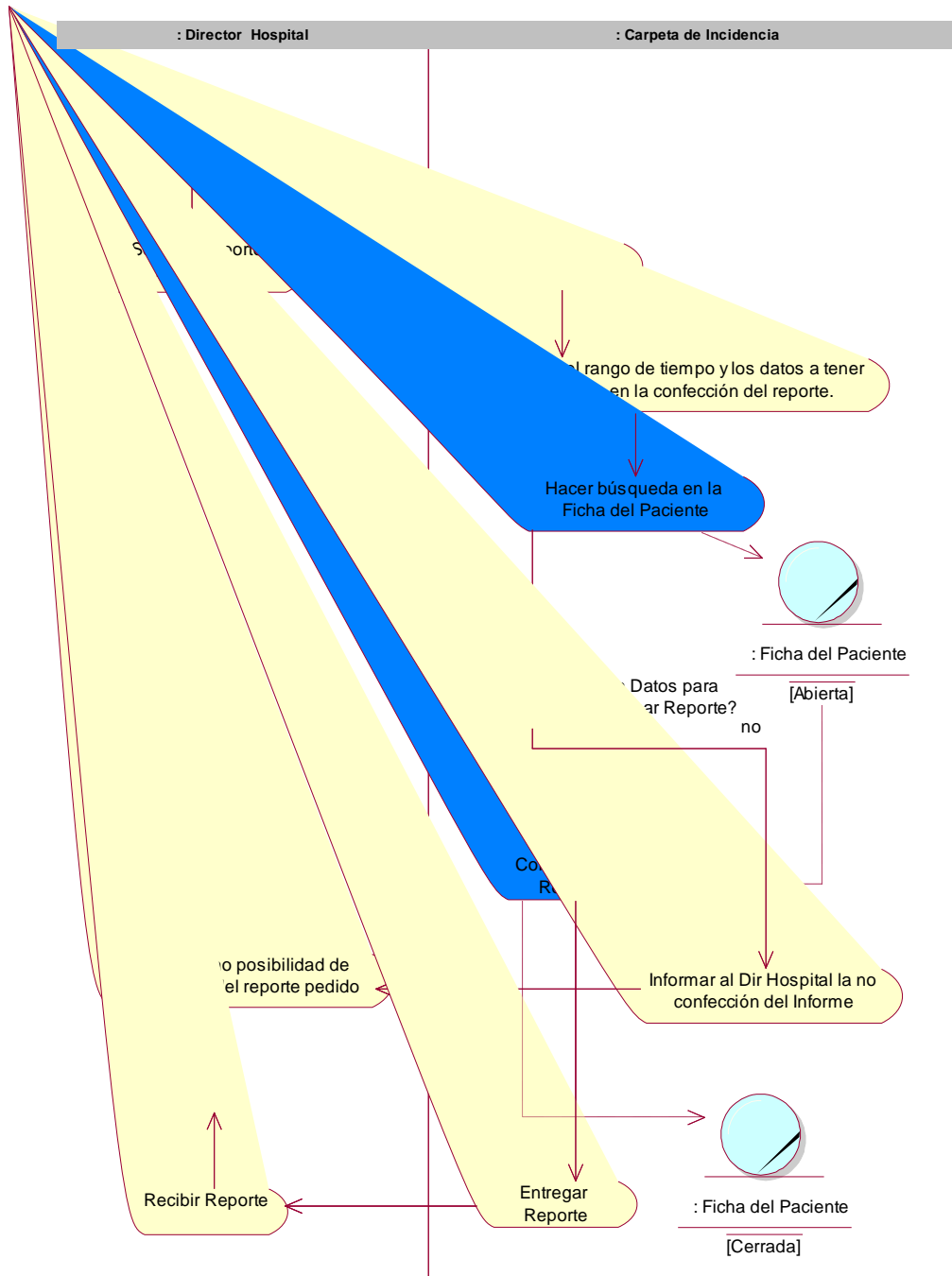


Figura 7 Diagrama de actividades del CUN Confeccionar reporte estadístico

Diagrama de clases del modelo de objeto del CUN (Confeccionar Reportes Estadísticos)

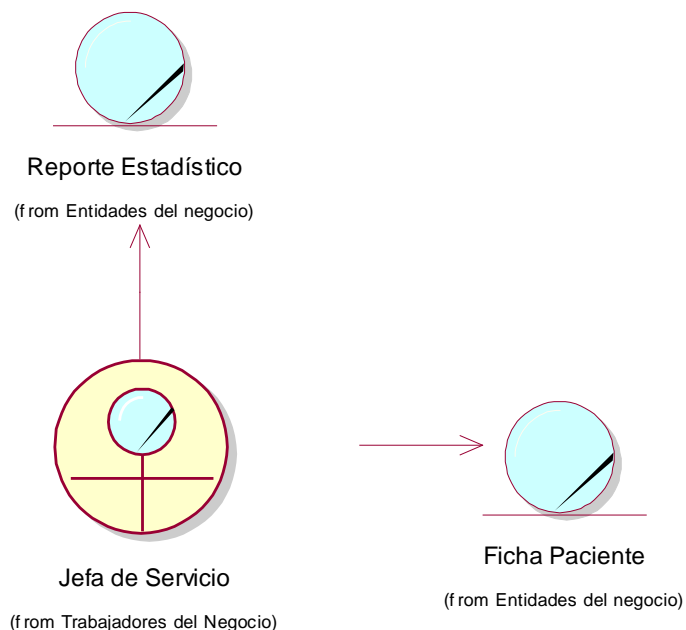


Figura 8 Diagrama de clases del modelo de objeto del CUN (Confeccionar Reportes Estadísticos).

1.3 Sistemas informáticos actuales vinculados al campo de acción

Al comenzar a desarrollarse el proyecto NeoGest no existía ningún sistema para servicios neonatales. Luego de una búsqueda exhaustiva en la Web de salud del país, solo se encontró un solo sistema de gestión de información para servicios neonatales, el cual lleva por nombre NEONAT y el mismo se encuentra en su primera versión, desarrollado en la Tunas. Este sistema inserta, busca, actualiza y extrae reportes referentes sólo a la información de los pacientes de la Sala de Neonatología intensiva del Hospital Docente Provincial Dr. Ernesto Guevara de la Serna adaptándose a las necesidades del mismo (Sánchez 2010). Por lo tanto se puede afirmar que actualmente no existe en la provincia Holguín ningún sistema de gestión de información de pacientes para servicios de Neonatología siendo NeoGest el primero en ponerse en práctica en el HPUH.

1.3.1 Comparación entre el sistema existente (NEONAT) y la propuesta de solución NeoGest

Comparando ambas propuestas de soluciones se puede llegar a la conclusión de que los dos sistemas gestionan la información de pacientes del servicio de Neonatología al cual responden de acuerdo a las necesidades específicas de cada uno, tratando información relevante que los caracteriza. No obstante NeoGest ofrece potencialidades mayores en cuanto a interfaz, rendimiento, reportes gráficos, búsquedas más dinámicas, tecnología empleada, gestiones agregadas que lo hacen un software más perdurable y flexible en el tiempo como es el caso de gestionar áreas de salud, patologías, factores de riesgos y tipo de infecciones, además de ser diseñado para un centro de referencia de Neonatología en las provincias orientales donde los servicios prestados poseen la mayor calidad posible.

1.4 Tendencias y tecnologías actuales

Para llevar a cabo la realización de una aplicación Web es primordial realizar un análisis de las tendencias y tecnologías actuales, A continuación se detallan aquellas que fueron estudiadas y se especifican las escogidas para el desarrollo de la presente investigación.

1.4.1 Arquitectura cliente –servidor

Consiste básicamente en un remitente de una solicitud (cliente) que realiza peticiones a otro programa (servidor) y este le da respuestas. Esta idea se puede aplicar a programas que se ejecutan en una sola computadora, es más ventajosa en un sistema operativo multiusuario distribuido a través de una red de computadoras. (Valle 2005)

Ventajas del uso de la arquitectura cliente–servidor

- Muestran información de forma sencilla a través de un hipervínculo.
- Es posible visualizar datos, documentos, multimedia por medio de los estándares de Internet.

Fundamentación Teórica

- Cuenta con centralización del control, ya que los accesos, recursos y la integridad de los datos son controlados por el servidor, lo que asegura que un programa cliente defectuoso o no autorizado, no pueda dañar el sistema.
- Esta centralización, también facilita la tarea de poner al día datos u otros recursos.
- Permite añadir nuevos nodos a la red ya sea en la parte de clientes, de servidor o ambos. Cualquier elemento puede ser aumentado y mejorado en cualquier instante (Escalabilidad). Como están distribuidas las responsabilidades y funciones entre varias máquinas independientes unas de otras, es posible reemplazar, reparar, actualizar, inclusive hasta trasladar un servidor, sin que los clientes se vean afectados por ese cambio.
- Las tecnologías están suficientemente desarrolladas y diseñadas para el paradigma de cliente –servidor que asegura seguridad en las transacciones, una interfaz amigable y facilidad en el empleo. (Álvarez 2009)

1.4.2 Las aplicaciones Web

Una aplicación Web es un conjunto de páginas Web enlazadas, en el que los usuarios acceden a un servidor Web a través de Internet o de una Intranet: mediante un navegador. Constituye una herramienta para divulgar, manipular y compartir la información; generándose dinámicamente una serie de páginas en un formato estándar, soportado por navegadores Web comunes como el Lenguaje de Marcas Hipertextuales (HTML, por sus siglas en inglés) o XHTML (Lenguaje Extensible de Marcado de Hipertexto, por sus siglas en inglés). En otras palabras, es una aplicación software que se codifica en un lenguaje que los navegadores Web soportan y ejecutan (Vegas 2009).

Ventajas del uso de una aplicación Web

Las aplicaciones Web ofrecen grandes ventajas que pueden ser aprovechadas por muchas organizaciones. Entre las mismas se pueden mencionar:

Fundamentación Teórica

- **Alta disponibilidad:** Se puede realizar consultas en cualquier parte del mundo donde se tenga acceso a Internet, a cualquier hora.
- **Compatibilidad multiplataforma:** Las aplicaciones Web tienen un camino mucho más sencillo para la compatibilidad multiplataforma que las aplicaciones software de escritorio. Varias tecnologías como Java, Flash, Páginas Activas del Servidor (ASP, por sus siglas en inglés) y Ajax permiten un desarrollo efectivo de programas que soportan los sistemas operativos principales.
- **Actualización:** Las aplicaciones basadas en Web están siempre actualizadas sin demandar que el usuario tome acciones pro-activas.
- **Menos requerimientos de memoria:** Al residir en los servidores del proveedor, las aplicaciones Web usan en la mayoría de los casos la memoria de las computadoras en que ellas se ejecutan, dejándose más espacio para múltiples aplicaciones del mismo o de otro tipo.
- **Menos bugs:** Con aplicaciones basadas en Web, todos utilizan la misma versión, y todos los bugs pueden ser corregidos tan pronto como son descubiertos. Esta es la razón por la cual las aplicaciones basadas en Web tienen menos bugs que el software de escritorio tradicional.
- **Datos online:** En las aplicaciones basadas en Web los datos pueden ser accedidos en cualquier instante (online), sin tener la necesidad el usuario de llevarlos físicamente, evitándose la pérdida o el robo.
- **Múltiples usuarios concurrentes:** Son utilizadas por múltiples usuarios al mismo tiempo, estos pueden ver e incluso editar un documento de manera conjunta.
- **Datos más seguros:** En el caso de la información que viaja por intranet las compañías que proveen aplicaciones Web brindan servicios de resguardo de datos. Los usuarios tienen menos riesgo de perder sus datos por ruptura del disco o por virus en la computadora. (Smith 2010)

1.4.3- Software libre

El software libre es un programa o secuencia de instrucciones, usada por un dispositivo de procesamiento digital de datos para llevar a cabo una tarea

específica o resolver un problema determinado, sobre el cual su dueño renuncia a la posibilidad de obtener utilidades por las licencias, patentes, o cualquier forma que adopte su derecho de propiedad sobre él (o sea, el software libre puede estar licenciado, o protegido por una patente autoral), esto permite que se utilice o transfiera sin pago alguno al licenciante, o a su creador. Otra característica es que se encuentra disponible el código fuente, lo que permite modificar el mismo sin ningún límite, y sin pago a quien lo inventó o lanzó al mercado. (Juárez, Herrera et al. 2002)

1.4.4 Tecnologías y herramientas para el desarrollo de aplicaciones Web

La tecnología Web es usada tanto por Intranet, como por Internet. Los sitios Web como las aplicaciones Web, se basan en esta tecnología para su funcionamiento. Un sitio Web es un conjunto de páginas Web que está dedicado específicamente a algún tema particular o propósito. Está escrito en HTML y puede ser estático. El contenido no cambia con frecuencia y para actualizar el contenido es necesario editar la página en algún software de edición de páginas Web. El sitio Web también puede ser dinámico, ya que puede tener cambios frecuentes en su contenido. La información de las páginas que conforman el sitio es generada en el servidor y mostrado el resultado al cliente, o sea, el navegador.

Una aplicación Web en una red de computadoras permite que siempre se esté accediendo a una información actualizada en cualquier parte de dicha red. Mediante el uso de la tecnología multimedia permite disponer de una interfaz gráfica muy bien lograda en cuanto a su diseño y una mayor y mejor interacción con el usuario. Además, mediante un sistema de seguridad se garantiza la integridad de la información y la eliminación de accesos indeseables y no permitidos.(Castillo 2004)

La Web ha ido evolucionando de lo tradicional a la creación de entornos RIA (Rich Internet Applications -Aplicaciones Enriquecidas de Internet). Existen muchas herramientas que permiten el desarrollo de aplicaciones enriquecidas de Internet, entre las que se encuentran las plataformas Flash y Flex de Adobe, AJAX, OpenLaszlo, Silverlight de Microsoft, JavaFX de Sun Microsystems y Bindows de MB Technologies.

Fundamentación Teórica

En la actualidad muchas son las tecnologías que se dedican a proveer soporte para el desarrollo de aplicaciones Web, cada propuesta cuenta con sus particularidades y bondades, de ahí que se debe tener bien definido qué se quiere lograr y si la herramienta seleccionada lo permite hacer. A continuación se describen algunas de las tecnologías más funcionales para el desarrollo de productos informáticos Web.

Ajax

Ajax, acrónimo de JavaScript asíncrono y XML, es una técnica de desarrollo Web para crear aplicaciones interactivas o Rich Internet Applications (RIA, por sus siglas en inglés), y el concepto es: cargar y renderizar una página, luego mantenerse en esa página mientras scripts y rutinas van al servidor buscándose, por detrás, los datos que son usados para actualizar la página y mostrar u ocultar porciones de la misma. Ajax no es una tecnología, sino la unión de varias tecnologías.

Normalmente las aplicaciones Web funcionan de esta forma: gran parte de las acciones del usuario en la interfaz desprenden un requerimiento http al servidor Web. El mismo efectúa un proceso que puede ser recuperar información o procesar números y le devuelve una página HTML al cliente con la respuesta a la petición efectuada. De esta forma el servidor ejecuta el pedido mientras que el usuario está en espera, prolongándose la misma en cada paso de la tarea. En cambio, una aplicación AJAX elimina la naturaleza “arrancar-frenar- arrancar-frenar” al introducir un intermediario (un motor AJAX) entre el usuario y el servidor. A primera vista puede parecer que sumar una capa a la aplicación la tornaría menos reactiva, pero es todo lo contrario. Al inicio de la sección, el navegador carga al motor, el cual es el responsable de renderizar la interfaz del usuario, lo que permite que la interacción del usuario-aplicación suceda asincrónicamente (independientemente de la comunicación con el servidor). Así la persona nunca esperará por la respuesta del servidor.

Mediante AJAX se pueden realizar múltiples peticiones asincrónicas al servidor, evitándose que el usuario detenga su trabajo, pudiéndose remplazar partes del

Fundamentación Teórica

documento sin tener que recargar toda la página y bloquear los eventos en el navegador.(Crane 2006; Chen 2007)

Cada acción de un usuario toma la forma de un llamado JavaScript al motor AJAX. Cualquier respuesta a una acción del usuario que no demande un viaje de vuelta al servidor es manejado por el motor. Si el motor necesita algo del servidor para responder (el envío de datos para procesar, cargar código adicional, o recuperar nuevos datos) hace esos pedidos asincrónicamente, usualmente usando el Lenguaje Extensible de Marcado (XML, por sus siglas en inglés), sin frenar la interacción del usuario con la aplicación.(Crane 2006)

Unas de las razones que hacen de AJAX una técnica popular son (Zervaas 2008):

- Basado en los estándares abiertos
- Usabilidad
- Válido en cualquier plataforma y navegador
- Beneficia las aplicaciones Web
- No es difícil su utilización
- Constituye una de las características que debe poseer una página Web para clasificarse en Web 2.0
- Es independiente del tipo de tecnología de servidor que se utilice.
- Mejora la estética de la Web

JSP

Páginas de Servidor de Java (JSP, por sus siglas en inglés) es una tecnología orientada a crear páginas Web que se ejecutan en variados servidores, de múltiples plataformas. Estas páginas, contienen código HTML/XML mezclado con etiquetas especiales para programar scripts de servidor en sintaxis Java. Antes de que sean funcionales los archivos, el motor JSP lleva a cabo una fase de traducción de esa interfaz en un servlet, implementado en un archivo class (Byte codes de Java). Esta fase de traducción se lleva a cabo, habitualmente, cuando se recibe la primera solicitud de la página .jsp, aunque existe la opción de precompilar el código para evitar ese tiempo de espera la primera vez que un cliente solicita la misma. (Alvarez 2002)

1.4.5- Servidores Web

Un servidor Web es un programa que está en ejecución, y en espera de solicitudes de forma continua. Consta de un intérprete de Protocolo de Transferencia de Hipertextos (HTTP, por sus siglas en inglés) el cual se mantiene a la espera de peticiones y responde con el contenido solicitado. El cliente se encarga de interpretar el código y lo exhibe en pantalla.

Apache

Hoy es común establecer los servidores Web dentro de la propia empresa, sin verse obligado a alojar los sitios en proveedores externos y que por lo general requieren un alto pago por sus servicios. Lo mismo es posible gracias a Apache, uno de los mejores servidores Web que existen en cuanto a estabilidad, confiabilidad y a la gran cantidad de personal voluntario que desarrolla todo lo relativo a esta plataforma. El mismo es un software libre, de código abierto para las plataformas Windows, MacOS X y UNIX (GNU, BSD, etc.), está estructurado en módulos, estos se pueden clasificar en tres categorías:

- **Módulos base:** módulo con las funciones elementales del Apache.
- **Módulos multiproceso:** manejan las peticiones, son los responsables de la unión con los puertos de la máquina, aceptándose las peticiones y envía a los hijos a atenderlas. Se han diseñado para cada uno de los sistemas operativos sobre los que se ejecuta el Apache.
- **Módulos adicionales:** cualquier otro módulo que le añada una funcionalidad al servidor. (Anónimo 2010)

Entre las muchas ventajas que cuenta Apache como servidor Web, se pueden citar:

- Es personalizable, la arquitectura modular de Apache permite construir un servidor hecho a la medida. Además, posibilita la implementación de los últimos y más nuevos protocolos
- En cuanto a la administración, los archivos de configuración de Apache están en ASCII, por lo que tiene un formato simple que puede ser editado tan solo con un editor de texto además de ser transferibles, posibilitándose la clonación efectiva de un servidor

- Puede ser administrado mediante líneas de comandos, lo que hace su administración remota muy conveniente
- Corre en una amplia variedad de sistemas operativos, incluyéndose varias versiones de UNIX, Windows9x/NT, MacOS (Sobre Power PC), entre otros
- Su soporte es provisto por “The Apache Group” o “La Fundación Apache”, así como compañías que ofrecen versiones pagadas de Apache.(Kabir 2002)

IIS

Servidor Información Internet (IIS, por sus siglas en inglés), es una serie de servicios para los ordenadores que funcionan con Windows. Convierte a un ordenador en un servidor de Internet o Intranet, ofreciéndose una infraestructura de gran fiabilidad, capacidad de manejo y escalabilidad para aplicaciones Web. Originalmente, era parte del Paquete de Opciones para Windows NT. Luego fue integrado en otros sistemas operativos de Microsoft destinados a ofrecer servicios, como Windows 2000 o Windows Server 2003. Windows XP Profesional incluye una versión limitada de IIS. Este servidor Web se basa en varios módulos, que le dan capacidad para procesar distintos tipos de páginas, por ejemplo: Microsoft incluye los de ASP y ASP.NET. También pueden ser incluidos los de otros fabricantes, como PHP o Perl. (Group 2010)

1.4.6 Lenguajes de Programación para la Web

Los lenguajes del lado del servidor más utilizados para el desarrollo de páginas dinámicas son el ASP, JAVA, PERL, PHP y Python.

JavaScript

Su creador es Brendan Eich, en sus inicios lo llamaron Mocha, luego LiveScript, pero no es hasta el 4 de diciembre de 1995 que formalmente es nombrado JavaScript. El organismo ECMA (European Computer Manufacturers Association, por sus siglas en inglés) estandarizó a la versión JavaScript 1.1 nombrándola ECMAScript de ahí que algunos programadores prefieren esta denominación para referirse al mismo.

Fundamentación Teórica

No es necesario compilar los programas para ejecutarlo, ya que es un lenguaje de programación interpretado (los programas escritos con JavaScript se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios). Se utiliza para crear páginas Web dinámicas. Múltiples lenguajes han servido de influencia para su desarrollo. Su sintaxis es similar a la de Java, con la bondad de ser más fácil de utilizar para personas con pocos conocimientos de programación.

En la actualidad, todos los navegadores interpretan el código JavaScript integrado dentro de las páginas Web. Su uso se basa, fundamentalmente, en la creación de efectos especiales en las páginas y la definición de interactividades con el usuario.

Las normas básicas que definen su sintaxis son las siguientes:

- No se tienen en cuenta los espacios en blanco y las nuevas líneas
- Se distinguen las mayúsculas y minúsculas
- No se define el tipo de las variables
- No es necesario terminar cada sentencia con el carácter de punto y coma
- Se pueden incluir comentarios

En resumen es un lenguaje bastante sencillo y pensado para realizar sistemas con rapidez. (Flanagan and Ferguson 2002)

PHP

PHP, acrónimo de Hypertext Preprocessor (inicialmente PHP Tools, o Personal Home Page Tools), es un lenguaje de programación Web usado, principalmente, en la interpretación del lado del servidor (server-side scripting). La mayoría de su sintaxis es similar a C, Java y Perl.

Cuenta con los entornos open source. En su desarrollo hay un proceso de colaboración que permite obtener, de forma gratuita, una enorme cantidad de recursos: el propio lenguaje, el servidor para ejecutarlo, manuales y tutoriales además de scripts que pueden ser descargar y usar inmediatamente. Incorpora el paradigma de la Programación Orientada a Objetos (POO). Tiene capacidad de conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad como MySQL, PostgreSQL, mSQL, Oracle, dbm, filepro, Hyperwave, Informix, InterBase y Sybase. (Manrique Chávez 2010)

Java

En un principio JAVA fue pensado como un lenguaje para programar equipos electrodomésticos y artefactos electrónicos. Su popularidad fue aumentando con el paso del tiempo llegándose a convertir en uno de los lenguajes de programación de propósito general más difundidos en el mundo. Posee características que lo hace un lenguaje de programación poderoso, sencillo, distribuido y confiable, capaz de ejecutarse en todo tipo de procesadores.(Darwin 2004; Morelli 2005)

Para abundar más a continuación se muestran un conjunto de ventajas que ofrece la utilización de este lenguaje.

- Se pueden escribir aplicaciones para intrarredes, aplicaciones cliente/servidor, aplicaciones distribuidas en redes locales y en Internet.
- Es fácil de aprender y está bien estructurado
- Las aplicaciones son fiables. Puede controlarse su seguridad frente al acceso a recursos del sistema y es capaz de gestionar permisos y criptografía. También, según Sun, la seguridad frente a virus a través de redes locales e Internet está garantizada. Aunque al igual que ha ocurrido con otras tecnologías y aplicaciones, se han descubierto, y posteriormente subsanado, “agujeros” en la seguridad de Java
- Es intrínsecamente orientado a objetos
- Multiplataforma
- Funciona perfectamente en red
- Aprovecha características de la mayoría de los lenguajes modernos evitando sus inconvenientes, en particular los del C++
- Tiene una gran funcionalidad gracias a sus librerías (clases)
- No tiene punteros manejables por el programador, aunque los maneja interna y transparentemente
- El manejo de la memoria no es un problema, la gestiona el propio lenguaje y no el programador
- Genera aplicaciones con pocos errores posibles por lo que se considera un lenguaje robusto

- Incorpora Multi-Threading (para permitir la ejecución de tareas concurrentes dentro de un mismo programa)

Existen en Internet una gran cantidad de códigos reutilizables, librerías, Plataformas o Entornos de desarrollo y frameworks basados en este lenguaje los cuales reducen considerablemente el tiempo de desarrollo de un sistema informático.

Plataforma Eclipse (IDE)

Eclipse: es un entorno de desarrollo integrado, de Código abierto y Multiplataforma. Mayoritariamente se utiliza para desarrollar lo que se conoce como "Aplicaciones de Cliente Enriquecido", opuesto a las aplicaciones "Cliente-liviano" basadas en navegadores. Es una potente y completa plataforma de Programación, desarrollo y compilación de elementos tan variados como sitios web, programas en C++ o aplicaciones Java. No es más que un entorno de desarrollo integrado (IDE) en el que encontrarás todas las herramientas y funciones necesarias para tu trabajo, recogidas además en una atractiva interfaz que lo hace fácil y agradable de usar.

La utilización de Eclipse tiene las siguientes ventajas:

- El (IDE) de Eclipse emplea módulos (en inglés plug-in) para proporcionar toda su funcionalidad al frente de la Plataforma de Cliente rico, a diferencia de otros entornos monolíticos donde las funcionalidades están todas incluidas, las necesite el usuario o no, Este posee un plug-in específico llamado ZK Studio para el desarrollo de aplicaciones Web el cual brinda un conjunto de prestaciones y funcionalidades que lo hacen ideal para este tipo de desarrollo
- La arquitectura plug-in permite escribir cualquier extensión deseada en el ambiente, como sería Gestión de la configuración. Se provee soporte para Java y CVS en el SDK de Eclipse. Y no tiene por qué ser usado únicamente para soportar otros Lenguajes de programación

- La definición que da el proyecto Eclipse acerca de su Software es: "una especie de herramienta universal - un IDE abierto y extensible para todo y nada en particular"
- El SDK de Eclipse incluye las herramientas de desarrollo de Java, ofreciendo un IDE con un compilador de Java interno y un modelo completo de los archivos fuente de Java. Esto permite técnicas avanzadas de refactorización y análisis de código

1.4.7 Frameworks Web basados en JAVA para el desarrollo de aplicaciones Web

“Los frameworks Web son una estructura de soporte definida en la cual otro proyecto de Software puede ser organizado y desarrollado. Típicamente, un framework puede incluir soporte de programas, bibliotecas y un lenguaje de scripting entre otros programas para ayudar a desarrollar y unir los diferentes componentes de un proyecto. Representan una arquitectura de software que modela las relaciones generales de las entidades del dominio”. (Inoto 2006)

Estos son el esqueleto sobre el cual varios objetos son integrados para una solución dada. Son diseñados con el intento de facilitar el desarrollo de software, permitiendo de esa manera a los diseñadores y programadores pasar más tiempo identificando requerimientos de software tratando con los detalles de bajo nivel para proveer un sistema funcional.

Fuera del ámbito de la Informática puede ser considerado como el conjunto de procesos y tecnologías usados para resolver un problema complejo. Aunque un framework no debe ser consciente de todos estos requerimientos sin tener que ser intrusivo con las aplicaciones que permite dentro de sí mismo, sumándole a esto la capacidad de extenderse sin prejuicios para diversificar la expresión del programa.

Posibilitan además el uso del patrón Model-View-Controller (MVC), dando la facilidad para lograr aplicaciones con mayor efectividad y dinamismo.

En fin, un framework de tipo Web no es más que una base de programación que atiende a sus descendientes de una forma estructural y/o en cascada permitiendo cualquier respuesta ante sus miembros, o secciones de una aplicación Web.

Fundamentación Teórica

Son numerosos los framework disponibles en Internet desarrollados en el lenguaje de programación JAVA para el desarrollo de sistemas Web, entre los más famosos por las ventajas que trae consigo su empleo se encuentran: Struts, Trapestry, Java Server Faces, Vaadin, ZK y Spring. (Inoto 2006)

Struts

Struts es un framework de código abierto destinado a crear aplicaciones Web en Java implementando el patrón MVC. “Realmente lo que provee es un conjunto de clases y TAG-LIBS que conforman el Controlador, la integración con el Modelo (o lógica de negocio) y facilitan la construcción de vistas”. (Greenspan 2001) Utilizando Struts nunca se llega a una página de la capa de presentación directamente. (Inoto 2006)

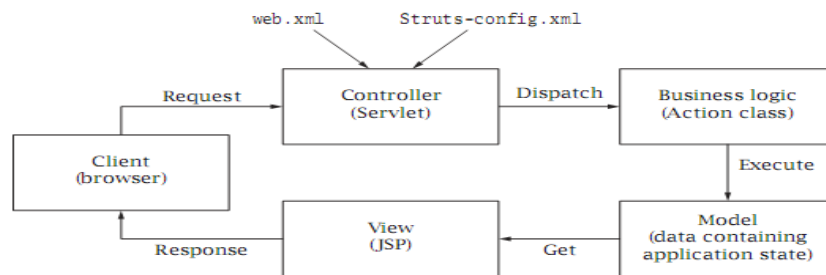


Figura 1.2 Modelo de una aplicación usando componentes de Struts

ZK

ZK es un framework para la construcción de aplicaciones Web basado en AJAX completamente desarrollado en JAVA. ZK permite la construcción de interfaces de usuario sin necesidad de usar ni conocer el lenguaje del lado del cliente JavaScript. (Chen 2007)

Proporciona que el uso de AJAX sea transparente para los desarrolladores de aplicaciones Web, aumentando considerablemente el rendimiento de los programadores y la disminución del tiempo de desarrollo. Los programadores diseñan las páginas de su aplicación con componentes XUL y XHTML, además pueden codificar su aplicación de una forma muy parecida al modelo de programación encontrado en las aplicaciones GUI (Graphical User Interface) de

escritorio.(Stauble 2010)

“ZK incluye un AJAX basado en un motor de control de eventos, conjuntos de componentes ricos mediante XUL y XHTML y un lenguaje de marcas denominado ZUML (ZK User Interface Markup Language)”. (Flanagan 2002).

Además de poseer interfaces similares a las aplicaciones de escritorio. ZK contiene XML basado en GUI, soporta un lenguaje de marcas denominado ZUML, así como XHTML, que permiten a los desarrolladores diseñar interfaces de usuario sin programación.

Este framework se ejecuta en código Java en el servidor, se puede escribir y manejar los eventos de la interfaz de usuario (GUI) con java. Todo el código Java, será implementado en zscript y en ZUML, o mediante un fichero java, que será ejecutado en el lado del servidor, y usa también expresiones EL para manipular los componentes y acceder a los datos.

ZK utiliza el acercamiento llamado centrado en el servidor para que la sincronización de componentes y el pipelining entre clientes y servidores se haga automáticamente por el motor, y los códigos de Ajax sean completamente transparentes para los desarrolladores de aplicaciones Web. Por lo tanto, los usuarios finales obtienen una interacción y respuesta similar a las de una aplicación de escritorio como se indica antes.

El uso del framework ZK en el desarrollo de sistemas Web provee las ventajas siguientes(Chen 2010; Chen 2011):

- Está desarrollado con el lenguaje JAVA, por lo que se puede hacer uso de todo el potencial de este lenguaje
- Se puede representar una aplicación Web con componentes XUL y XHTML, y manipularlos sobre eventos lanzados por la actividad del usuario similar a las aplicaciones de escritorio
- La sincronización del contenido de componentes y la cola de eventos son gestionados automáticamente por el motor de ZK
- Provee un marco sencillo, aunque no obligatorio, para el uso del patrón de

diseño MVC (Modelo Vista Controlador)

- Permite la internacionalización de los sistemas
- ZUML permite a los no expertos diseñar eficientemente interfaces de usuario
- Embeber script en Java ayuda al prototipado rápido y personalizaciones
- No hay requisitos de Ajax o JavaScript para los programadores
- Modelo basado en componentes conducido por eventos intuitivos
- Se integra perfectamente con CMS como por ejemplo LifeRay

No obstante, posee las siguientes desventajas:

- No hay soporte para XAML (Extensible Application Markup Language)
- No es apropiado para aplicaciones con alto grado de interacción, tales como los videojuegos de acción.

Modelo – Vista – Controlador (MVC) en ZK

En inglés Model-View-Controller (MVC), es un patrón de arquitectura de software que interviene en la separación de datos de una aplicación entre la interfaz de usuario y la lógica de control. Su usual aparición es en las aplicaciones de tipo Web, donde la vista es la página HTML y el código que provee de datos dinámicos a la página; el modelo es el Sistema de Gestión de Base de Datos y la Lógica de negocio; y el controlador es el responsable de recibir los eventos de entrada desde la vista.

El Patrón MVC está distribuido en las siguientes partes:

- *Modelo*: Esta es la representación específica de la información con la cual el sistema opera. La lógica de datos asegura la integridad de estos y permite derivar nuevos datos; por ejemplo, no permitiendo comprar un número de unidades negativo, calculando si hoy es el cumpleaños del usuario o los totales, impuestos o importes en un carrito de la compra.
- *Vista*: Este presenta el modelo en un formato adecuado para interactuar, usualmente la interfaz de usuario.
- *Controlador*: Este responde a eventos, usualmente acciones del usuario e

invoca cambios en el modelo y probablemente en la vista”.

ZK permite desarrollar plenamente el MVC en aplicaciones Web, haciendo que todo sistema a desarrollar sea lo más eficiente.

1.4.8 Sistema gestores de bases de datos

Se define una base de datos como un conjunto de información relacionada que se encuentra agrupada o estructurada en una computadora. Un sistema gestor de base de datos (SGBD) es una herramienta dedicada a servir de interfaz entre las bases de datos (BD), el usuario y las aplicaciones que la utilizan, admitiéndole crear y mantener sus BD, al proveerle una herramienta que sea capaz de transformar el nivel lógico que estos diseñaban en un conjunto de datos, representaciones y relaciones, y de traducirlo al nivel físico correspondiente, además se encargan de la privacidad, la integridad y la seguridad de los datos .(Álvarez 2007)

En la actualidad existen una gran cantidad de sistemas gestores de bases de datos en el mercado. MySQL y PostgreSQL son considerados los sistemas gestores de bases de datos libres más usados en el mundo.

PostgreSQL

Es un Sistema de Gestión de Bases de Datos Objeto-Relacionales (ORDBMS, por sus siglas en inglés). Se considera el sistema de BD de código abierto más avanzado del mundo. Permite crear nuevos tipos de datos, hacer herencias entre objetos. Es servidor de BD relacional libre, liberado bajo la licencia BSD.

En PostgreSQL se utilizan las vistas como tablas virtuales (una tabla que no existe físicamente en la BD, pero aparece al usuario como si existiera). Cuando se habla de una tabla base, se refiere a que realmente hay un equivalente almacenado para cada fila en la tabla en algún sitio del almacenamiento físico.

Es más completo que MySQL ya que permite métodos almacenados, restricciones de integridad, vistas, etc. PostgreSQL soporta operadores, funciones, métodos de acceso y tipos de datos definidos por el usuario, aproxima los datos a un modelo objeto-relacional, y es capaz de manejar complejas rutinas y reglas. Ejemplos de su avanzada funcionalidad son: consultas SQL declarativas, control de concurrencia multiversión, soporte multiusuario, transacciones, optimización de

Fundamentación Teórica

consultas, herencia, y arrays. PostgreSQL tiene soporte para lenguajes procedurales internos, incluyendo un lenguaje nativo denominado PL/pgSQL. (John Worsley 2001)

Características operacionales:

- Transacciones (Transactions)
- Disparadores (Triggers)
- Restricciones (Constraints)
- Replicación (Replication)
- Backup y Recuperación (Backup & Recovery)
- Reglas (Rules)
- Procedimientos Almacenados/Funciones (Stored Procedures/Functions)
- Integridad Referencial
- Outer Joins
- Logging.
- Extensivo y programable
- Orientado a Objetos
- Características sofisticadas de integridad de datos
- Tipos de datos y funciones definidos por el usuario
- Cliente/servidor.(Aldarias 2004)

MySQL

MySQL es un sistema de administración de bases de datos relacional (RDBMS, por sus siglas en inglés), muy fácil de instalar y administrar. Cuenta con una gran comunidad de usuarios al estar incluido en casi todas las distribuciones de Linux y es uno de los motores de base de datos más usados en Internet. El código fuente de MySQL se puede descargar y está accesible a cualquiera, por otra parte, usa la licencia GPL para aplicaciones no comerciales.

Es muy indicado para iniciarse en el mundo de las BD, puesto que dispone de una infinidad de utilidades, tutoriales y documentación. Se debe señalar que no tiene integridad referencial, por lo que puede llegar a tener datos inconsistentes, realiza los bloqueos de escritura al nivel de tabla, castigándose la concurrencia de

inserciones en la BD. Posee opciones de almacenamiento que permiten elegir velocidad en las operaciones o el mayor número de operaciones disponibles. Su popularidad como aplicación Web está muy ligada a PHP, que a menudo aparece en combinación con MySQL. (Greenspan and Bulger 2001; Greenspan 2001)

JDBC (Java Database Connectivity)

Java Database Connectivity, más conocida por sus siglas JDBC, es una API que permite la ejecución de operaciones sobre bases de datos desde el lenguaje de programación Java, independientemente del sistema operativo donde se ejecute o de la base de datos a la cual se accede, utilizando el dialecto SQL del modelo de base de datos que se utilice.

El API JDBC se presenta como una colección de interfaces Java y métodos de gestión de manejadores de conexión hacia cada modelo específico de base de datos. Un manejador de conexiones hacia un modelo de base de datos en particular es un conjunto de clases que implementan las interfaces Java y que utilizan los métodos de registro para declarar los tipos de localizadores a base de datos (URL) que pueden manejar.

Actualmente existe una tendencia al empleo de frameworks ORM (Object Relation Mapping) en el desarrollo de aplicaciones informáticas, los cuales gestionan las conexiones a la base de datos y sentencias SQL, y le permite a los programadores manejar los datos a través del paradigma de POO, disminuyendo el tiempo de desarrollo y manteniendo la portabilidad de una aplicación entre la mayoría de los motores de bases de datos existentes. No obstante el empleo de estos frameworks conlleva un incremento en el tiempo de ejecución, así como en el uso de memoria RAM (Bauer 2007).

Producto a las condiciones existentes en la infraestructura tecnológica de la entidad en la cual se implantará el sistema desarrollado en la presente investigación, se determinó que el uso del JDBC en el desarrollo del producto informático era la solución más factible, pues este influye positivamente en un mayor rendimiento de la aplicación.

1.4.9 Herramientas para Reportes

Los principales proyectos de reporting son:

Pentaho Reporting

Herramienta de reporting anteriormente llamada JFreeReports, adquirida e integrada en la suite Business Intelligence open source Pentaho. Para la generación de informes existen diferentes aplicaciones tanto a nivel de cliente (report wizard, report designer) como web (web ad-hoc reporting). (Díaz 2007)

Pentaho Reporting es la solución proporcionada por pentaho e integrada en su suite para el desarrollo de informes. Originariamente el proyecto se llamaba JFreeReports hasta la adquisición por parte de pentaho.

Existen tres productos diferentes con diferentes enfoques y dirigidos a diferentes tipos de usuarios.

Por una parte, existe un editor basado en eclipse con prestaciones profesionales de customización de informes denominado report redesigner destinado a desarrolladores de informes.

Por otro lado, existen un par de herramientas destinadas a usuarios con menos conocimientos técnicos basadas en wizards. Una como aplicación cliente llamada report design wizard, además otra vía web, llamada web ad-hoc reporting. El esquema de desarrollo de las aplicaciones de pentaho reporting sigue un esquema parecido al de BIRT (consultable a posteriori). El informe generado está encapsulado en un xml que el runtime transforma en el formato de visualización escogido.

Eclipse Birt

Herramienta especializada en reporting basada en la distribución eclipse con capacidad de ser embebida en proyectos J2EE de manera independiente. Es uno de los proyectos que existen en la fundación eclipse y está apoyado tanto por IBM como por Actuate.

Birt presenta un framework integrado con el IDE Eclipse para el desarrollo de informes y de un runtime para aplicaciones J2EE para visualizar estos informes.

Fundamentación Teórica

Como en las diferentes herramientas analizadas, los informes generados están encapsulados en xml. El flujo de creación de informes viene representado por el esquema de la figura 9:

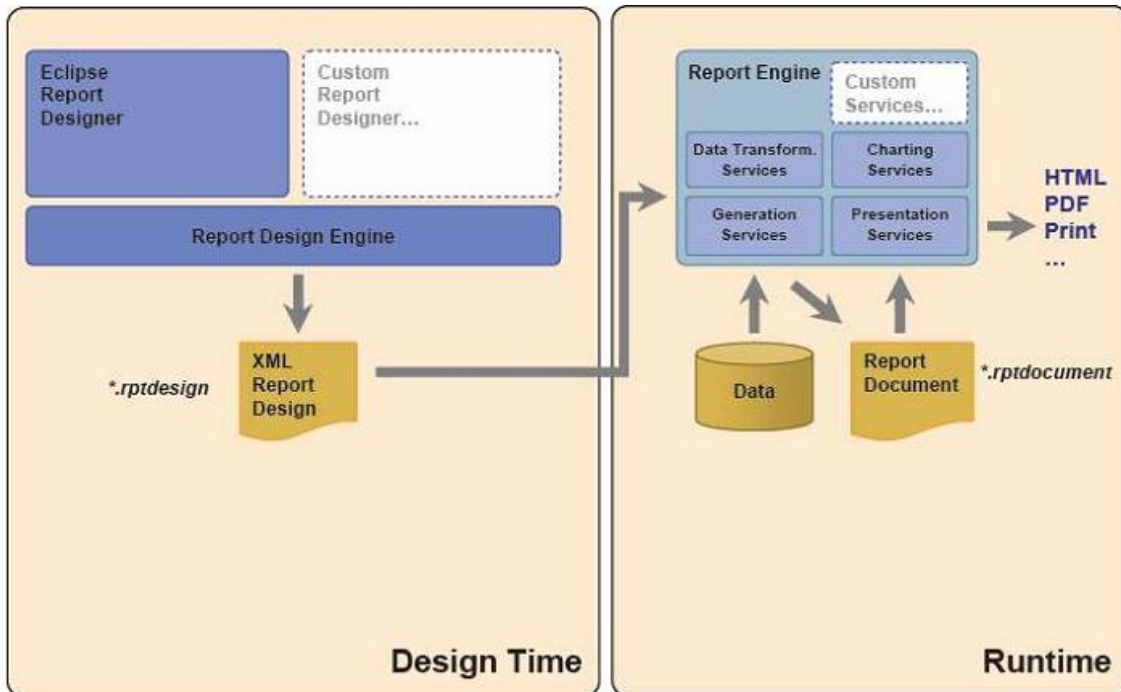


Figura 9 Esquema de flujo de creación de informes en BIRT

El visor de informes de Birt es una aplicación J2EE con capacidades de ser embebido en una aplicación existente asimismo customizable. Esta herramienta brinda un conjunto de prestaciones similares a JasperReport no obstante el manejo de los recursos y el rendimiento así como la capacidad de sus librerías o plugins integrados hacen que sea una herramienta menos optimizada que JasperReport. (Díaz 2007; Meyer 2007)

Jasper Reports

Esta herramienta es el producto que lleva más tiempo en el mercado. Cuenta con una herramienta llamada iReports para el desarrollo de informes.

Características IReport

- 100% escrito en Java y además Open Source y gratuito
- Maneja el 98% de las etiquetas de JasperReports
- Permite diseñar con sus propias herramientas: rectángulos, líneas, elipses, campos de los textfields, cartas, y subreportes

Fundamentación Teórica

- Soporta internacionalización nativamente
- Browser de la estructura del documento
- Recopilador y exportador integrados
- Soporta JDBC
- Soporta JavaBeans como orígenes de datos (éstos deben implementar la interfaz JRDataSource)
- Incluye Wizard's (asistentes) para crear automáticamente informes
- Tiene asistentes para generar los subreportes
- Tiene asistentes para las plantillas
- Facilidad de instalación

Este proyecto se inició en 2001 como proyecto personal de Teodor Danciu. Posteriormente es adquirido por JasperReports para dar respuesta a las necesidades de reporting y para formar parte de la suite propia conjuntamente con Jasper Server.

El esquema del mecanismo de generación de informes es como se muestra en la figura 10:

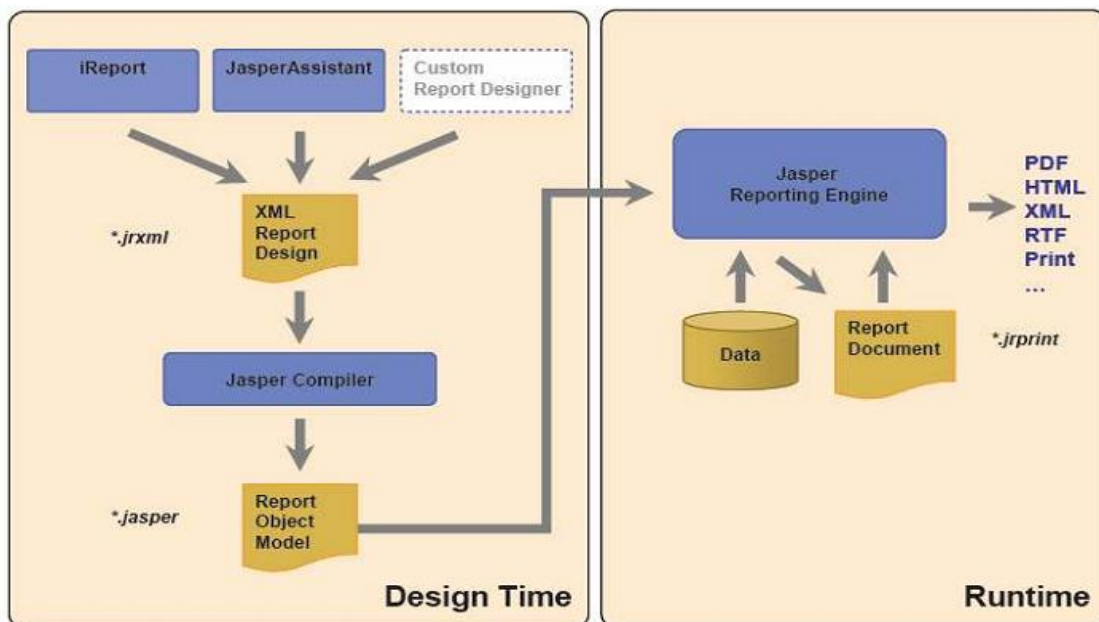


Figura 10 Esquema del mecanismo de generación de informes en JasperReports

Fundamentación Teórica

Cosa que nos indica dos detalles importantes respecto a las otras soluciones:

- Para poder visualizar informes generados con iReport estos deben ser compilados
- En el caso de necesitar modificar un informe existente es necesario recompilar el informe, es decir, es necesario pasar necesariamente por la fase intermedia de compilación

El visor de informes está realizado en swing, al igual que el editor de informes.

JasperReports nos ofrece una plataforma de desarrollo muy avanzada y que permite crear informes de una manera sencilla y sin necesidad de mucho conocimiento técnico. Además, ofrece un rendimiento mayor a la hora de ejecutar los informes ya que maneja mejor los recursos de la máquina, se integra de manera satisfactoria con Spring, permite múltiples data sources.

Por último, Jasper usa un output intermedio que maneja mucho mejor diferentes formatos de salida (pdf, excel, html, rtf, txt csv,...). Permite conectividad de fuente de datos JDBC, XML, EJB, Hibernate. En cuanto a gráficos posee varias formas o tipos así como formatos de salidas (PNG, JPG, PDF). (Díaz 2007; Meyer 2007)

Por todo lo anterior planteado y por las ventajas que brinda utilizar JasperReport e IReport como su herramienta de trabajo y en gran medida por su rendimiento que es una necesidad del proyecto, se escogen ambas para desarrollar la aplicación.

Fundamentado en los argumentos antes expuestos, se ha escogido para el desarrollo de la aplicación Web el framework ZK para el diseño e implementación por su facilidad y rapidez en cuanto a la realización del sistema Web propuesto. Todos sus componentes y sus respectivas funciones intervienen en cada interfaz o en cada código heredado de Ajax, Entorno de desarrollo Eclipse y lenguaje de programación utilizado por el mismo Java, como sistema gestor de base de datos al PostgreSQL, como servidor Web al Apache Tomcat, como herramientas para reportes JasperReport e IReport respectivamente.

1.5 Metodologías de ingeniería de software

Es el conjunto de filosofías, fases, procedimientos, técnicas, reglas, herramientas y un soporte documental que ayuda a los desarrolladores a realizar aplicaciones informáticas con el fin de hacerlo más predecible y eficiente. (Pressman 2002) Se adecuan al desarrollo de la mayoría de las aplicaciones. Las metodologías tradicionales tienden a ser burocráticas al llevar una documentación exhaustiva de todo el proyecto y cumplir cabalmente con un plan del proyecto, por lo que cuentan con poca popularidad. Ante el descontento de un grupo de usuarios, nacen los métodos ágiles, los cuales tienen como objetivo fundamental minimizar la documentación de desarrollo empleándola, fundamentalmente, como vehículo de comprensión de problemas dentro del grupo de trabajo y de comunicación con los usuarios. Se basan en dos aspectos principales, el retrasar las decisiones y la planificación adaptativa.

Estas metodologías revelan que la capacidad de respuesta ante un cambio importa más que el seguimiento de un plan estricto. Retrasar las decisiones será ventajoso tanto para el cliente como para la empresa, por reducir el número de decisiones de alta inversión, reducir el número de cambios necesarios en el proyecto, así como el costo de cambio.

Una de las principales metodologías tradicionales es RUP y ágil, Iconix. (Mendoza 2004)

RUP

Rational Unified Process (RUP, por sus siglas en inglés). Fue desarrollado por Rational Software, y está integrado con toda la suite Rational de herramientas. Es un proceso de desarrollo de software orientado a objetos. Utiliza UML (estándar del Object Management Group) como lenguaje de notación. (Rumbaugh 2002)

Es un proceso iterativo e incremental, divide el trabajo en partes más pequeñas. Específicamente en 4 fases, las cuales son:

- Concepción: para determinar la visión del proyecto
- Elaboración: con el objetivo de establecer la arquitectura óptima
- Construcción: con la meta de lograr la capacidad operacional inicial

Fundamentación Teórica

- Transición: para alcanzar la liberación del proyecto

A continuación se muestra la figura 11 que ilustra el flujo de trabajo de RUP:

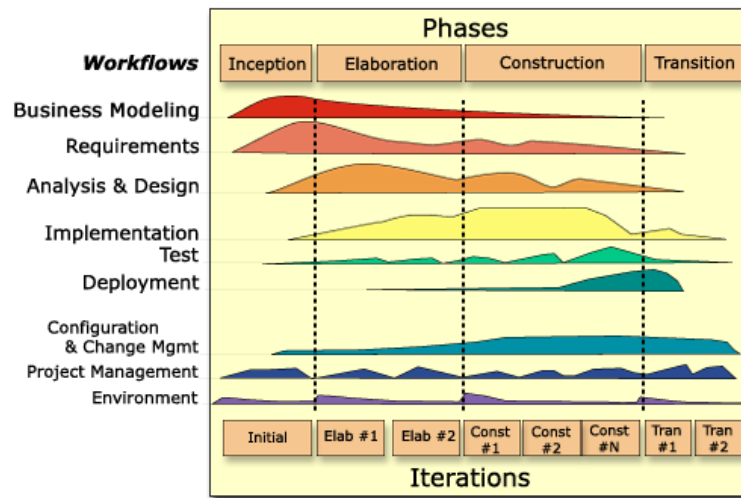


Figura 11 Flujo de trabajo de RUP

Los principales flujos de trabajo de la metodología RUP son:

- Modelado del negocio: describe los procesos de negocio, identificándose quiénes participan y las actividades que requieren automatización
- Requerimientos: define qué es lo que el sistema debe hacer, para lo cual se identifican las funcionalidades requeridas y las restricciones que se imponen
- Análisis y diseño: describe cómo el sistema será realizado a partir de la funcionalidad prevista y las restricciones impuestas (requerimientos), por lo que indica con precisión lo que se debe programar
- Implementación: define cómo se organizan las clases y objetos en componentes, cuáles nodos se utilizarán y la ubicación en ellos de los componentes y la estructura de capas de la aplicación
- Prueba (Testeo): busca los defectos a lo largo del ciclo de vida
- Instalación: produce el lanzamiento del producto realizándose actividades de empaque, instalación y asistencia a los usuarios finales del software
- Administración del proyecto: involucra actividades con las que se busca producir un producto que satisfaga las necesidades de los clientes

- Administración de configuración y cambios: describe cómo controlar los elementos producidos por todos los integrantes del equipo de proyecto en cuanto a: utilización/actualización concurrente de elementos, control de versiones, etc. (Jacobson 2000)

Lenguaje Unificado de Modelado

El Lenguaje Unificado de Modelado (UML, por sus siglas en inglés), es un lenguaje de especificación, visualización, construcción y documentación de propósito general, aunque especializado en sistemas software. El Lenguaje Unificado de Modelación proporciona ventajas en la representación del ciclo de vida de un software y de los artefactos específicos del Proceso Unificado de Desarrollo de Software. No se necesitan conocimientos profundos de Ingeniería de Software por parte de los clientes para que comprendan lo que los desarrolladores muestran (Leyton 2004).

UML es un lenguaje para especificar, construir, visualizar y documentar los artefactos de un sistema de software orientado a objetos. No define un proceso concreto que determine las fases de un sistema, las empresas pueden utilizarlo como lenguaje para definir sus propios procesos y lo único que tendrán en común con otras organizaciones que lo utilicen serán los tipos de diagramas. (Carrasquel Aguilera 2007)

UML es sólo el lenguaje de modelado. No define un proceso que diga cómo utilizarlo para desarrollar sistemas. Para UML una solución Web es un software diseñado para automatizar procesos cuyo resultado se entrega a través de Internet o una Intranet.

Iconix

Fue elaborado por Doug Rosemberg y Kerdall Scott. Se define como un “proceso” de desarrollo de software práctico. Unifica un conjunto de métodos orientados a objetos con el objetivo de abarcar todo el ciclo de vida de un proyecto. Presenta claramente las actividades de cada etapa y exhibe una secuencia de pasos que deben ser seguidos.

Iconix posee las siguientes características:

Fundamentación Teórica

- Es iterativo e incremental, pues varias iteraciones ocurren entre el desarrollo del modelo del dominio y la identificación de los casos de uso. El modelo estático es incrementalmente refinado por los modelos dinámicos
- Centrado en datos: se descompone en fronteras de datos
- Basado en escenarios que descomponen los casos de uso
- Cada paso está referenciado por algún requisito al tener la capacidad de seguir una relación entre los diferentes “artefactos del software” producidos. Seguimiento significa la posibilidad de seguir los objetos paso a paso como el análisis dentro del diseño
- Ofrece un uso dinámico del UML al no exigir la utilización de todos los diagramas del UML

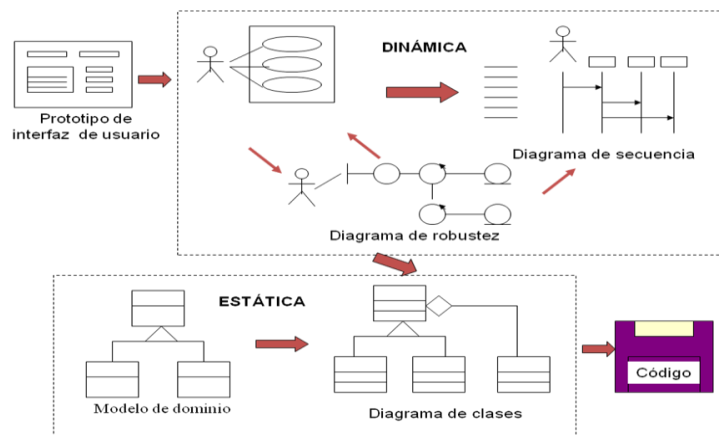


Figura 1.2. Esquema de trabajo Iconix

La participación y el compromiso de los usuarios finales es uno de los pilares fundamentales de las metodologías ágiles, que permite verificar la completitud y el cumplimiento de los requisitos. Esto se logra con la participación de los usuarios en la prototipación temprana, en la descripción de los casos de uso y en las pruebas del sistema.

La principal característica de esta metodología que lo diferencia de otras metodologías es su uso de análisis de robustez, el cual permite el puente entre el análisis y el diseño al reducir la ambigüedad en las descripciones de caso de uso y que estos sean mucho más fáciles de diseñar, evaluar y calcular. Cuenta con amplias bondades en los servicios de negocios. Las soluciones de negocios se enfocan en los servicios en áreas primarias (la experiencia del usuario,

Fundamentación Teórica

funcionalidad comercial, e infraestructura), con la planeación y la estrategia reforzando cada área. Esta especialización en las áreas primarias tributa al éxito final de los productos que se entregan a los clientes

Se adapta a los requerimientos ya que está menos orientada al documento y más centrada en el código fuente, exigiéndose menor documentación y ahorrándose tiempo. No existe una enorme plantilla de documentación para los casos de uso, sino que estos hablan de qué están haciendo los usuarios en las interfaces finales. Es conducida por casos de uso pero no incorpora tantos artefactos UML, sino que utiliza un subconjunto mínimo pero suficiente de estos para realizar un buen trabajo de Ingeniería de Software en poco tiempo.(Rosenberg 2005)

Comparación entre RUP e Iconix

A continuación se muestra la tabla 4 la cual establece una comparación entre las metodologías RUP y Iconix, en un grupo de aspectos, que ayudan a la toma de decisiones en términos de qué metodología usar.

Características	RUP	Iconix
Tipo de Metodología	Metodología tradicional (formal)	Metodología ágil
Orientadas	A los procesos.	A las personas.
Documentación	Lleva una documentación exhaustiva de todo el proyecto.	Prioriza crear un producto software funcional por encima de la documentación exhaustiva, pero sin prescindir de ella.
Plan de proyecto	Se centra en la planificación.	Se enfoca en la capacidad de respuesta ante un cambio por encima del seguimiento estricto de un plan. Cuenta con planificación adaptativa (tomar decisiones a lo largo del proyecto).
Tamaño del Proceso	Medio/Extenso	Pequeño / Medio

Fundamentación Teórica

Tamaño del Equipo	Medio/Extenso	Pequeño / Medio
Complejidad del Problema	Medio / Alto	Medio / Alto

Tabla 4 Comparación entre las metodologías de modelación RUP e Iconix

Por todo lo argumentado anteriormente se escoge como metodología para la modelación del sistema a RUP, ya que el cliente solicita una amplia documentación del sistema, así como una estricta organización y planificación.

1.6- Conclusiones del capítulo

De lo abordado en este Capítulo se puede concluir lo siguiente:

1. En este capítulo se realizó un estudio acerca de los procesos relacionados con la gestión de información de los pacientes de la sala de neonatología del HPUH permitiendo conocer las principales actividades del negocio y así comprender mejor su funcionamiento, favoreciendo el posterior desarrollo de los requisitos funcionales en el sistema.
2. El proceso de gestión de información referente a los pacientes de la sala de Neonatología, presenta deficiencias relacionadas con la velocidad, seguridad, actualización y manipulación.
3. Se decidió desarrollar una aplicación Web soportada en el framework ZK, el sistema de gestión de BD PostgreSQL y se integrará en un servidor Web Apache Tomcat además de JasperReport e IReport para el módulo de reportes.
4. La utilización de tecnologías software libre presenta ventajas vinculadas a la reducción de costos, independencia del proveedor y corrección de errores.
5. La metodología RUP contribuye a elaborar aplicaciones funcionales, posee una gran cantidad de artefactos, documentación y organización, características deseadas para el ciclo de vida del desarrollo del Sistema para la Gestión de Información de los pacientes de la sala de Neonatología del HPUH Octavio de la Concepción y la Pedraja, además cumple con los requerimientos de los clientes, características que la hacen adecuada para el modelado del proyecto objeto de la presente investigación.

CAPÍTULO 2 Descripción y elaboración de la solución propuesta

Introducción

El presente capítulo tiene como objetivo abordar los temas relacionados con el análisis y construcción del sistema propuesto de acuerdo con las buenas prácticas en la construcción de aplicaciones Web.

Se realiza la valoración de sostenibilidad de la aplicación Web según las dimensiones administrativa, socio-humanista, ambiental y tecnológica. Para la estimación de los costos del producto en la dimensión administrativa se empleó el Modelo Constructivo de Costo II (COCOMO II, por sus siglas en inglés).

2.1 Captura de Requerimientos

2.1.1 Requisitos funcionales

El paso desde la determinación de las necesidades del cliente hasta la implementación no es trivial. En primer lugar, las necesidades del cliente no son fáciles de discernir. Esto obliga a que se debe tener algún modo de capturar las necesidades del usuario de forma que puedan comunicarse fácilmente los requerimientos del sistema en toda su magnitud. (Jacobson 2000)

“Captura de requisitos es el acto de descubrimientos. Es el proceso de averiguar normalmente en circunstancias difíciles, lo que se debe construir. De hecho, es tan difícil que todavía no es poco común para los equipos de proyecto el comenzar a escribir código (lo cual es bastante fácil) antes de que hayan firmado simplemente lo que se supone que debe hacer el código (lo cual es difícil de determinar)” (Jacobson 2000).

Los requisitos son las condiciones o capacidades que un sistema debe satisfacer. El propósito de la gestión de requisitos es establecer un entendimiento común, entre el cliente y el desarrollador de software, de los requisitos que serán abordados por el proyecto de software. Los mismos constituyen la determinación (objetiva y clara) de las funcionalidades que el sistema debe ser capaz de hacer de forma visible o no para el usuario y especifican el comportamiento de entrada y salida del sistema.

Descripción y Elaboración de la Solución Propuesta

A continuación se listan los requisitos funcionales del sistema:

Paquete de Seguridad

R1: Autenticar Usuario.

R2: Validar usuarios.

R3: Gestionar usuarios del sistema.

R3.1 Añadir usuario al sistema.

R3.2 Modificar usuario del sistema

R3.3 Eliminar Usuario del sistema

R4: Salvas Cruzadas o Backups automáticos de la Base de Datos.

R5: Gestionar Procedencias.

R6: Gestionar Factores de Riesgo.

R7: Gestionar Tipo de Infecciones.

R8: Gestionar Patologías.

Paquete Paciente

R9: Actualizar ficha del paciente.

R9.1 Insertar ficha del paciente.

R9.2: Modificar ficha del paciente.

R9.3 Eliminar ficha de pacientes (solo de ser necesario).

R10 Mostrar Ficha del Paciente.

R11 Buscar paciente.

R12: Realizar Búsquedas por Criterios:

R12.1: Por nombre.

R12.2: Por primer apellido.

R12.3: Por segundo apellido

R12.4: Por Reporte (estado de ingreso).

R12.5: Por Edad gestacional

R12.6: Por patología

R12.7: Por No de historia clínica.

R12.8: Por fecha de ingreso.

R12.9: Buscar por ventilado.

R12.10: Buscar por operado.

Descripción y Elaboración de la Solución Propuesta

R12.11: Buscar por contrareferencia.

R12.12: Buscar por fallecido.

R12.13: Buscar por procedencia.

R12.14: Buscar por rescatado

R13 Mostrar Reportes de listado de paciente en varios formatos.

R14: Visualizar reportes estadísticos

R14.1 Reporte del total de pacientes ingresados (clínicos y quirúrgicos).

R14.2 Reporte del total de fallecidos tanto clínicos como quirúrgicos

R14.3 Reporte de la cantidad de graves al ingreso

R14.4 Reporte de la cantidad de ventilados.

R15 Mostrar Gráficos Datos comparativos por rangos de fecha.

R16 Extraer Gráficos de Datos Comparativos en formatos.

2.1.2 Definición de los requisitos no funcionales

Son propiedades o cualidades que el producto debe tener. Estas propiedades son las características que hacen al producto atractivo, usable, rápido o confiable. En muchos casos los requerimientos no funcionales son fundamentales en el éxito del producto. Describen atributos del sistema o del ambiente del sistema. (Jacobson 2000)

Los requisitos no funcionales explican las características que de una u otra forma puedan limitar el sistema. Describen atributos del sistema o del ambiente en que va a ser utilizado. También añaden funcionalidad al producto, pues hacen que un producto sea fácil de usar, seguro, o interactivo.

A continuación se describen los requerimientos no funcionales que caracterizan al sistema informático:

Apariencia o interfaz externa

El diseño de aplicación debe ser orientado a simular el entorno de trabajo del cliente para que el usuario se sienta cómodo e identificado con la aplicación.

El sistema debe ser asequible para todos los usuarios, ya que muchos no son expertos en computación. Los colores deben ser relajantes a la vista del usuario.

El sistema debe tener una interfaz cómoda, facilitando la navegación y evitando la utilización en exceso de imágenes, debe ser fácil de usar a través de

Descripción y Elaboración de la Solución Propuesta

elementos visibles en su diseño que identifiquen cada una de sus acciones, de no poderse ejecutar una acción por un usuario, visualizar un mensaje de error que muestre el motivo.

Usabilidad

El sistema debe apoyarse grandemente en facilidades de selección de listas desplegables predeterminadas, así como también en búsquedas automatizadas. Además, debe de estar disponible en todo momento del día y accesible desde cualquier lugar del HPUH con conexión de red.

Portabilidad

El sistema se encuentra implementado utilizando PostgreSql y el framework ZK que utiliza lenguaje de programación Java ambos son software multiplataforma por lo que pueden ser utilizados en distintas plataformas sin ningún tipo de problemas.

Seguridad

Restringir el acceso de los usuarios a las funcionalidades de acuerdo a los roles jugados por los actores del sistema.

El sistema garantizará que los usuarios puedan acceder a la información que le es permitida de acuerdo a los roles definidos para cada uno, es decir el nivel de acceso debe ser restringido. La información debe ser actualizada por el personal autorizado.

Identificar el usuario antes de ejecutar cualquier acción y verificar que tiene permiso para llevarla a cabo.

El sistema debe contar con métodos de seguridad ante la pérdida de la información causada por ruptura del servidor u otros accidentes.

Confiabilidad

La información almacenada en el sistema debe ser estrictamente confiable correspondiéndose con el resultado de los procesos de gestión de información en la sala, para evitar que la información almacenada o manipulada posea errores.

Ayuda y documentación en línea

Descripción y Elaboración de la Solución Propuesta

El sistema debe constar con la documentación necesaria acerca de cómo utilizarlo.

Software

Cliente

Se debe disponer de un navegador Web instalado en las máquinas clientes de la aplicación, con soporte para JavaScript y que implemente las características necesarias para el uso de la técnica AJAX, se recomienda el Internet Explorer, Mozilla Firefox u Opera.

Como una de las funcionalidades del sistema consiste en generar informes en formato PDF, se necesita tener instalado un visor de documentos portátiles (PDF) en la máquina cliente.

La resolución de pantalla debe ser 1024x768.

Servidor

El servidor debe contar con el Sistema gestor de base de datos POSTGRESQL con una versión mayor o igual a la 9.0, servidor Web Apache con una versión mayor o igual a la 2.2.8.

Hardware

Las computadoras clientes deben estar conectadas en red, y deben tener un procesador a 600 MGHZ o superior, con 128 MB de RAM como mínimo.

El servidor deberá tener un procesador a 1.0 GHz o superior, con 512 MB de memoria RAM como mínimo.

2.2 Actores del sistema

A continuación se muestran los distintos actores del sistema en la tabla 5 y el papel que desempeñan en la utilización de los requisitos funcionales:

Actores	Justificación
Usuario	Representa a todos los trabajadores o no, que harán uso en cierta forma del sistema. Es el rol primario del cual van a heredar todos los demás.
Administrador	Hereda del rol Usuario y es el rol principal encargado de realizar todas las operaciones del sistema.
Médico	Es el encargado de Actualizar la Ficha del Paciente,

Descripción y Elaboración de la Solución Propuesta

	Realizar Búsqueda por Criterios, hereda del rol Usuario.
Jefa de Servicio	Es la encargada de Visualizar Reporte Estadístico, hereda el rol que el Médico puede jugar respecto a los casos de uso Insertar Ficha del Paciente, Modificar Ficha del Paciente, Eliminar Ficha del Paciente, Realizar Búsqueda por Criterios.
Reloj	Representa al reloj del sistema, encargado de ejecutar cada cierto tiempo casos de usos automáticos Realizar Salvas Cruzadas y Backups automático de la BD.

Tabla 5 Actores del sistema

2.3 Diagrama de los Casos de Uso del Sistema

Un caso de uso especifica una secuencia de acciones, incluyendo variantes, que el sistema puede llevar a cabo, y que producen un resultado observable de valor para un actor concreto. (Jacobson 2000).

Paquetes del Sistema NeoGest

La organización del sistema se llevará a cabo en 3 paquetes principales Seguridad, Paciente y Reporte lo cual mejorará la estructura del sistema a desarrollar y permitirá modularizar la información a tratar. Los mismos se muestran en la figura 12 a continuación:

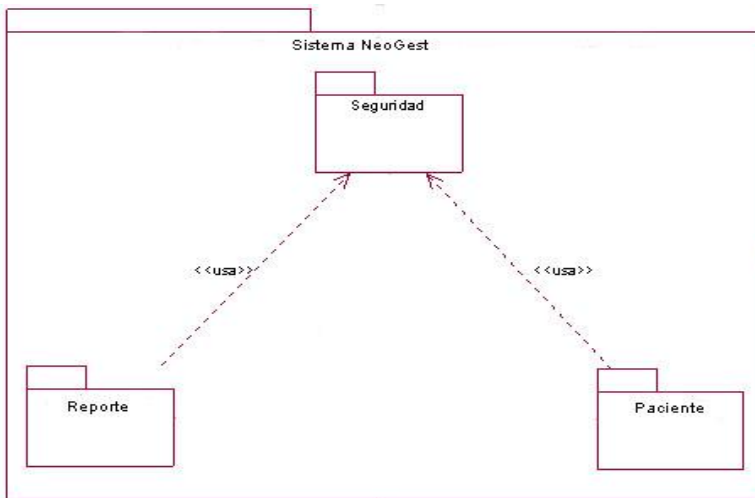


Figura 12 Diagrama de paquetes del Sistema NeoGest

2.3.1 Diagrama de Casos de Uso del Sistema del paquete de Seguridad

A continuación se muestra en la figura 13 los actores o roles del sistema con sus respectivas funcionalidades o tareas que los mismos juegan:

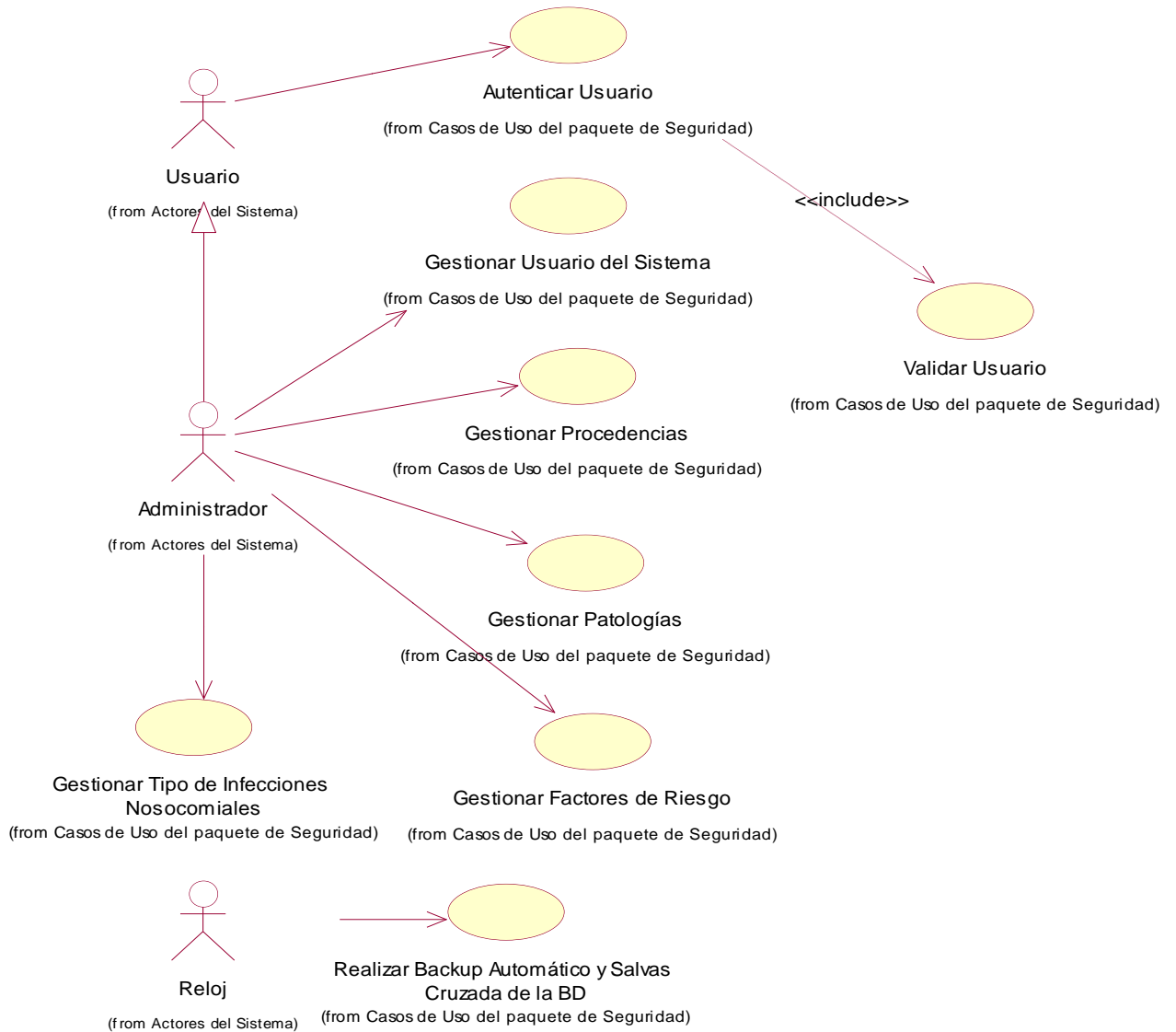


Figura 13 Diagrama de Casos de Uso del Sistema del paquete de Seguridad

2.3.2 Diagrama de Casos de Uso del Sistema del paquete Paciente y Reporte

A continuación se muestra en la figura 14 los actores o roles del sistema del paquete Reporte y Paciente con sus respectivas funcionalidades o tareas que los mismos juegan:

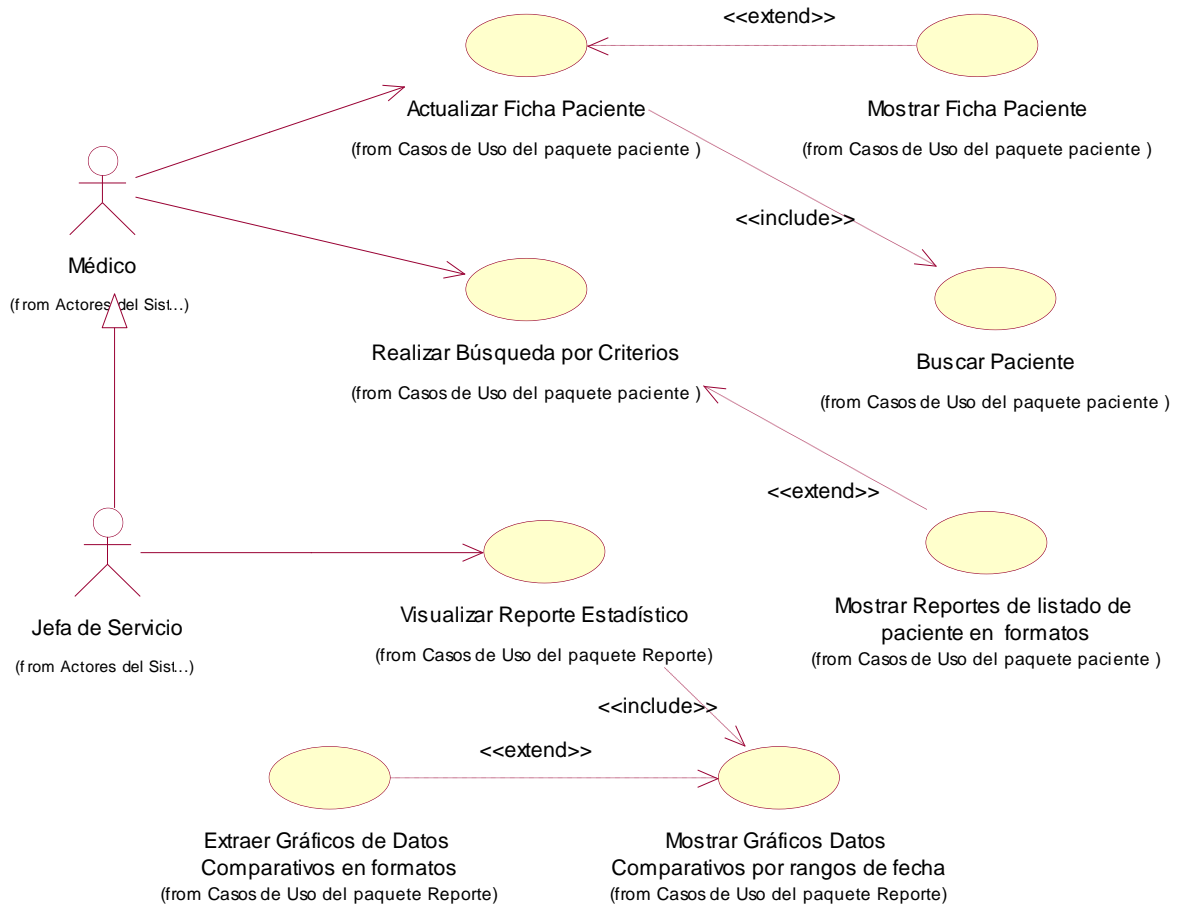


Figura 14 Diagrama de Casos de Uso del Sistema del paquete Paciente y Reporte

2.4 Diagrama de Generalización Especialización entre Actores del Sistema

Entre los diferentes actores del sistema se establece una jerarquía que define el acceso a la información según el rol al que pertenece el actor y los casos de uso del sistema con los que interactúa. Se utilizan las relaciones de

Descripción y Elaboración de la Solución Propuesta

generalización/especialización entre los actores del sistema los cuales pueden jugar el mismo rol en un caso de uso particular. En el siguiente diagrama de la figura 15 se muestran las relaciones entre los actores del sistema:

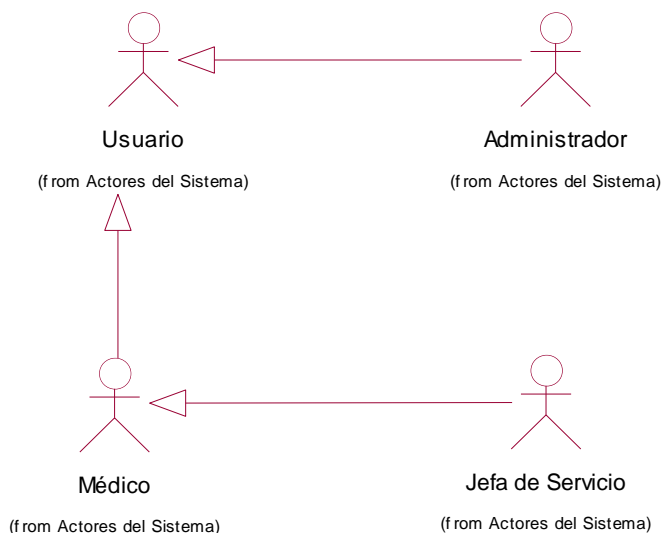


Figura 15 Diagrama de Generalización Especialización entre Actores del Sistema

2.5 Descripción Textual de los Principales Casos de Uso del Sistema en formato real y expandido

A continuación se describirán los principales Casos de Uso del Sistema (CUS).

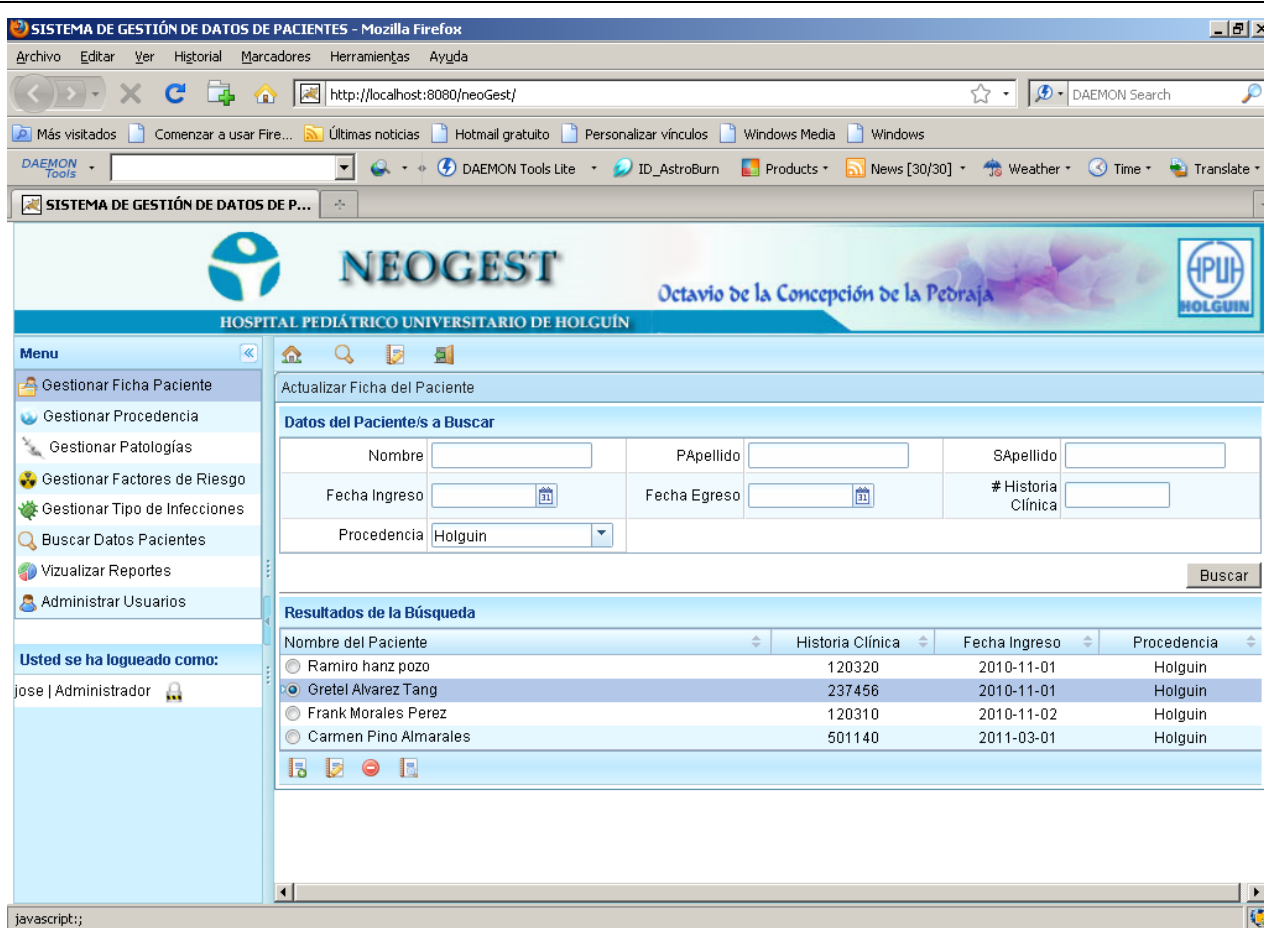
Caso de uso del sistema Actualizar Ficha Paciente

Caso de uso:	Actualizar Ficha Paciente
Actor(es):	Médico(inicia)
Propósito:	Mantener actualizada la Ficha de los datos del Paciente.
Resumen:	El caso de uso se inicia cuando el médico requiere registrar un nuevo paciente, actualizar la Ficha del mismo o eliminarla. (Insertar datos de la FP, modificar la FP o eliminar FP) de un paciente existente. El sistema interactúa siempre que sea necesario con la base de datos. De acuerdo a sus requerimientos inserta la FP, modifica la FP o elimina la FP, graba y el registro de la Ficha del Paciente queda

Descripción y Elaboración de la Solución Propuesta

	actualizado, finalizando el caso de uso.
Tipo:	Real y Expandido.
Referencias:	R9 R 9.1, R 9.2, R 9.3
Precondiciones:	El médico (usuario) ha ingresado al sistema y se encuentra en las opciones del menú principal.

Pantalla 3 Actualizar Ficha del Paciente



Flujo Normal de Eventos

Acción del Actor	Respuesta del Sistema
1. El Médico selecciona la opción "Gestionar FP" en la pantalla de inicio del	2. El sistema muestra la pantalla "Actualizar Ficha del Paciente".(Ver Fig. 3)

Descripción y Elaboración de la Solución Propuesta

sistema.	
3. El Médico requiere listar los pacientes bien por Nombre, primer apellido, segundo apellido, fecha de ingreso, fecha de egreso, No de historia clínica y/o procedencia. Llena los datos para la búsqueda(si lo desea filtrar) y pulsa el botón Buscar	4. El sistema muestra los resultados de la búsqueda en un listado en la misma pantalla Actualizar Ficha del Paciente donde muestra los pacientes con la patología y que coinciden con los otros datos correspondientes a filtrar.
5. El Médico selecciona otra opción del sistema o cerrar la aplicación	6. El sistema cambia de pantalla con la opción elegida o no aparece en pantalla.

Pantalla 4 Modificar Ficha del Paciente

Flujo Normal de Eventos

1. El Médico requiere modificar la ficha de un paciente y selecciona los indicadores por los que quiere buscar al paciente y	2. El sistema le muestra los Resultados de la búsqueda en la misma pantalla “Actualizar Ficha del Paciente” ver pantalla
--	--

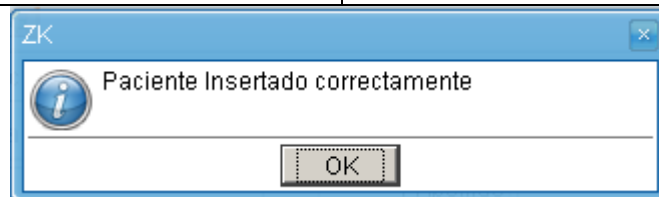
Descripción y Elaboración de la Solución Propuesta

oprime el botón Buscar .	
3. El Médico selecciona en un Listbox al paciente que quiere modificar y oprime el botón en la barra de botones Editar Paciente .	4. El sistema muestra la pantalla “Modificar Ficha del Paciente” con todos los indicadores anteriores seleccionados y editables para poder cambiarlos.(Ver Pantalla 4)
5. El Médico cambia aquellos indicadores que desee y oprime el botón Modificar .	6. El sistema modifica la información, valida que no haya errores al introducirla, la almacena en la BD, le muestra un mensaje de actualización satisfactoria y retorna nuevamente a la pantalla “Actualizar Ficha del Paciente”.
7. Si el Médico cambia aquellos indicadores que desee y oprime el botón Cancelar .	8. El sistema vuelve a la pantalla “Actualizar Ficha del Paciente”.
9. El Médico selecciona otra opción en el menú principal o cierra la aplicación	10. El sistema cambia de pantalla en base a la opción elegida o no aparece.
Pantalla 5 Insertar Ficha del Paciente	

Descripción y Elaboración de la Solución Propuesta

Flujo Normal de Eventos

- | | |
|---|--|
| 1. El Médico requiere insertar una nueva ficha del paciente y oprime el botón en la barra Añadir Paciente (pantalla 3). | 2. El sistema muestra la pantalla "Insertar Ficha del Paciente".(Pantalla 5) |
| 3. El Médico llena la información correspondiente a los indicadores y oprime el botón Insertar. | 4. El sistema valida la información y la almacena en la BD si está correcta, le muestra un cartel al usuario de inserción satisfactoria y vuelve nuevamente a la pantalla "Actualizar Ficha del Paciente". |



- | | |
|--|--|
| 5. Si el Médico llena la información y | 6. El sistema vuelve la pantalla "Actualizar |
|--|--|

Descripción y Elaboración de la Solución Propuesta

oprime el botón Cancelar.	FP”.
7. El Médico selecciona otra opción en el menú principal o cierra la aplicación	8. El sistema cambia de pantalla en base a la opción elegida o no aparece.
Poscondiciones	Queda actualizado el registro de los pacientes.
Cursos Alternos	Antes de realizar cualquier acción el sistema verifica si el servidor de BD está apagado o existe algún problema con la conexión, de ser así entonces muestra un mensaje de error. Antes de insertar los datos en el registro el sistema los valida, en caso de errores en los datos, se muestra un mensaje de error de acuerdo con el dato erróneo o faltante.

Tabla 6 Descripción textual del CUS Actualizar Ficha Paciente

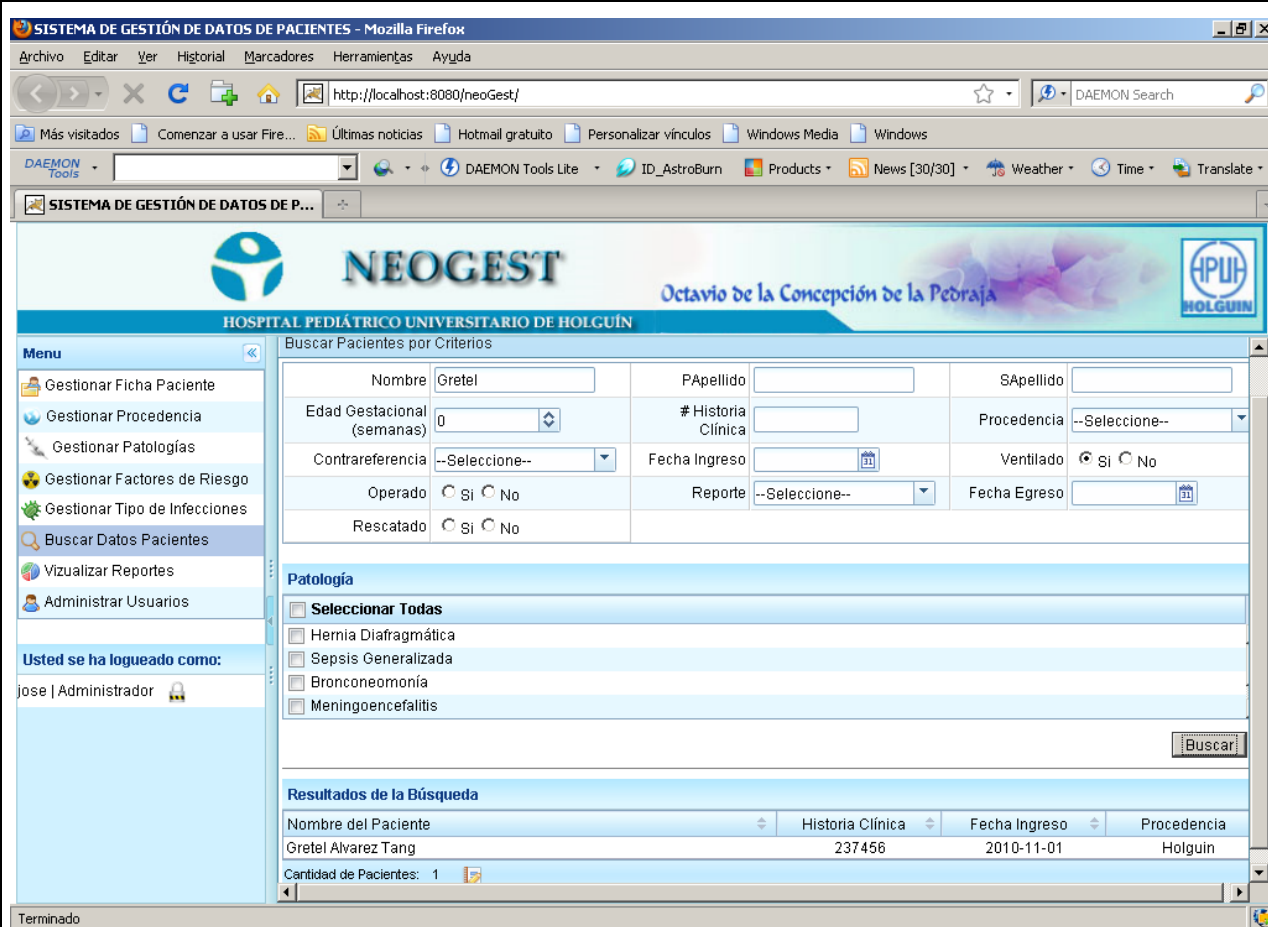
Descripción Textual en formato real y expandido del CUS Realizar Búsqueda por Criterios

Caso de uso:	<i>Realizar Búsquedas por Criterios</i>
Actor(es):	Médico
Propósito:	Listar pacientes que cumplen con determinados criterios de búsqueda ya sea para mostrar los resultados o bien reportes de dichas búsquedas.
Resumen:	El caso de uso se inicia cuando el médico requiere listar los pacientes que cumplan con determinados criterios de búsquedas ya sea para la obtención de dichos datos como modo de consulta, o realizar reportes. De acuerdo a los requerimientos lista los pacientes que cumplen con los parámetros de búsqueda o crea un reporte con los resultados finalizando el caso de uso.
Tipo:	Real y Expandido.

Descripción y Elaboración de la Solución Propuesta

Referencias:	R12: R12.1, R12.2, R12.3, R12.4, 12.5, R12.6, R12.7, R12.8, R12.9, R12.10, R12.11, R12.12, R12.13, R12.14, R13
Precondiciones:	El médico (usuario) ha ingresado al sistema y se encuentra en las opciones del menú principal.

Pantalla 6. Buscar Ficha de Paciente por Parámetros



Flujo Normal de eventos

<p>1. El Médico selecciona la opción “Buscar Datos Paciente” en la pantalla de inicio del sistema.</p>	<p>2. El sistema genera la pantalla “Buscar Pacientes por Criterios” en la que se muestran los indicadores con los que se puede realizar la búsqueda. (Ver Pantalla 6).</p>
--	---

Descripción y Elaboración de la Solución Propuesta

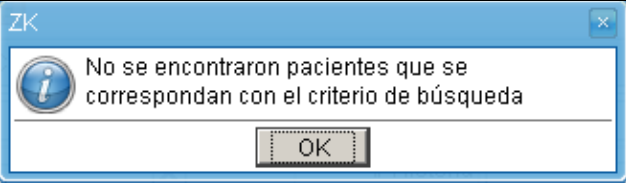
<p>3. El Médico elige los indicadores por los que quiere listar a los pacientes ya sea uno, la combinación, todos los criterios o bien ninguno y presiona el botón “Buscar”.</p>	<p>4. El sistema ejecuta la búsqueda de la información y le muestra nuevamente la pantalla “Buscar Ficha de Paciente por Criterios” con el listado de pacientes coincidentes en los Resultados de la Búsqueda si hay coincidencias sino muestra un mensaje de información de que no existen coincidencias.</p>
	
<p>5. El Médico selecciona otra funcionalidad del sistema o cierra la aplicación.</p>	<p>6. El sistema cambia la pantalla o no aparece.</p>
<p>Poscondiciones</p>	<p>Es obtenida o visualizada la Lista de pacientes.</p>
<p>Cursos Alternos</p>	<p>Antes de realizar cualquier acción el sistema verifica si el servidor de BD está apagado o existe algún problema con la conexión, de ser así entonces muestra un mensaje de error. Antes de buscar los datos en el registro el sistema los valida, en caso de errores en los datos, se muestra un mensaje de error de acuerdo con el dato erróneo.</p>

Tabla 7 Descripción textual del CUS Realizar Búsqueda por Criterios

El resto de las descripciones textuales en formato real y expandido de los CUS pueden consultarse en el [Anexo 3](#).

2.6 Valoración de Sostenibilidad del Producto informático

La valoración de sostenibilidad de un producto informático (PI) no es más que el proceso de evaluación de impactos ambientales, socio humanistas, administrativos y tecnológicos de un producto informático, previsible desde el diseño del proyecto, que favorece su autorregulación, para la satisfacción de la necesidad que resuelve, con un uso racional de recursos y la toma de decisiones adecuadas a las condiciones del contexto y el cliente. (García 2006)

DIMENSIÓN ADMINISTRATIVA

Es aquella relacionada con el ahorro, gasto, calidad de la producción y los servicios, administración de recursos, así como la toma de decisiones administrativas.

Para el análisis de la factibilidad económica del software llevado a cabo en el sistema **NeoGest**, se utilizó el Modelo Constructivo de Costos 2.0 (COCOMO, por sus siglas en inglés).

El COCOMO tiene como objetivo la estimación del costo asociado al desarrollo de un software. Utiliza los puntos de función para calcular el tamaño de un producto de software y el esfuerzo asociado a su desarrollo. El empleo de COCOMO permite realizar una estimación de la cantidad de personas que se requieren utilizar para el desarrollo de un sistema, así como el tiempo de desarrollo aproximado. (Ruiz 1999)

Las tablas y fórmulas asociadas pueden ser consultadas en el [Anexo 4](#).

Según el método de estimación de costos COCOMO elaborar el PI el sistema se puede desarrollar en 4 meses aproximadamente dedicando 2 hombres por mes incurriendo en un costo total de \$ 1600. El sistema de implementarse brindará en un futuro muchas ventajas a la institución posibilitando un trabajo eficiente y con un costo moderado del mismo.

Se puede decir que con el PI NeoGest se ahorran materiales de impresión y papel, se ahorra el tiempo necesario en las actividades relacionadas con el proceso de gestión de las fichas de los pacientes así como la confiabilidad y disponibilidad de los reportes y la información referentes a los mismos.

Descripción y Elaboración de la Solución Propuesta

Con el PI se gestiona de una manera rápida, eficaz y eficiente toda la información referente a los pacientes, logrando una mejor calidad en el proceso de gestión.

El sistema no genera ingresos monetarios directos o indirectos.

El PI ayuda en gran medida al proceso de toma de decisiones administrativas ya que agiliza el proceso de gestión de la información de los pacientes de la sala de neonatología, logrando disminuir considerablemente el tiempo en la generación de reportes y búsquedas relacionadas con la toma de decisiones.

Todas las herramientas y software utilizados son de carácter libre o código abierto, por lo que no se incurren en gastos monetarios referentes a la utilización de los mismos.

DIMENSIÓN SOCIO-HUMANISTA

Esta dimensión está relacionada con el modo de vida, desarrollo de un grupo social, satisfacción de necesidades sociales, formación ético-humanista de los gestores del PI, la ciencia y la tecnología como procesos sociales.

El sistema **NeoGest** favorece al proceso de gestión de la información de los pacientes, propiciando una mayor eficiencia, control y calidad en las actividades. El producto ha sido desarrollado teniendo en cuenta las necesidades del personal que ha de utilizarlo.

Referente a la necesidad social que el mismo resuelve y por las perspectivas y similitud en el proceso de gestionar la información en los servicios de Neonatología del país puede ser generalizado de ser puesto en marcha en los mismos.

El PI disminuye el tiempo y esfuerzo físico de los trabajadores al no tener que realizar manualmente todo el proceso de gestión de la información así como las probabilidades de errores a la hora de manipular la información.

No se generan ni se disminuyen empleos con el uso del PI ya que la finalidad del mismo es resolver un problema social y facilitar las condiciones de trabajo de los clientes, ajustándose a las políticas definidas por la entidad.

El PI aumenta la cultura profesional y general, ya que introduce al usuario en el mundo de la Web, siendo hoy en día uno de las plataformas de intercambio de información más popular. Desarrolla en los trabajadores la responsabilidad,

Descripción y Elaboración de la Solución Propuesta

haciéndole entender la necesidad de utilizar correctamente el sistema e informar de cualquier fallo que esté presente.

La aceptación del producto se favorece dando a conocer en las reuniones las capacidades de un producto informático para gestionar de manera rápida la información, en vez de realizar el proceso de gestión de forma manual, lo cual trae consigo un conjunto de deficiencias que entorpecen el desarrollo del mismo.

Por los beneficios que trae consigo el uso del PI no ha existido hasta el momento rechazo por parte de los trabajadores, aunque es recomendable capacitarlos para usarlo.

DIMENSIÓN AMBIENTAL

La dimensión está relacionada con las condiciones favorables o no a las personas o cosas, si el PI minimiza daños e impactos.

El PI ahorra fundamentalmente papel, materiales de impresión lo cual tiene aspectos positivos ya que evita la tala de árboles en cierta medida así como el proceso de producción industrial de dichos productos.

Se utilizan colores adecuados en la gama de los azules claros y un tamaño de letra que no dañe ni canse la vista, logrando una mayor identificación del usuario con la aplicación, no genera contaminación por ruido.

Una interfaz agradable al entorno del usuario favorece a que el mismo no se sienta confundido o perdido, crear un estándar o una política de diseño le facilita al usuario el trabajo con la aplicación ya que va creando un patrón de trabajo.

Para evitar el estrés psicológico de los usuarios por el uso del PI se crea un Manual de Usuario para el trabajo con la aplicación, el cual le mostrará los pasos principales para su correcto uso.

Todos los software que se utilizaron en el desarrollo del PI son libres, esto contribuye a que se puedan reutilizar códigos existentes en el mundo y que sirvan para disminuir el tiempo de desarrollo. Así como el mismo código generado puede ser reutilizado con vista a la implementación de otros módulos o nuevos requerimientos del sistema.

Descripción y Elaboración de la Solución Propuesta

Por la forma de gestionar la información similar a otros procesos de gestión de información en la rama de la medicina, el PI puede ser utilizado aplicándosele cambios necesarios.

La implantación del PI aumenta el uso del ordenador pudiéndole causar enfermedades a los usuarios. Por lo que se recomienda una correcta postura de los usuarios en las sillas, así como tener el monitor a la altura de los ojos y utilizar protector de pantalla.

Por todo lo antes mencionado la solución propuesta en el presente trabajo no tendrá un impacto negativo sobre el medio ambiente.

DIMENSIÓN TECNOLÓGICA

La dimensión está relacionada con el uso de tecnología adecuada y asimilable con el usuario.

El usuario tiene cierta experiencia con el trabajo en entornos web, esto favorece a la aplicación del PI no obstante es necesario la capacitación adecuada de todos los implicados así como el uso del Manual de Usuario.

El Hospital Pediátrico Octavio de la Concepción y de la Pedraja a pesar de algunas deficiencias en este sentido cuenta con computadoras en casi todos los servicios o salas conectadas a la red, además cuenta con un nodo central con personal calificado.

El PI es implementado con software libre por lo que el cliente puede realizar cambios o modificaciones.

La decisión tecnológica más apropiada asimilable al usuario para elaborar y aplicar el PI es el desarrollo de una aplicación Web ya que la misma le permitirá al usuario el acceso desde distintas estaciones de trabajo, interfaz amigable y conocida, permite actualizar la información constantemente entre otras cosas.

El sistema puede verse afectado por la ruptura del servidor de Base de Datos, por lo tanto este cuenta con la posibilidad de realizar salvallas periódicamente.

A partir de la valoración de sostenibilidad en las dimensiones administrativa, socio-humanista, ambiental y tecnológica del sistema NeoGest, expresada anteriormente se llegó a la conclusión que la solución empleada es factible, el sistema es sostenible y perdurable en el tiempo por la necesidad que existe de

una herramienta informática que les permita gestionar de una forma rápida y fiable la información referente a los pacientes en el servicio de Neonatología del HPUH.

2.7 Flujo de diseño

Según la metodología RUP el flujo de trabajo del diseño tiene la responsabilidad de modelar el sistema y encontrar la forma para que este soporte todos los requisitos necesarios para cumplir su objetivo. Este flujo describe la realización física de los casos de uso usando artefactos como diagramas de clases. (Jacobson 2000)

En el diseño, el sistema es modelado y se conforma para que soporte todos los requisitos que se le suponen, adquiriendo una comprensión en profundidad de los no funcionales, restricciones relacionadas con el lenguaje de programación a utilizar, componentes reutilizables, entre otros. (Jacobson 2000)

Una clase incluye definiciones para atributos y operaciones. Es una construcción similar en la implementación del sistema, es decir, se pueden establecer comparaciones entre los componentes de las clases de diseño y los de una aplicación.

Los elementos más comunes que conforman la tecnología Web son las entre otras las páginas y los formularios. El lenguaje de modelado UML brinda varios estereotipos para extensiones Web, entre los más comunes se encuentran:

<<**Server Page**>> Representa la página Web que radica en el servidor, la cual contiene código que se ejecuta en este. La misma como un patrón de diseño en el sistema se le pondrá el prefijo SV.

<<**Client Page**>> Representa a la página Web que son entregadas al navegador, con formato HTML y a veces con código que puede ser ejecutado por el navegador. A estas páginas clientes se le pondrán el prefijo CL.

<<**Html Form**>> Colección de elementos de entrada que son parte de un página cliente. A los formularios se le pondrán el prefijo Fr.

2.7.1 Organización de los casos de uso a desarrollar por paquetes

El sistema se distribuyó en varios paquetes del diseño como se muestran a continuación:

Descripción y Elaboración de la Solución Propuesta

- Paquete (0): Diagrama de clases del menú principal.
- Paquete (1): Gestionar Ficha del Paciente.
- Paquete (2): Realizar Búsqueda por Criterios.
- Paquete (3): Visualizar Reportes.
- Paquete (4): Gestionar Usuarios.
- Paquete (5): Gestionar Procedencia.
- Paquete (6): Gestionar Factores de Riesgo.
- Paquete (7): Gestionar Patologías.
- Paquete (8): Gestionar Tipo de Infecciones.

Diagrama de Clases del diseño del paquete 0 (Menú principal)

A continuación se muestra la figura 16 que refleja la distribución del paquete del menú principal:

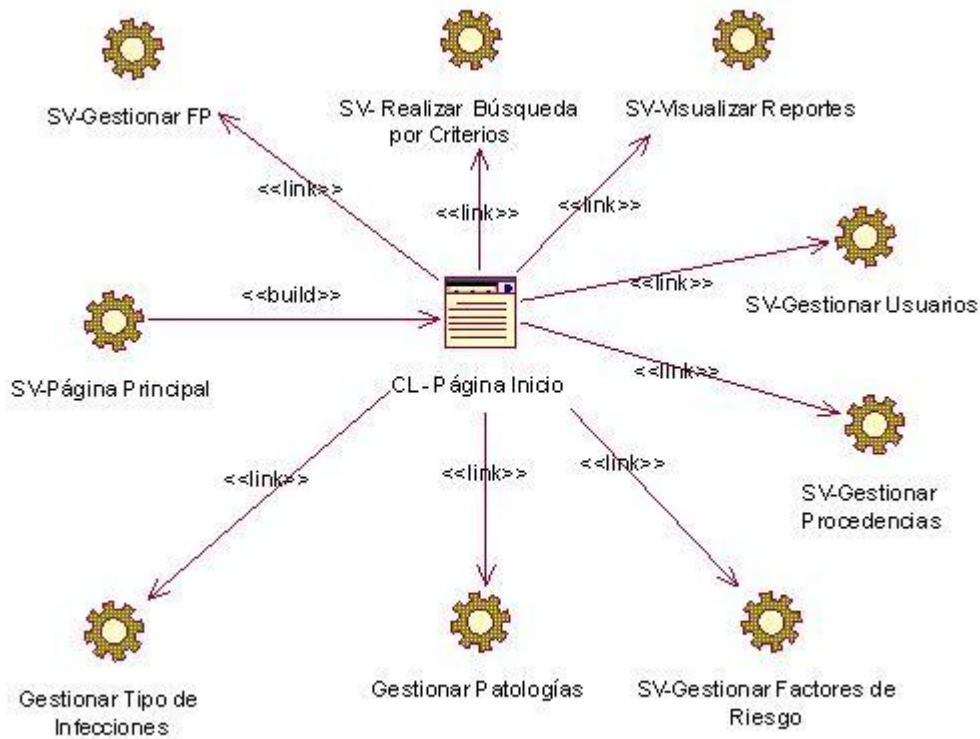


Figura 16 Diagrama de Clases del diseño del paquete del menú principal

Descripción y Elaboración de la Solución Propuesta

El resto de los diagramas de Clase del Diseño Web pueden encontrarse en el [Anexo 5](#).

El desarrollo modular fue el patrón de diseño utilizado en la implementación del sistema NeoGest, y por ello en la modelación realizada en los diagramas de clases del diseño se tuvo en cuenta dicho patrón. El desarrollo modular permite modularizar las aplicaciones Web, de forma que toda la aplicación se ejecute sobre una misma página, siendo esto un factor importante de seguridad ya que el usuario o persona que navegue ya sea legal o ilegal no podrá observar el camino que se describe en la URL de la página.

La clase servidora sv- index o inicio modela a la página encargada de incluir todas las otras páginas del sistema, por lo cual será el único punto de entrada. Esto representa una gran ventaja a la hora de programar, ya que se colocan todas las tareas repetitivas, inicialización de variables, conexión a la Base de Datos y lectura de otros archivos de funciones o configuración en un solo fichero.

2.8 Diagrama de despliegue

El modelo de despliegue pertenece al flujo de trabajo diseño. Es un modelo de objetos que describe la distribución física del sistema en términos de cómo se distribuye la funcionalidad entre los nodos de cómputo. Contiene los nodos que forman la topología hardware sobre la que se ejecuta el sistema y la distribución de las partes del sistema en ellos. [Jacobson, 2000].

A continuación se muestra la figura 17 que muestra la distribución o ubicación de los nodos de cómputo:

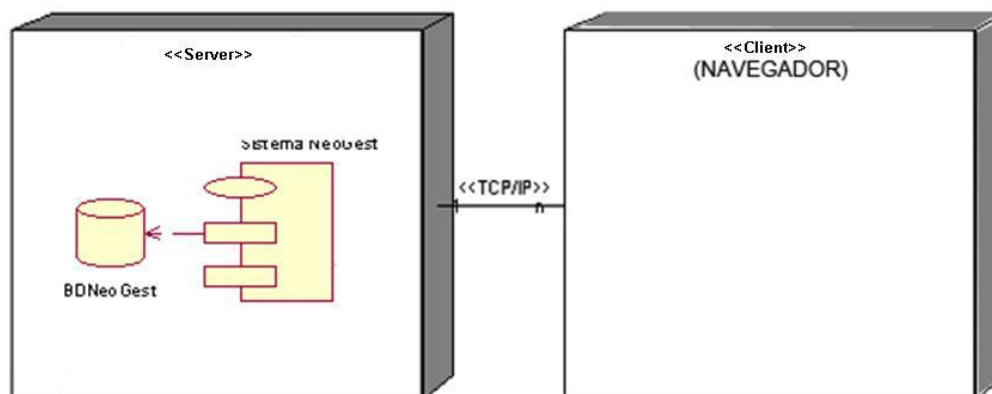


Figura 17 Diagrama de despliegue

El nodo a la izquierda representa al Servidor Central de la aplicación, en él están ubicados el Servidor Web y el Servidor de Base de datos, a la derecha se encuentran simbolizadas bajo el nombre de Cliente, todas aquellas computadoras desde las cuales se accederá al sistema, se visualizará la información, etc.

2.9 Flujo de implementación

“En la implementación empezamos con el resultado del diseño e implementamos el sistema en términos de componentes, es decir, ficheros de código fuente, scripts, ficheros de código binario, ejecutables y similares. “ (Jacobson 2000)

El modelo de implementación describe también cómo se organizan los componentes de acuerdo con los mecanismos de estructuración y modularización disponibles en el entorno de implementación y en el lenguaje o lenguajes de programación utilizados.(Jacobson 2000)

El lenguaje de modelado UML consta de varios estereotipos para clasificar los componentes de implementación:

- executable: representa a un programa que puede ser ejecutado
- file: representa a un fichero que contiene código fuente o datos
- library: representa a una librería estática o dinámica
- table: representa a una tabla de la base de datos
- document: representa a un documento

Descripción y Elaboración de la Solución Propuesta

En el modelado del diagrama de componentes se creó un nuevo estereotipo <<ZUL-page>> que representa la página .ZUL que contiene los componentes de la vista.

El diagrama de componentes del Sistema NeoGest puede ser consultado en el [Anexo 7](#).

Empleo de XML en el sistema

El lenguaje extensible de marcado (XML por sus siglas en inglés) se ha convertido con el devenir del tiempo en uno de los formatos de intercambio de información más usado, así como también se puede encontrar que se utiliza para almacenar información, configurar sistemas, definir procesos y relaciones, exportar datos, etc. En fin, el XML es un lenguaje que debido a sus características son muchas sus aplicaciones.

En el proceso de desarrollo del sistema propuesto en la presente investigación se empleó el XML en gran medida, con el objetivo de permitir una mejor evolución y mantenimiento del sistema. A continuación se describe el empleo del XML en la construcción del sistema:

Definición de las interfaces de usuarios: El framework ZK permite crear las interfaces de usuarios de dos maneras, mediante código Java o a través del lenguaje ZUML (ZK User Markup Language).

Utilizar la primera variante de código Java implica emplear mayor tiempo en el diseño de las interfaces de usuarios y mayor complejidad en el mantenimiento del software. Una vez compiladas las clases Java a código binario, para realizar el mantenimiento del sistema sería necesario descompilar las clases.

La variante utilizada para el diseño de las interfaces de usuarios fue emplear el lenguaje ZUML. Este lenguaje es una extensión del XML, donde se encuentran definidos todos los componentes visuales mediante etiquetas. Con ZUML el diseño de las interfaces de usuarios es mucho más sencillo y legible, siendo menor el tiempo de desarrollo. Por otra parte cuando se requiere un mantenimiento del sistema en el cual es necesario cambiar las interfaces de usuarios o realizar cualquier cambio de esta índole, sólo se debe modificar el o los

Descripción y Elaboración de la Solución Propuesta

ficheros ZUL necesarios, siendo efectivo el cambio automáticamente en el servidor.

Aspectos de interés en relación a la implementación

En la construcción o diseño de las páginas del sistema que como bien se mencionó anteriormente se utilizó el lenguaje ZUML, con el objetivo de hacer más comprensible o darle estructura a los componentes visuales de la aplicación se cumplió con la siguiente estructura mostrada en la tabla 8 a continuación:

Componente	Prefijo	Ejemplo
Button	btn	Button btnBuscar;
Textbox	tbx	Textbox tbxNombre;
Listbox	lbx	Listbox lbxDiagnosticos;
Window	Win	Window winDP;
Page	page	Page pageInicio;
Spinner	sp	spEdad
Datebox	dbx	dbxFecha_Inicio
Intbox	ibx	ibxHC
Panel	pn	pnNuevoUsuario
Toolbarbutton	tbtn	tbtnEliminar_Usuario
Combobox	cbx	cbxProcedencia

Tabla 8 Estándar o estructura de codificación de los componentes ZK

En relación a la implementación o estructura del código así como la declaración de paquetes, métodos, clases se siguió el siguiente patrón.

Paquetes

Los paquetes siempre se nombrarán en minúscula. Ejemplo **seguridad, módulos, reporte.**

Métodos

Los nombres de métodos siempre empezarán por un verbo en minúscula y lo que le sigue irá en mayúscula sólo la primera letra de las palabras internas (notación camello).

Los mismos tendrán un previo comentario con una breve descripción de lo que se desea hacer.

Clases

Descripción y Elaboración de la Solución Propuesta

Las clases se nombrarán con mayúsculas y separadas por guión bajo. Ejemplo: Modificar_Ficha_Paciente. Las clases controladoras se nombrarán de igual manera seguidas de la palabra Controller. Ejemplo: Modificar_Ficha_PacienteController.

Las clases persistentes de la aplicación se nombrarán con mayúsculas. Ejemplo: Paciente.

Tablas de la base de datos

Las tablas de la base de datos se nombrarán con el prefijo tb ejemplo tbpaciente, ver Diagrama de clases persistentes [Anexo 6](#).

Seguridad del Sistema

El Administrador del sistema es el encargado de gestionar los usuarios, además de la gestión general de todas las funcionalidades de la aplicación. El sistema permite a un usuario desempeñar un rol, mostrándole solo aquellas funcionalidades a las cuales tiene derecho realizar.

A continuación se describen algunos aspectos de la seguridad del sistema:

URL única: Una vez mostrada la página de bienvenida del sistema, los invitados o usuarios siempre estarán viendo la misma URL en el navegador, debido a que el sistema está implementado en su totalidad con la técnica AJAX y se incluyen dinámicamente las páginas ZUL sobre el contenedor principal del sistema, por lo que nunca ocurre una recarga completa de las páginas o redireccionamiento hacia otras URL. Esta característica permite que los usuarios no conozcan en ningún momento la estructura de carpetas y ficheros del sistema.

Encriptación de contraseñas: Para encriptar las contraseñas de los usuarios se utiliza el algoritmo SHA-1. Las contraseñas encriptadas se encuentran almacenadas en la base de datos del sistema. Para que los usuarios puedan entrar en el sistema, deben primero autenticarse en el panel de autenticación que se encuentra en la página de bienvenida. La contraseña del usuario nunca viaja al servidor en texto plano lo que impide que sea descifrada con facilidad ya que la misma se encripta del lado del cliente.

Creación de sesión: Una vez que se comprueba que la persona que se está autenticando es un usuario del sistema, se crea una sesión la cual contiene datos del usuario, tales como: nombre de usuario y rol en el sistema.

Seguridad en los controladores de las vistas: Cuando se invoca una página ZUL, el controlador correspondiente a la misma comprueba en la sesión, que el usuario tenga el privilegio necesario de acuerdo al rol que juega para acceder a la página, de no tenerlo o no estar logueado es automáticamente redireccionado hacia la página de bienvenida del sistema.

2.10 Modelo de prueba

El modelo de prueba describe cómo se prueban los componentes ejecutables (como las construcciones) en el modelo de implementación con pruebas de integración y de sistema. El modelo puede describir también cómo han de ser probados aspectos específicos del sistema, por ejemplo, si la interfaz de usuario es utilizable y consistente o si el manual de usuario del sistema cumple con su cometido. Es una colección de casos de prueba, procedimientos de prueba y componentes de prueba. (Jacobson, 2000)

Es importante construir el software incrementalmente en pasos manejables, de forma que cada paso de lugar a pequeños problemas de integración o prueba. El resultado de cada paso es llamado “construcción”, que es una versión ejecutable del sistema. Cada construcción es sometida a pruebas de integración antes de que se cree ninguna otra. Por otra parte resulta necesario una vez completado el software realizarle pruebas como un todo.

Teniendo en cuenta lo antes expresado se desarrollaron comprobaciones en cada construcción, que permitieron probar de manera sistémica si el comportamiento del sistema descrito en cada caso de uso estaba implementado correctamente. Los problemas detectados como resultado de las pruebas realizadas se tomaron en cuenta y a la vez fueron corregidos en las siguientes iteraciones.

Una vez terminado el sistema se pasó a probarlo como un todo, para ello se siguió la siguiente estrategia:

- Cada caso de uso fue probado para su flujo normal y sus flujos alternativos.

Descripción y Elaboración de la Solución Propuesta

- Seis pruebas de instalación, verificando que el sistema pudiera ser instalado en varias configuraciones de hardware y sistemas operativos.
- Seis pruebas negativas verificando cómo se comporta el sistema. Cuatro pruebas ante la caída repentina de la red, y dos ante un servidor con 256 MB de RAM. Ante la caída repentina de la red el sistema respondió como se esperaba, advirtiendo al usuario acerca de los problemas de conexión. En las pruebas negativas que se efectuó en un servidor con 256 MB de RAM el sistema corrió con dificultad, pero aún se podía hacer uso de sus funcionalidades.

2.10.1 Casos de pruebas

Un caso de prueba específica una forma de probar el sistema, incluyendo la entrada o el resultado con que se ha de probar y las condiciones bajo las que ha de probarse. Es una forma de organizar las pruebas del sistema relacionadas con la funcionalidad que tenga, es por eso que los casos de prueba están directamente relacionados con los casos de uso que prueban. Es un conjunto de entradas y resultados esperados que ejercitan a un componente con el propósito de causar fallas y detectar defectos.

Una de la forma de control más importante que se aplica a la hora de realizar la implementación son los casos de prueba para verificar el resultado de la implementación probando cada construcción, incluyendo las versiones finales que se entregarán a los clientes y usuarios, esto posibilita detectar los posibles errores y erradicarlos con mayor facilidad dándole más calidad, Robustez y confiabilidad al software.

Se realizaron casos de pruebas para los principales casos de uso, los mismos pueden ser consultados en el [Anexo 8](#).

2.10.2 Procedimientos de pruebas

Un procedimiento de prueba es una especificación de cómo llevar a cabo la preparación, ejecución, y evaluación de los resultados de un caso de prueba particular. Los procedimientos de prueba también pueden derivarse de los casos de uso. Los defectos hallados se analizan para localizar el problema. Después

Descripción y Elaboración de la Solución Propuesta

estos problemas se priorizan y se corrigen por orden de importancia. [Jacobson,2000]

Los Procedimientos de pruebas se pueden consultar en el [Anexo 8](#).

2.11 Valoración de los resultados obtenidos

Una vez implantado el sistema NeoGest resultante, se realizaron encuestas a los usuarios, y entrevistas al cliente, con el objetivo de evaluar la satisfacción de los mismos respecto al producto.

2.11.1 Encuestas

La encuesta aplicada ([Anexo 9](#)) fue procesada por el método Delphy ([Anexo 10](#)), Este método se basa en la utilización sistemática del juicio intuitivo de un grupo de expertos para obtener un consenso de opiniones informadas, o también en la utilización sistemática del criterio práctico de un grupo de usuarios para llegar a un acuerdo de opiniones sobre la posible aplicabilidad de una propuesta determinada.(Rodríguez Expósito 1997)

Los aspectos medidos para encontrar el consenso de los encuestados fueron los siguientes:

- Las funcionalidades de la aplicación Web cubren las necesidades de los usuarios
- El sistema actual permite que el usuario se sienta familiarizado con la aplicación Web
- El producto informático permite el ahorro de recursos como el tiempo y materiales de oficina
- Agiliza el proceso de gestión de información de los pacientes del servicio Neonatal
- Se logró disminuir la probabilidad de errores al manipular la información
- El diseño de la aplicación Web resulta intuitivo y facilita la navegabilidad
- El sistema actual facilita el acceso a la información referente al proceso de gestión de información de los pacientes del servicio de Neonatología
- El tiempo de respuesta de la aplicación Web en la generación de búsquedas y reporte es más rápido

Descripción y Elaboración de la Solución Propuesta

- El sistema actual está acorde con el Procedimiento Específico que rige al proceso de gestión de información de los pacientes del servicio Neonatal

Las encuestas se realizaron a los usuarios del sistema NeoGest, encuestándose a aquellos Médicos, Jefa de Servicio y Administradores, que posteriormente harán uso del sistema y que pudieran establecer comparaciones con la manera actual de gestión de información.

Los usuarios seleccionados poseen conocimientos sobre el proceso de gestión de información en el servicio así como de computación y el ambiente Web. Tienen una alta preparación académica, todos son graduados de nivel superior, en su mayoría M.Sc.

El resultado final arrojó que los encuestados concuerdan en que están “totalmente de acuerdo” con todos los aspectos relacionados y las valoraciones generales del sistema actual emitidas por los encuestados fueron muy favorables.

2.11.2 Entrevista

Se realizaron entrevistas ([Anexo 11](#)) con el objetivo de evaluar la satisfacción del cliente respecto a los resultados obtenidos del proceso de implantación del sistema NeoGest. Los entrevistados, de manera general, coincidieron en que están satisfechos con los resultados obtenidos, valorando que se logró aumentar la seguridad del sistema, disminuyó la probabilidad de errores al manipular la información, el tiempo de respuesta de la aplicación Web es más rápido y se ajusta más a lo que realmente desea el cliente.

Se valoró dentro del sistema el módulo de reportes en general como una funcionalidad de valor incalculable, la cual permite disminuir los tiempos en la generación de los mismos, cobrando mayor importancia si se tiene en cuenta que el hecho de disminuir este tiempo incide directamente en el rápido desempeño del servicio en su afán de brindar atención con calidad a los neonatos y de remitir sus resultados a esferas superiores. Se emitieron opiniones favorables, resaltando las posibilidades que brinda el sistema de filtrar la información, así como la posibilidad de mostrar los informes en varios formatos (PDF, XLS, RTF).

Conclusiones del capítulo

Se concluye que el desarrollo de un buen producto de software además de consistir en comprender el ámbito del trabajo a realizar, los recursos que se requieren, y las tareas específicas a ejecutar, es necesario tener en cuenta si el mismo es sostenible pues esto puede influir grandemente en la aceptación y desarrollo del sistema deseado.

El sistema NeoGest se valora como sostenible en las dimensiones socio-humanista, ambiental, administrativa y tecnológica. El mismo perdurará en el tiempo por la necesidad de su uso.

La satisfacción de los usuarios y el cliente fue percibida a través de encuestas y entrevistas, que arrojaron que el sistema cumple con las expectativas de los usuarios y cliente, y resuelve el problema definido en esta investigación.

Se mostró el desarrollo del Sistema en las distintas etapas de la metodología de desarrollo de software empleada, reflejando que, sin lugar a dudas, una óptima planificación del proyecto contribuye a un desarrollo organizado y por ende a la calidad del producto obtenido.

Conclusiones Generales

Con el desarrollo de este trabajo de tesis que está orientado a la concepción de una herramienta informática para la gestión de información de los pacientes del servicio de Neonatología del Hospital Pediátrico Universitario de Holguín, se ha dado cumplimiento al objetivo de la investigación, dando como resultado una aplicación Web (NeoGest) que se caracteriza por la confiabilidad de los datos, seguridad, usabilidad y rendimiento, además de constar con funcionalidades que favorecen al proceso de gestión de información al cual responde.

Contar con un histórico de cada paciente desde que este ingresa en el servicio es de vital importancia para profundizar en los procesos de análisis integral de los pacientes, así como la generación de reportes que repercute en la evaluación, caracterización y desempeño del servicio, a partir de elementos precisos y concretos.

Las tecnologías, herramientas y metodologías empleadas para el desarrollo del sistema fueron acertadas y adecuadas, favoreciendo en gran medida al progreso del mismo, cubriendo completamente las necesidades de los clientes y desarrollador respectivamente.

El sistema NeoGest se valora como sostenible en las dimensiones socio-humanista, ambiental, administrativa y tecnológica. El mismo perdurará en el tiempo por la necesidad de su uso.

Actualmente la provincia de Holguín cuenta con un sistema de gestión de información de los pacientes en servicios neonatales, siendo NeoGest el primero en ponerse en marcha en el HPUH.

La hipótesis enunciada al inicio de esta investigación queda demostrada a través de los resultados arrojados en las encuestas y entrevistas realizadas. Lo que permite arribar a la conclusión que con el sistema NeoGest se logra favorecer el proceso de gestión de información de los pacientes del servicio de Neonatología del HPUH Octavio de la Concepción y la Pedraja.

Recomendaciones

Por los resultados obtenidos en esta investigación y para continuar el desarrollo de este trabajo se recomienda:

- Agregar funcionalidades al módulo de reportes con el objetivo de obtener otros datos que puedan resultar de interés
- Agregar un módulo de imágenes para una mejor comprensión de los exámenes, patologías y anomalías visibles en el neonato
- Extender el uso de la aplicación a todos los servicios de neonatología del país
- Registrar el sistema en el Centro Nacional de Derecho de Autor (CENDA)

Glosario de términos

Gestión de la información: La gestión de la información es el proceso de analizar y utilizar la información que se ha recabado y registrado para permitir a los administradores (de todos los niveles) tomar decisiones documentadas. La información para la gestión es la información necesaria para tomar decisiones de gestión.(Bartle 2009)

Por lo tanto, la gestión de la información implica:

- determinar la información que se precisa
- recoger y analizar la información
- registrarla y recuperarla cuando sea necesaria
- utilizarla
- divulgarla

Neonatología: La neonatología (palabra compuesta del griego *véo-*, *néo-*, "nuevo"; del latín *natus*, "nacido" y del griego *-λογία*, *-logía*, "estudio" o "conocimiento") es una rama de la pediatría dedicada al diagnóstico y tratamiento de las enfermedades del ser humano durante los primeros 28 días de vida, desde la atención médica del recién nacido en la sala de partos, el período hebdomadario (los primeros 7 días postparto), hasta los 28 días de vida del niño, posterior a los cuales se le considera como "lactante" entrando ya al campo de la pediatría propiamente dicha. Los principales pacientes de los neonatólogos son los recién nacidos que están enfermos o requieren un cuidado especial debido a que son prematuros, tienen bajo peso o sufren malformaciones.(Schaffer 2005)

Neonato: Un neonato o recién nacido es un bebé que tiene 27 días o menos desde su nacimiento, bien sea por parto o por cesárea. La definición de este período es importante porque representa una etapa muy corta de la vida; sin embargo, en ella suceden cambios muy rápidos que pueden derivar en

Glosario de términos

consecuencias importantes para el resto de la vida del recién nacido. El término se ajusta a nacidos pretérmino, a término o pasados los 9 meses del embarazo.

Durante los primeros 30 días de vida, se pueden descubrir la mayoría de los defectos congénitos y genéticos. No todas las anomalías genéticas se manifiestan por su clínica en el momento del nacimiento, pero con el manejo adecuado, se pueden descubrir, prevenir y tratar gran parte de las enfermedades del nuevo ser humano.

El recién nacido puede presentar aspectos muy diferentes atendiendo a numerosos factores propios, de su madre o del periodo gestacional. Por otra parte, existen numerosos fenómenos transicionales, derivados de la adaptación del neonato al nuevo entorno en el que se desenvuelve. Este hecho conlleva una serie de cambios, la mayoría predecibles, que establecen la norma del desarrollo y crecimiento infantil y que tienden a señalar la aparición de signos patológicos.(Schaffer 2005)

Patología: La patología —del griego, estudio (λογία, logía) del sufrimiento o daño (πάθος, pathos)— es la parte de la medicina encargada del estudio de las enfermedades en su más amplio sentido, es decir, como procesos o estados anormales de causas conocidas o desconocidas.(Schaffer 2005) . Son diversas las patologías de los recién nacidos entre las principales son las siguientes: (Schaffer 2005)

RESPIRATORIAS: Enfermedad de Membrana Hialina, Infecciones Respiratorias, Neumonía Neonatal, Atelectasias de repetición, Displasia Broncopulmonar, Síndrome de Aspiración de Meconio, Enfisema Pulmonar Intersticial, Hemorragia Pulmonar, Síndromes de Fuga Aérea, Enfermedad adenomatoidea quística, Secuestro pulmonar, Quistes broncogénicos, Hipoplasia pulmonar, Hipertensión pulmonar persistente del recién nacido.

CARDIOVASCULARES: Conducto Arterioso Persistente, Cardiopatías Congénitas, Insuficiencia cardiaca congestiva, Hipertensión Pulmonar Persistente.

Glosario de términos

GASTROINTESTINALES: Trastornos de succión y deglución, Insuficiencia Velopalatina, Atresias de esófago e intestinales, Gastrosquisis, Onfaloceles, Enterocolitis Necrosante, Colestasis, Atresias biliares, Mal rotación Intestinal, Síndrome de Intestino Corto, Alergia a Proteínas de leche y Lactosa.

NEUROLOGICAS: Encefalopatía Hipóxico-Isquémica, Asfixia Perinatal, Crisis Convulsivas Neonatales, Hemorragia Intracraneana, Hemorragia de la Matriz Germinal e Intraventricular del Prematuro, Epilepsias.

METABOLICAS: Errores innatos del metabolismo, Trastornos hidroelectrolíticos, Intoxicaciones.

HEMATOINFECCIOSAS: Sepsis Neonatal, Anemia del prematuro, Síndrome mieloproliferativo transitorio, Leucemia Neonatal, Isoinmunización Maternofetal, Enfermedad hemorrágica del recién nacido, Eritroblastosis fetal, Trombocitopenias neonatales.

ATENCIÓNES PERINATAL: Hijo de madre Diabética, Hijo de madre con Lupus Eritematoso, Hijo de madre con Miastenia Gravis, Hijo de madre Drogadicta.

Test de Apgar: El test de Apgar es un examen clínico de neonatología, tocología, empleado útilmente en ginecobstetricia y en la recepción pediátrica, en donde el médico clínico pediatra o neonatólogo certificado realiza una prueba medida en 3 estándares sobre el recién nacido para obtener una primera valoración simple (macroscópica), y clínica sobre el estado general del neonato después del parto. Mediante sencillos parámetros cardiovasculares y neurológicos puntúa de 0 a 10 el estado del neonato. Se considera un neonato sano al que puntúa por encima de 8, aunque una puntuación de 10 no es muy frecuente. Es habitual realizar esta prueba al minuto y luego a los cinco—ocasionalmente se repite a los diez minutos—de nacer, especificando el tiempo en el reporte. (Schaffer 2005)

Edad Gestacional: La edad gestacional se refiere a la edad de un embrión, un feto o un recién nacido desde el primer día de la última regla. Es un sistema estandarizado para cuantificar la progresión del embarazo y comienza aproximadamente dos semanas antes de la fertilización. De por sí, no constituye el

Glosario de términos

comienzo del embarazo, un punto que se mantiene controversial, por lo que se han diseñado sistemas alternos de conteos para dar con el comienzo del embarazo.(Schaffer 2005)

Necropsia: La palabra necropsia proviene de las voces griegas νεκρός { necrós(cadáver)} y ὄψις { òpsis(observar)}, que significa observar un cadáver.

Es el procedimiento técnico y científico de disección anatómica sistemática de un animal o persona después de su muerte para dilucidar la causa de la misma.(Schaffer 2005)

Bibliografía y Referencias Bibliográficas

- Aldarias, P. (2004). Sistema de Gestión de Base de Datos (SGBD): Postgres. España.
- Alvarez, M. A. (2002). "¿Qué es JSP?", from <http://www.desarrolloweb.com/articulos/831.php>.
- Álvarez, P. (2009). Arquitectura Cliente_Servidor.
- Álvarez, S. (2007). "Sistemas gestores de bases de datos." from <http://www.desarrolloweb.com/articulos/sistemas-gestores-bases-datos.html>.
- Anónimo (2010) "Web Server Apache."
- Bartle, P. (2009). Información para la Gestión y Gestión de la Información.
- Bauer, C. y. K., Gavin (2007). *Java Persistence with Hibernate*. s.l. : Manning.
- Carrasquel Aguilera, Y. (2007). "Sistema Integral para el Control de Operaciones".
- Castillo, J. (2004). Sistema de Control de Acceso a las Áreas del Instituto Superior Politécnico José Antonio Echeverría.
- Crane, D. (2006). AJAX in Action.
- Chen, H. y. C., R (2007). Ajax without JavaScript Framework, Potix Corporation. 1.
- Chen, H. y. C., R (2010). ZK Component Development Essentials. P. Corporation. s.1.
- Chen, H. y. C., R (2011). ZK Component Reference. P. Corporation.
- Cruz Ramírez, M., Ed. (2007). El método Delphi en las investigaciones educativas. La Habana, Editorial Académica.
- Darwin, I. (2004). Java Cookbook. O. Reilly. 1.
- Degiovannini, M. (2005) "Comparativa de frameworks Web."
- Díaz, J. C. (2007) "Comparativa herramientas reporting open source ".
- Dotres, C. (2003). La salud del niño en las Américas y en Cuba. Revista Cubana de Pediatría. 2: 122-8.

Bibliografía y Referencias Bibliográficas

- Douglas Olson, S. (2007). Ajax on Java. s.l. O. Reilly.
- Dueñas, A. M. E. (2009). Breve Historia de la Neonatología en Cuba. Revista Cubana de Pediatría: 25-8.
- Esteban, A. (2006). Tecnologías de Servidor con Java: Servlets, JavaBeans, JSP. s.l. EIDOS.
- Flanagan, D. and P. Ferguson (2002). JavaScript: The Definitive Guide
- García, R. C. (2006). Procedimiento para la valoración de sostenibilidad de un Producto Informático. Holguín.
- Greenspan, J. and B. Bulger (2001). My SQL/PHP Database Applications, M&T Books.
- Greenspan, J. a. B. B. (2001). My SQL/PHP Database Applications. M. T. Books.
- Group, M. (2010). "IIS." Retrieved 15/enero/2010, from <http://www.microsoft.com/spain/windowsserver2003/technologies/webapp/iis.aspx> .
- Inoto, R. (2006) "*Framework de Persistencia en Java.*"
- Jacobson, I. (2000). El Proceso Unificado de Desarrollo de Software. Adison Wesley.
- Jacobson, I., G. Booch (2000). El Proceso Unificado de Desarrollo de Software, Addison Wesley.
- John Worsley, J. D. (2001). PostgreSQL Práctico.
- Juárez, C., M. G. Herrera, et al. (2002) "Software libre vs software propietario. Ventajas y desventajas. ."
- Kabir, M. J. (2002). Apache Server 2 Bible. H. Minds, Inc.
- Kumbia (2007). Libro del Framework Kumbia. s.l. EIDOS.
- Leyton, E. (2004). Ingeniería de Software con UML. Auditorias de tecnologías de la información. <http://www.eduardoleyton.com/Uml.pdf>
- Manrique Chávez, J. (2010). " Lenguaje de programación PHP ", from <http://www.monografias.com/trabajos38/programacion-hp/programacion-php.shtml>
- Mato García, R. M. (1999). Diseño de Base de Datos.

Bibliografía y Referencias Bibliográficas

- Mendoza, M. A. (2004) "Metodologías de Desarrollo de Software."
- Meyer, A. (2007) "JasperReport VS. BIRT."
- Morelli, R. (2005). Java: Object-Oriented Problem Solving. P. Hall.
- Perry, B., Ed. (2006). Ajax Hacks. s.l., O' Reilly.
- PostgreSQL (2010). PostgreSQL 9.0 Documentation. T. P. G. D. Group.
- Pressman, R. (2002). Ingeniería de Software. Un enfoque práctico. s.l. P. Hall.
- Pressman, R. S. Ingeniería del Software. Un enfoque práctico.
- Ramonet, I., Ed. (2006). Cien Horas Con Fidel. La Habana, Consejo de Estado.
- Rodríguez Expósito, F. y. C. G., Rita (1997) "El Método Delphy para el procesamiento de los resultados de encuestas a expertos o usuarios en estudios de mercado y en la investigación educacional."
- Rosenberg, D. S., M. y Cope, M. C (2005). Agile Development with ICONIX Process-People, Process, and Pragmatism.
- Ruiz, F. G. (1999) "COCOMO v2. Modelo de Estimación de Costes para proyectos software."
- Rumbaugh, J. J., I. y Booch, G (2002). "El Lenguaje Unificado de Modelado. Manual de Referencia."
- Sánchez, A. G. (2010) "Sistema de Gestión de información para la Atención a Pacientes en Salas Neonatológicas (NEONAT)".
- Schaffer, A. J., Ed. (2005). Enfermedades del Recién Nacido.
- Smith, W. (2010). "Aplicaciones Web." Retrieved 22/enero/2010, from <http://www.delivering.info/soluciones/experiencia-web/aplicaciones-web-saas.php>.
- Stauble, M. (2010) "ZK Developer's Guide."
- Valle, J. G. (2005). "Definición arquitectura cliente servidor." from <http://www.monografias.com/trabajos24/arquitectura-cliente-servidor/arquitectura-cliente-servidor.shtml>
- Vegas, J. (2009) "Introducción a las Aplicaciones Web."
- Zervaas, Q. (2008). Practical Web 2.0 Applications with PHP.

ANEXOS

ANEXO 1 Entrevista preliminar realizada al cliente.

Estudio del negocio. Captura de requisitos

ENTREVISTA AL CLIENTE

Fecha: __/__/____

Nombre: _____ Cargo: _____

Puntos a analizar:

- ¿Cuáles son los principales procesos que se llevan a cabo en el servicio?
- ¿Qué deficiencias usted cree que afectan el desempeño de los mismos?
- ¿Usted tiene conocimiento sobre algún PI existente que realice la gestión de información de pacientes en servicios de neonatología del país?
- ¿Qué información de la tratada tiene mayor relevancia? ¿En qué documentos se guarda?
- ¿Qué funcionalidades le gustaría que tuviera el producto informático? ¿Por qué?
- ¿Qué otras funcionalidades a su entender debería tener el PI?

Respuestas del cliente:

ANEXO 2 Descripciones textuales de los CUN. Diagrama de Actividades y del Modelo de Objeto

Descripción textual DEL C.U.N (Verificar estado evolutivo)

Nombre Caso de Uso	Verificar estado evolutivo
Actores	Paciente
Propósito	Chequear el estado de salud del paciente con el objetivo de tomar decisiones acertadas.
Resumen	El caso de uso inicia cuando el paciente es ingresado en el servicio de neonatología del hospital pediátrico. El médico lo examina nuevamente, realiza pruebas y determina cual es el estado o reporte del paciente finalizando así el caso de uso.
FLUJO NORMAL DE EVENTOS	
Acción del Actor	Respuesta del negocio
2- El paciente queda examinado	1- El médico examina y realiza pruebas al paciente. 3- El médico determina el estado del paciente. 4- El médico actualiza en la ficha del paciente el reporte finalizando el caso de uso.
Mejoras	La automatización de este proceso agilizará la confección de la ficha del paciente.
Prioridad	CUN secundario

Tabla 9 Descripción textual DEL C.U.N (Verificar estado evolutivo)

Diagrama de Actividades del CUN Verificar estado evolutivo

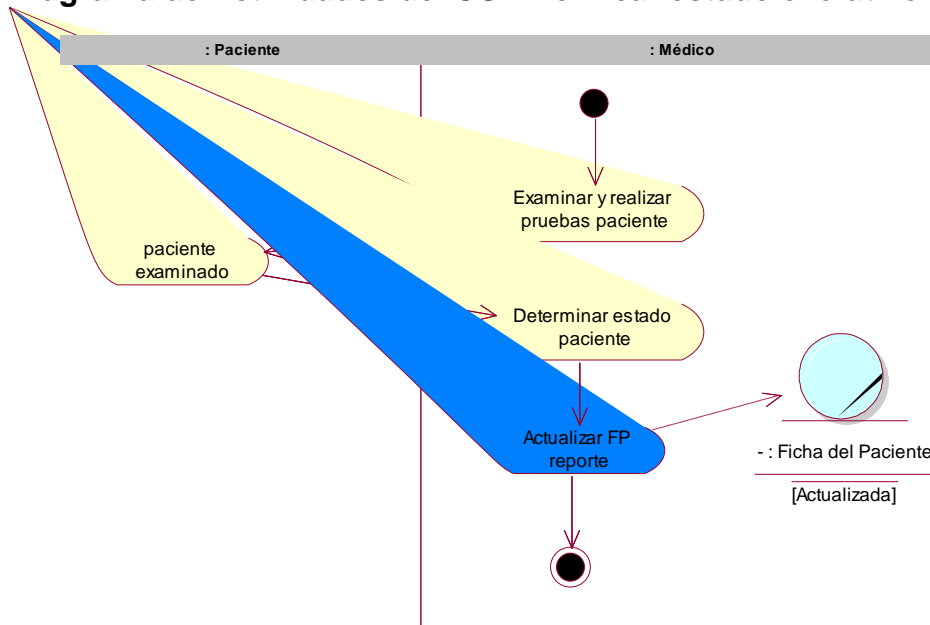


Figura 18 Diagrama de Actividades del CUN Verificar estado evolutivo

Diagrama de clases del modelo de objeto del CUN Verificar estado evolutivo

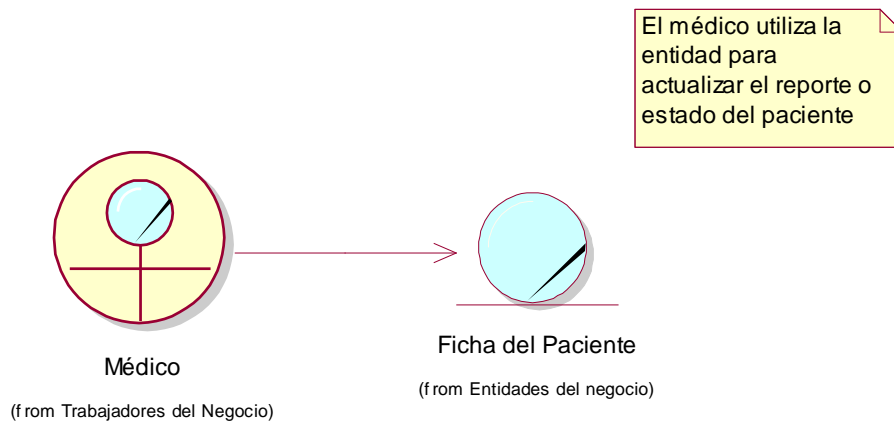


Figura 19 Diagrama de clases del modelo de objeto del CUN Verificar estado evolutivo.

Anexos

Descripción textual DEL C.U.N (Realizar análisis)

Nombre Caso de Uso	Realizar análisis
Actores	Paciente
Propósito	Encontrar las posibles causas del padecimiento, en otras llegar a descubrir el diagnóstico del paciente.
Resumen	El caso de uso inicia cuando el paciente ingresado en el servicio de neonatología del hospital pediátrico es analizado por el médico el cual realiza un grupo de análisis (cultivos, hurocultivos, Hemocultivos, entre otros) hasta que logra definir el diagnóstico del paciente finalizando así el caso de uso.
FLUJO NORMAL DE EVENTOS	
Acción del Actor	Respuesta del negocio
3- El paciente queda diagnosticado	1- El médico realiza un conjunto de análisis. 2- Determina el diagnóstico del paciente. 4- El médico actualiza en la ficha del paciente el diagnóstico (inicial o final) finalizando así el caso de uso.
Mejoras	La automatización de este proceso agilizará la confección de la ficha del paciente.
Prioridad	CUN secundario

Tabla 10 Descripción textual DEL C.U.N (Realizar análisis)

Diagrama de Actividades del CUN Realizar análisis

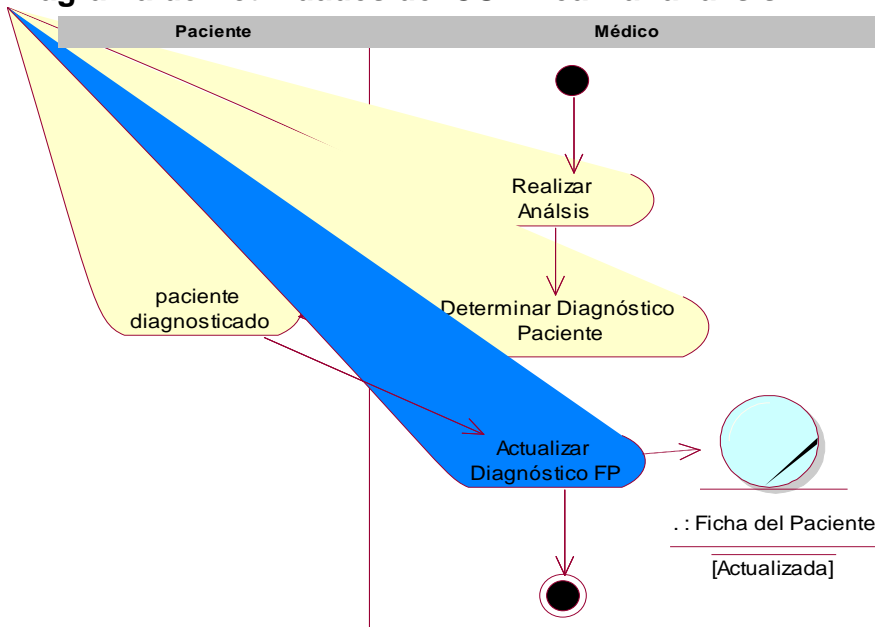


Figura 20 Diagrama de Actividades del CUN Realizar análisis

Diagrama de clase del modelo de objeto del CUN Realizar análisis

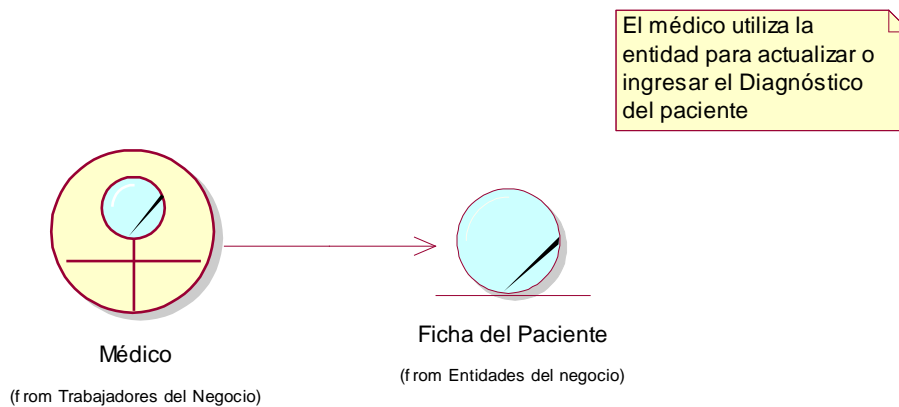


Figura 21 Diagrama de clase del modelo de objeto del CUN Realizar análisis

Descripción textual DEL C.U.N (Registrar fallecimiento paciente)

Nombre Caso de Uso	Registrar fallecimiento paciente
Actores	Paciente
Propósito	Registrar las posibles causas del fallecimiento del paciente con el objetivo de conocer el motivo que originó la muerte.
Resumen	El caso de uso inicia cuando el paciente ingresado en el servicio de

Anexos

	<p>neonatología del hospital pediátrico es examinado para chequear su estado evolutivo, el cual empeora gradualmente sin importar el tratamiento asignado debido a las patologías que presenta, hasta que finalmente fallece y el médico actualiza en la ficha del paciente la contrareferencia del mismo así como los datos de la necropsia, finalizando así el caso de uso.</p>
FLUJO NORMAL DE EVENTOS	
Acción del Actor	Respuesta del negocio
<p>3- El paciente no responde a los tratamientos nuevos y empeora cada vez más hasta que fallece.</p>	<p>1- El médico examina y chequea el estado evolutivo del paciente y nota el deterioro del mismo. 2- El médico establece nuevos tratamientos. 4- se le realiza la necropsia al paciente y el médico registra en la ficha del paciente, los datos de la misma así como actualiza la contrareferencia y la fecha de egreso, finalizando así el caso de uso.</p>
Mejoras	<p>La automatización de este proceso agilizará la confección de la ficha del paciente.</p>
Prioridad	CUN secundario

Tabla 11 Descripción textual DEL C.U.N (Registrar fallecimiento paciente)

Diagrama de actividades del CUN Registrar fallecimiento paciente

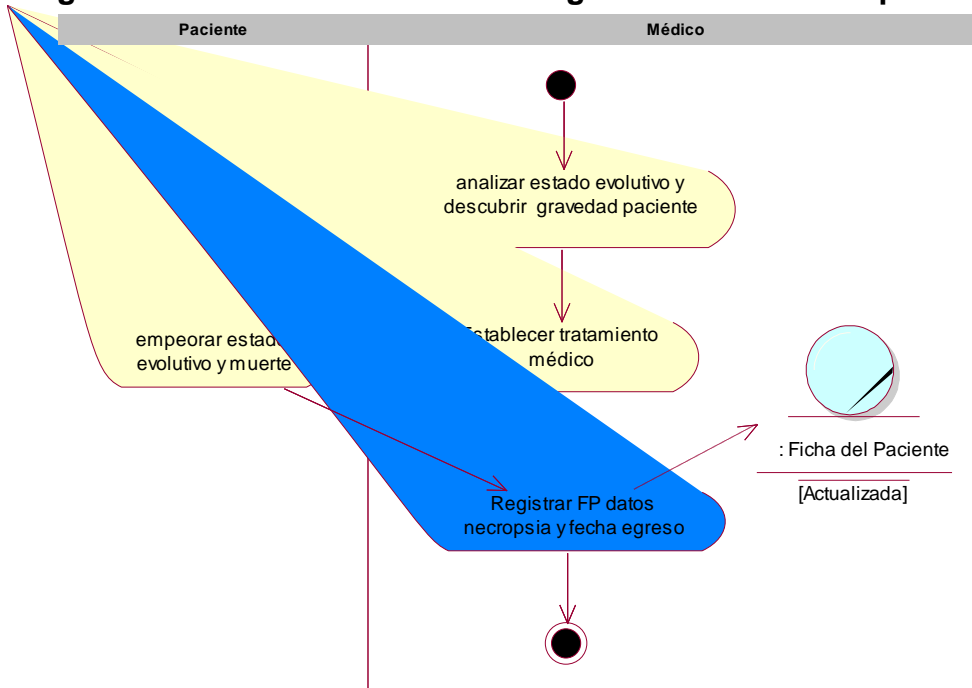


Figura 22 Diagrama de actividades del CUN Registrar fallecimiento paciente

Diagrama de clase del modelo de objeto del CUN Registrar fallecimiento paciente

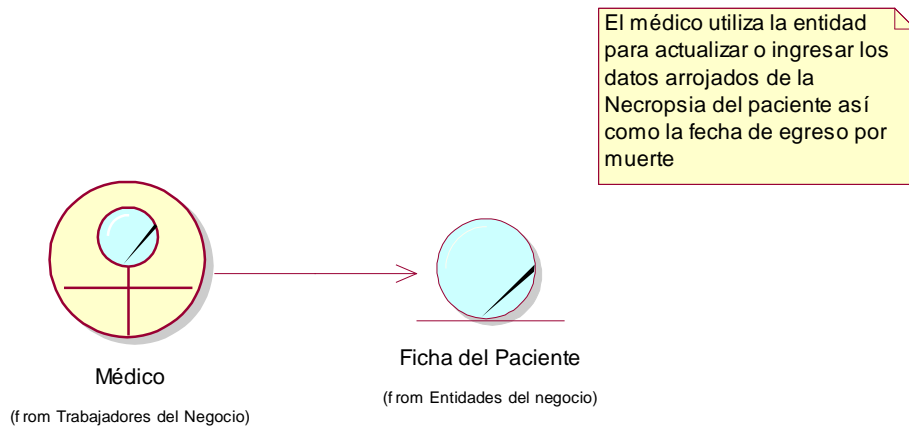


Figura 23 Diagrama de clase del modelo de objeto del CUN Registrar fallecimiento paciente

Descripción textual DEL C.U.N (Realizar traslado paciente)

Nombre Caso de Uso		Realizar traslado paciente
Actores	Paciente	
Propósito	Trasladar al paciente de la sala ya sea	

Anexos

	<p>por cumplir mayoría de edad (mayor a 28 días de nacido), estado evolutivo suficientemente satisfactorio como para retornar a su área de salud correspondiente (en caso que no sea de Holguín), o ambos casos respectivamente.</p>
Resumen	<p>El caso de uso inicia cuando el paciente ingresado en el servicio de neonatología de carácter quirúrgico es examinado para chequear su estado evolutivo, el cual empeora o mejora gradualmente de acuerdo al tratamiento asignado, si el estado evolutivo del paciente que tiene alta quirúrgica mejora, se traslada a su área de salud (procedencia) correspondiente (paciente mayor o no a los 28 días), o para otra sala a terminar su tratamiento para pacientes de la provincia Holguín mayores a 28 días de nacido, terminando así el caso de uso. Si su estado evolutivo empeora (sin importar edad o procedencia se mantiene en el servicio recibiendo tratamiento médico) finalizando así el caso de uso.</p>
FLUJO NORMAL DE EVENTOS	
Acción del Actor	Respuesta del negocio
	1- El médico examina y chequea el estado evolutivo del paciente y la edad

Anexos

<p>3- El paciente se mantiene en la sala siendo tratado. Finalizando el caso de uso.</p> <p>5- El paciente es trasladado hacia otra sala o a su área de salud de origen.</p>	<p>del mismo, nota la mejoría o el deterioro del estado de salud y analiza si necesita ser trasladarlo o no.</p> <p>1.1 Si no lo va a trasladar pasa al paso 2.</p> <p>1.2 Si lo traslada pasa al paso 4.</p> <p>2- El médico mantiene al paciente bajo tratamiento médico en el servicio de neonatología.</p> <p>4- El médico traslada al paciente de alta quirúrgica (Holguín) hacia otra sala a recibir tratamiento médico de recuperación o hacia su área de salud correspondiente.</p> <p>6- El médico actualiza en la ficha del paciente la contrareferencia del mismo y la fecha de egreso, finalizando el caso de uso.</p>
<p>Mejoras</p>	<p>La automatización de este proceso agilizará la confección de la ficha del paciente.</p>
<p>Prioridad</p>	<p>CUN secundario</p>

Tabla 12 Descripción textual DEL C.U.N (Realizar traslado paciente)

Diagrama de actividades del CUN Realizar traslado paciente

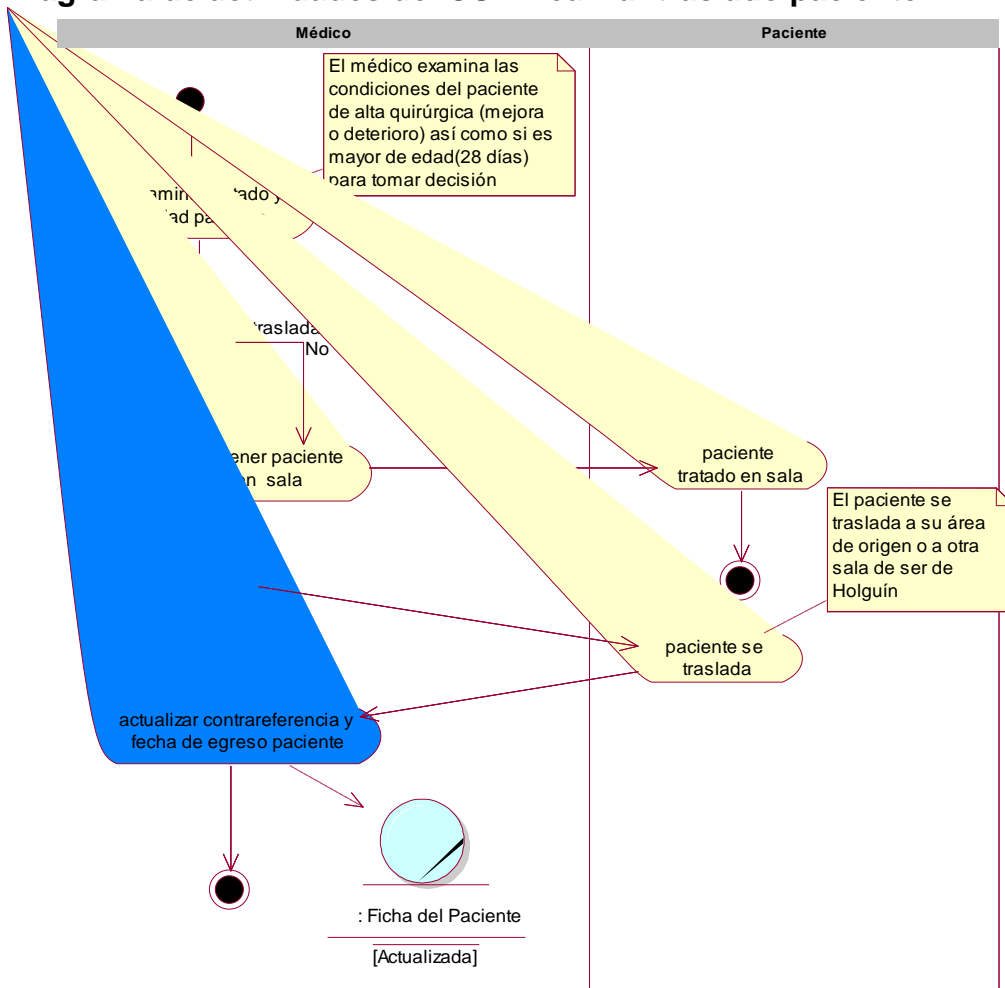


Figura 24 Diagrama de actividades del CUN Realizar traslado paciente

Diagrama de clase del modelo de objeto del CUN Realizar traslado paciente

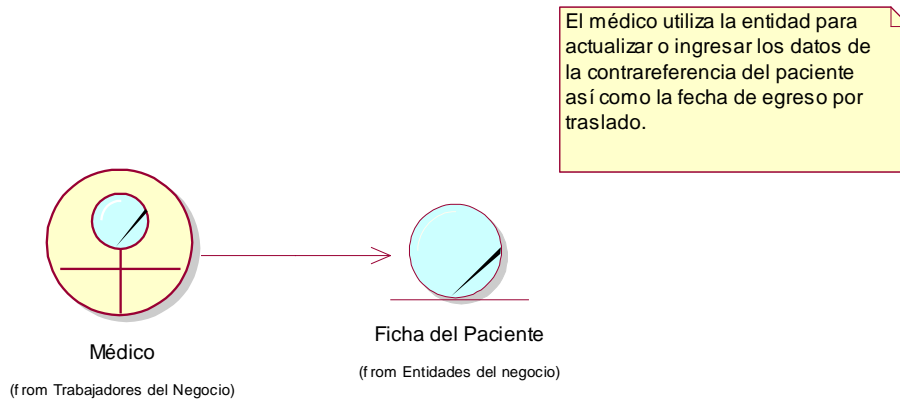


Figura 25 Diagrama de clase del modelo de objeto del CUN Realizar traslado paciente

Anexos

Descripción textual DEL C.U.N (Establecer alta paciente)

Nombre Caso de Uso	Establecer alta paciente
Actores	Paciente
Propósito	Darle de baja o salida del centro definitivo al paciente.
Resumen	El caso de uso inicia cuando el paciente ingresado en el servicio de neonatología del hospital pediátrico es examinado por el médico el cual chequea su estado evolutivo, realiza pruebas de rutina y determina que el paciente posee un estado de salud libre de riesgos en su totalidad, procediendo así a darle la baja del servicio, finalizando así el caso de uso.
FLUJO NORMAL DE EVENTOS	
Acción del Actor	Respuesta del negocio
3- El paciente queda examinado y retorna a su casa.	1- El médico examina nuevamente al paciente y realiza pruebas de rutina. 2- El médico determina que el estado del paciente es satisfactorio y libre de riesgos y establece el alta del mismo. 4- El médico actualiza en la ficha del paciente la contrareferencia del mismo así como su fecha de egreso del servicio, finalizando el caso de uso.
Mejoras	La automatización de este proceso agilizará la confección de la ficha del paciente.
Prioridad	CUN secundario

Tabla 13 Descripción textual DEL C.U.N (Establecer alta paciente)

Diagrama de Actividades del CUN Establecer alta paciente

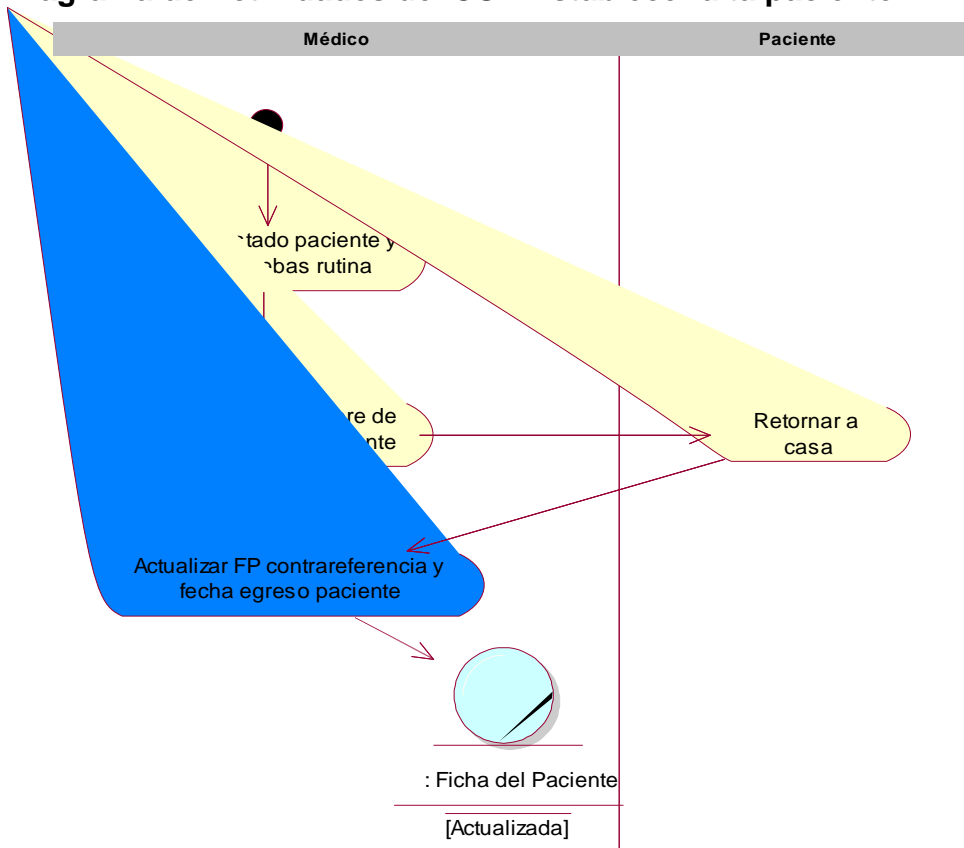


Figura 26 Diagrama de Actividades del CUN Establecer alta paciente

Diagrama de clase del modelo de objeto del CUN Establecer alta paciente

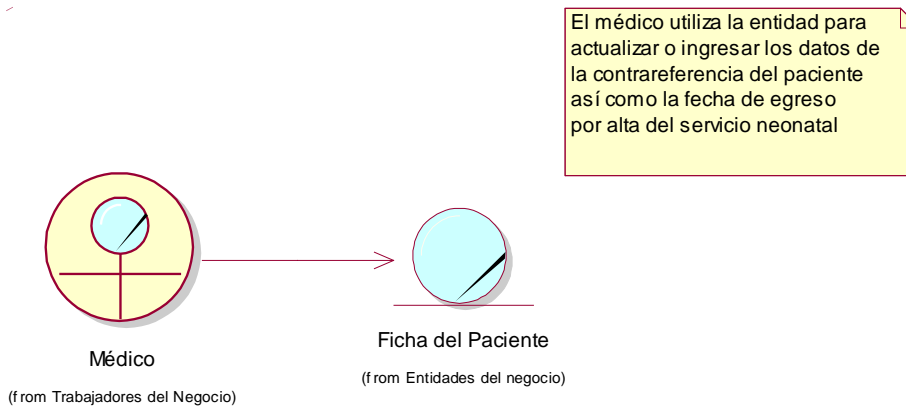


Figura 27 Diagrama de clase del modelo de objeto del CUN Establecer alta paciente

Descripción textual DEL C.U.N (Registrar infección nosocomial)

Nombre Caso de Uso	Registrar infección nosocomial
Actores	Paciente
Propósito	Conocer aquellas infecciones adquiridas en el servicio de

Anexos

	neonatología, así como los factores de riesgo que la propiciaron y el tipo de infección que fue la causa de la patología, con el objetivo de prevenirlas.
Resumen	El caso de uso inicia cuando el paciente ingresado en el servicio de neonatología es chequeado por el médico nuevamente, y determina que el estado o reporte del paciente, se ha deteriorado producto a factores diferentes a aquellos por los que ingresó, realiza pruebas y nota que ha adquirido una nueva patología la cual debe plasmar en el registro de infecciones nosocomiales, finalizando así el caso de uso.
FLUJO NORMAL DE EVENTOS	
Acción del Actor	Respuesta del negocio
2- El paciente queda examinado	<p>1- El médico examina nuevamente al paciente y nota el deterioro de su estado.</p> <p>3- El médico realiza pruebas.</p> <p>4- Determina que estado del paciente se deteriora producto a factores diferentes a los que ingresó, es decir que el mismo ha adquirido una infección en el servicio.</p> <p>5- El médico actualiza en la ficha del paciente registro de infecciones nosocomiales finalizando el caso de</p>

Anexos

	uso.
Mejoras	La automatización de este proceso agilizará la confección de la ficha del paciente.
Prioridad	CUN secundario

Tabla 14 Descripción textual DEL C.U.N (Registrar infección nosocomial)

Diagrama de Actividades del CUN Registrar infección nosocomial

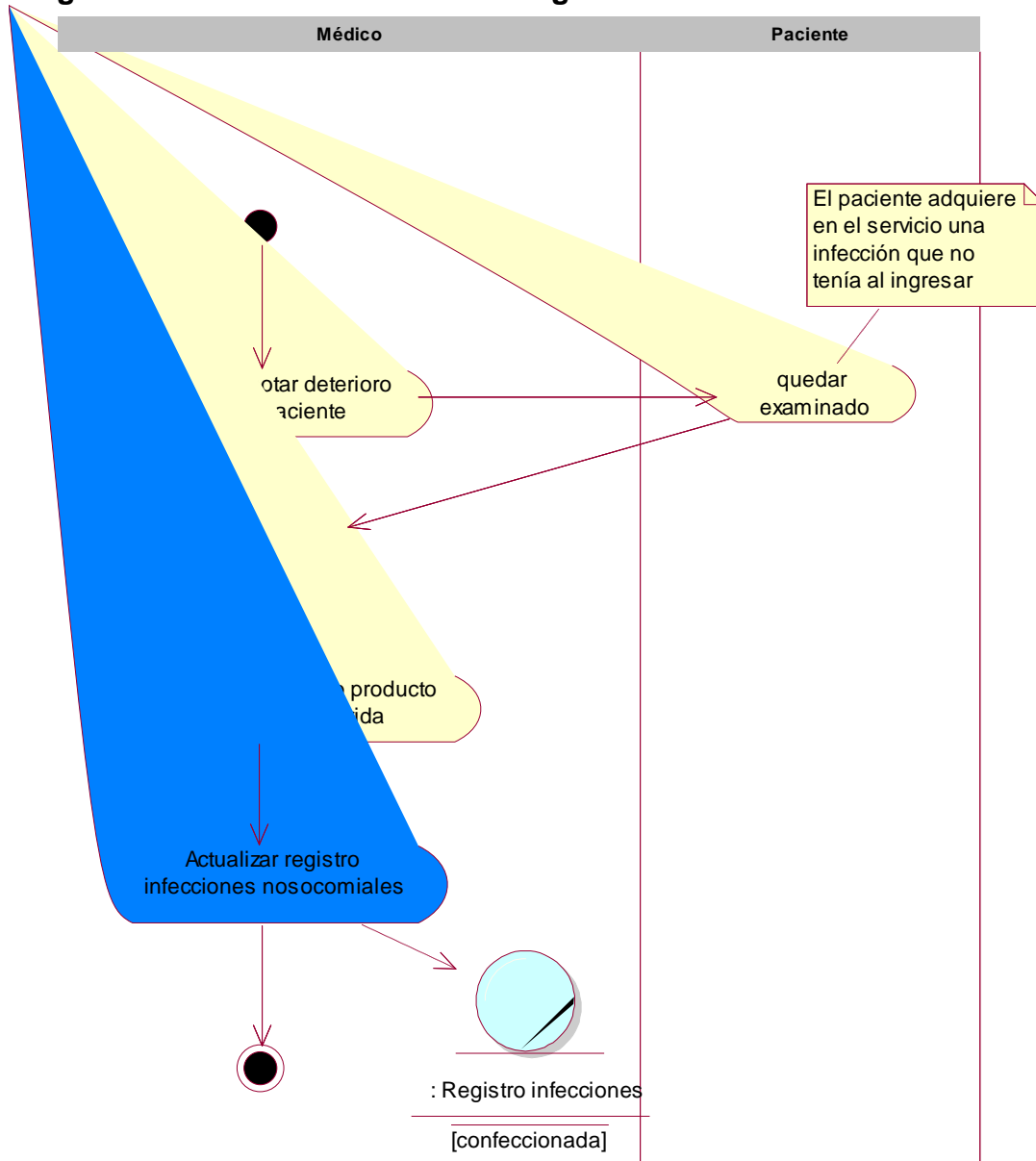


Figura 28 Diagrama de Actividades del CUN Registrar infección nosocomial

Diagrama de clase del modelo de objeto del CUN Registrar infección nosocomial

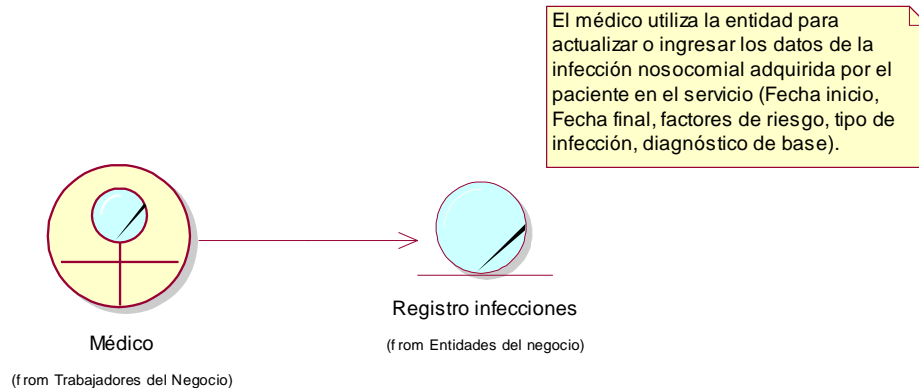


Figura 29 Diagrama de clase del modelo de objeto del CUN Registrar infección nosocomial

Descripción textual DEL C.U.N (Registrar ventilación paciente)

Nombre Caso de Uso		Registrar ventilación paciente
Actores	Paciente	
Propósito	Conocer aquellos datos relacionados con el proceso de ventilación del paciente.	
Resumen	El caso de uso inicia cuando el paciente ingresado en el servicio de neonatología es chequeado por el médico nuevamente, y determina que el estado o reporte del paciente, se ha deteriorado producto a factores u anomalías respiratorias, para contrarrestar esto ventila al paciente, y luego plasma la información asociada a este proceso en el registro de ventilados, finalizando así el caso de uso.	
FLUJO NORMAL DE EVENTOS		
Acción del Actor	Respuesta del negocio	

Anexos

2- El paciente queda examinado. 6- El paciente queda ventilándose.	1- El médico examina nuevamente al paciente y nota el deterioro de su estado. 3- El médico realiza pruebas. 4- Determina que estado del paciente se deteriora producto a factores u anomalías respiratorias. 5- El médico procede a ventilar al paciente. 7- El médico actualiza en la ficha del paciente el registro de ventilado finalizando el caso de uso.
Mejoras	La automatización de este proceso agilizará la confección de la ficha del paciente.
Prioridad	CUN secundario

Tabla 15 Descripción textual DEL C.U.N (Registrar ventilación paciente)

Diagrama de Actividades del CUN Registrar Ventilación paciente

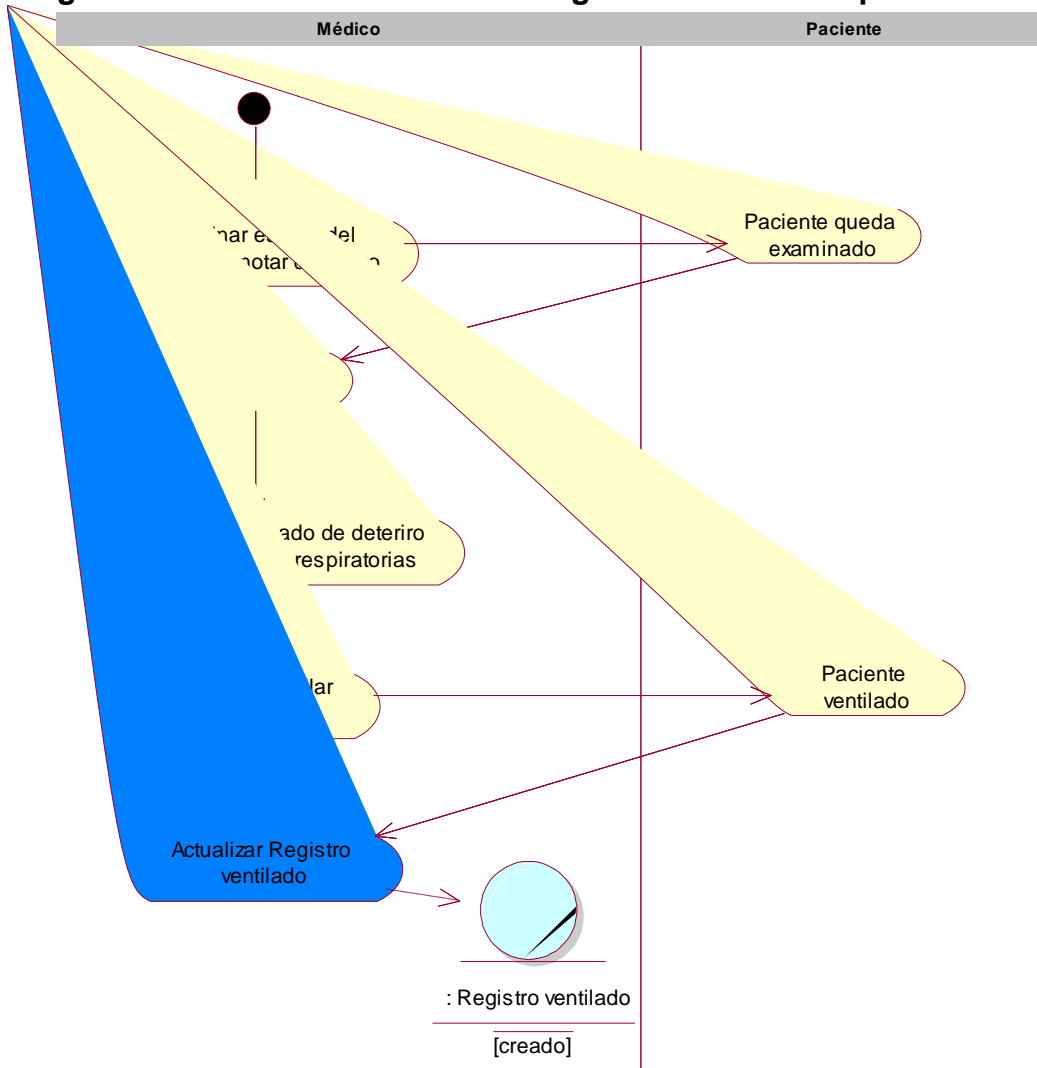


Figura 30 Diagrama de Actividades del CUN Registrar Ventilación paciente
Diagrama de clase del modelo de objeto del CUN Registrar ventilación paciente

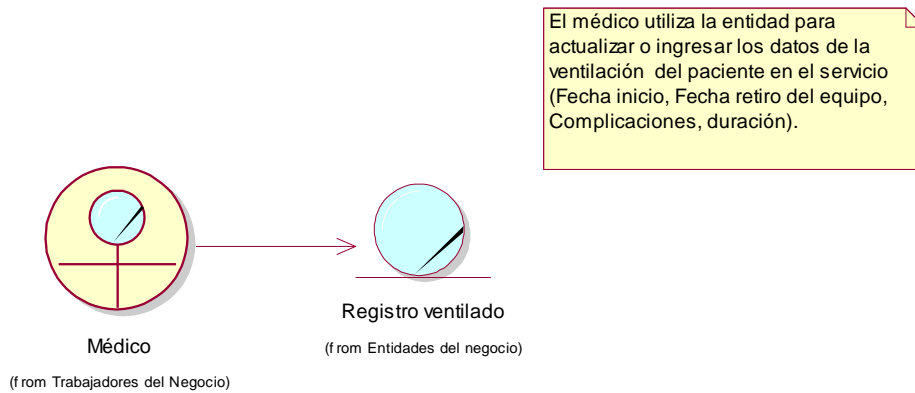


Figura 31 Diagrama de clase del modelo de objeto del CUN Registrar ventilación paciente

ANEXO 3 Descripciones textuales en formato real y expandido de los CUS

Descripción Textual en formato real y expandido del Caso de uso del sistema Gestionar Usuarios del Sistema

Caso de uso:	<i>Gestionar Usuarios del Sistema</i>
Actor(es):	Administrador (inicia)
Propósito:	Mantener actualizados los usuarios del sistema.
Resumen:	El caso de uso se inicia cuando el Administrador requiere registrar un nuevo usuario del sistema, modificar la cuenta del mismo o eliminarla. El sistema interactúa siempre que sea necesario con la base de datos. De acuerdo a sus requerimientos inserta la cuenta, modifica o elimina, graba y el registro de la cuenta del usuario queda actualizado, finalizando el caso de uso.
Tipo:	Real y Expandido.
Referencias:	R3: R3.1, R3.2, R3.3
Precondiciones:	El Administrador ha ingresado al sistema y se encuentra en las opciones del menú principal.
Pantalla Administrar Usuarios	

Anexos

The screenshot displays the NEOGEST web application interface. At the top, the logo and name 'NEOGEST' are visible, along with the text 'HOSPITAL PEDIÁTRICO UNIVERSITARIO DE HOLGUÍN' and 'Octavio de la Concepción de la Pedraja'. A navigation menu on the left includes options like 'Gestionar Ficha Paciente', 'Gestionar Procedencia', 'Gestionar Patologías', 'Gestionar Factores de Riesgo', 'Gestionar Tipo de Infecciones', 'Buscar Datos Pacientes', 'Vizualizar Reportes', and 'Administrar Usuarios'. The 'Administrar Usuarios' option is selected, leading to a page titled 'Administrar Usuarios'. This page features a table titled 'Listado de Usuarios' with two columns: 'Usuario' and 'Rol'. The table contains the following data:

Usuario	Rol
admin	Administrador
jose	Administrador
Tania	Jefa
mpilar	medico
invitado	invitado
oreyesp	medico

Below the table, the user is identified as 'jose | Administrador'.

Flujo Normal de Eventos

Acción del Actor	Respuesta del Sistema
1. El Administrador selecciona la opción "Administrar Usuarios" en la pantalla de inicio del sistema.	2. El sistema muestra la pantalla "Administrar Usuarios" con el listado de usuarios existentes en el sistema.
1. El Administrador requiere insertar una cuenta de usuario nueva y oprime el botón en la barra Añadir.	2. El sistema muestra el Panel "Nuevo Usuario".



<p>3. El Administrador llena la información correspondiente a los indicadores(Usuario, Clave, Repetir Clave, Rol) y oprime el botón Agregar.</p>	<p>4. El sistema valida la información y la almacena en la BD si está correcta, le muestra un cartel al Administrador de inserción satisfactoria.</p>
<p>5. El Administrador selecciona otra opción en el menú principal o cierra la aplicación</p>	<p>6. El sistema cambia de pantalla en base a la opción elegida o no aparece.</p>
<p>Poscondiciones</p>	<p>Queda actualizada la cuenta de los usuarios.</p>
<p>Cursos Alternos</p>	<p>Antes de realizar cualquier acción el sistema verifica si el servidor de BD está apagado o existe algún problema con la conexión, de ser así entonces muestra un mensaje de error. Antes de insertar los datos en el registro el sistema los valida, en caso de errores en los datos, se muestra un mensaje de error de acuerdo con el dato erróneo o faltante.</p>



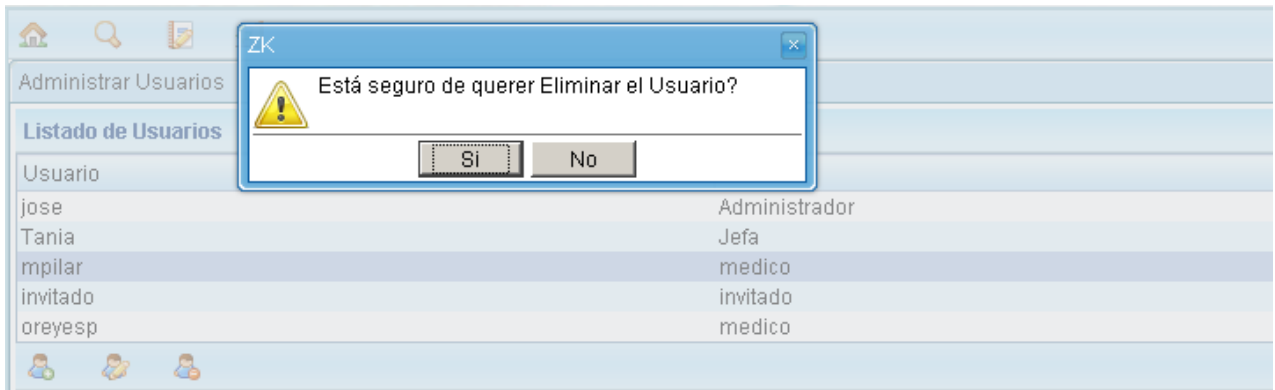
Flujo Normal de Eventos

<p>1. El Administrador requiere modificar la cuenta de un usuario y selecciona el mismo en la lista, luego presiona el botón de la barra Modificar.</p>	<p>2. El sistema le muestra en el Panel “Modificar Usuario” todos los campos con los datos correspondientes al usuario seleccionado, los cuales pueden ser editados.</p>
<p>3. El Administrador cambia los datos que desee y oprime el botón Actualizar.</p>	<p>4. El sistema modifica la información, valida que no haya errores al introducirla, la almacena en la BD, le muestra un mensaje de actualización satisfactoria y muestra nuevamente el listado de usuarios.</p>
<p>5. El Administrador selecciona otra opción en el menú principal o cierra la aplicación</p>	<p>6. El sistema cambia de pantalla en base a la opción elegida o no aparece.</p>
<p>Poscondiciones</p>	<p>Queda actualizada la cuenta de los usuarios.</p>
<p>Cursos Alternos</p>	<p>Antes de realizar cualquier acción el sistema verifica si el servidor de BD está</p>

Anexos

	<p>apagado o existe algún problema con la conexión, de ser así entonces muestra un mensaje de error. Antes de Modificar los datos en el registro el sistema los valida, en caso de errores en los datos, se muestra un mensaje de error de acuerdo con el dato erróneo o faltante.</p>
--	--

Pantalla Eliminar Usuario



Flujo Normal de Eventos

1. El Administrador desea eliminar un usuario existente en el sistema, y selecciona el mismo de la lista, luego pulsa el botón de la barra Eliminar .	2. El sistema muestra un mensaje indicándole al Administrador si está seguro de querer eliminar el usuario seleccionado.
3. El administrador pulsa el botón del mensaje de confirmación Si .	4. El sistema elimina de la Base de Datos el registro del usuario y muestra un mensaje indicándole al Administrador que la acción se realizó satisfactoriamente.
5. El administrador pulsa el botón del mensaje de confirmación No .	6. El sistema vuelve al listado de usuarios del sistema y no elimina al usuario seleccionado.
7. El Administrador selecciona otra opción en el menú principal o cierra la aplicación	8. El sistema cambia de pantalla en base a la opción elegida o no aparece.

Anexos

Poscondiciones	Queda eliminada la cuenta del usuario.
Cursos Alternos	Antes de realizar cualquier acción el sistema verifica si el servidor de BD está apagado o existe algún problema con la conexión, de ser así entonces muestra un mensaje de error. Antes de eliminar los datos en el registro el sistema valida que se halla seleccionado un usuario, en caso contrario muestra un mensaje de error.

Tabla 16 Descripción Textual en formato real y expandido del Caso de uso del sistema Gestionar Usuarios del Sistema

Descripción Textual en formato real y expandido del CUS Visualizar Reportes

Caso de uso:	Visualizar Reportes
Actor(es):	Jefa Servicio (inicia)
Propósito:	Crear reportes por determinados criterios.
Resumen:	El caso de uso se inicia cuando la Jefa de Servicio requiere crear un reporte a fin de mes por determinados criterios a tener en cuenta, los mismos se sitúan en un rango de tiempo y se muestran en una tabla (rango actual) finalizando el caso de uso.
Tipo:	Real y Expandido.
Referencias:	R14: R: 14.1, R:14.2, R: 14.3, R: 14.4, R15, R16
Precondiciones:	La Jefa de Servicio ha ingresado al sistema y se encuentra en las opciones del menú principal.

Pantalla Visualizar Reportes

Parámetros a Medir

Criterios	Rango
<input checked="" type="checkbox"/> Ingresados	Fecha inicial: 01/01/2010
<input checked="" type="checkbox"/> Fallecidos	Fecha final: 17/03/2010
<input checked="" type="checkbox"/> Graves	
<input checked="" type="checkbox"/> Ventilados	
<input type="button" value="Mostrar"/>	

Resultados

Criterio	Clínicos	Quirúrgicos	Total
Ingresados	1	1	2
Fallecidos	0	0	0
Graves	0	0	0
Ventilados	1	0	1

Rango a comparar

Fecha inicial: 01/02/2011
Fecha final: 01/02/2011

Flujo Normal de Eventos

1. La Jefa de Servicio selecciona Visualizar Reportes en el menú principal del sistema.	2. El sistema muestra la pantalla Reportes Neonatología.
3. La Jefa de Servicio chequea todos los parámetros a medir en el reporte y presiona el botón Mostar .	4. El sistema muestra los datos coincidentes del mes actual en una tabla en el Panel de Resultados.
5. La jefa de Servicio selecciona otra opción o cierra la aplicación	6. El sistema realiza la acción, cambia de pantalla en base a la opción elegida o no aparece.
Poscondiciones	Se visualiza el reporte asociado a los criterios.
Cursos Alternos	Antes de realizar cualquier acción el sistema verifica si el servidor de BD está apagado o existe algún problema con la conexión, de ser así entonces muestra un





Anexos

	mensaje de error. Antes de mostrar los datos, el sistema verifica que no existan errores o ausencia de los mismos, en caso contrario muestra un mensaje de error.
--	---

Tabla 17 Descripción Textual en formato real y expandido del CUS Visualizar Reportes

Descripción Textual en formato real y expandido del CUS Mostrar Gráficos de Datos Comparativos por Rangos de Fecha

Caso de uso:	Mostrar Gráficos de Datos Comparativos por Rangos de Fecha
Actor(es):	Jefa Servicio
Propósito:	Graficar datos de distintos criterios a reportar con el objetivo de comparar el comportamiento de los pacientes en relación a esos criterios en rangos de fecha diferentes.
Resumen:	El caso de uso se inicia cuando la Jefa de Servicio requiere graficar el reporte de fin de mes por determinados criterios a tener en cuenta, los mismos se sitúan en un rango de tiempo (actual) y a comparar, finalmente se visualizan las gráficas finalizando el caso de uso.
Tipo:	Real y Expandido.
Referencias:	R15
Precondiciones:	La Jefa de Servicio ha ingresado al sistema y se encuentra en la pantalla Visualizar Reportes.

Pantalla de Reportes de Neonatología			
Reportes Neonatología			
Parámetros a Medir			
Criterios <input checked="" type="checkbox"/> Ingresados <input checked="" type="checkbox"/> Fallecidos <input checked="" type="checkbox"/> Graves <input checked="" type="checkbox"/> Ventilados <input type="button" value="Mostrar"/>		Rango Fecha inicial: 01/04/2011  Fecha final: 30/04/2011 	
Resultados			
Criterio	Clínicos	Quirúrgicos	Total
Rango a comparar			
Fecha inicial: 01/02/2010  Fecha final: 28/02/2010 			
<input type="button" value="Graficar"/>   			
Flujo Normal de Eventos			
1. La Jefa de Servicio luego de escoger todos los criterios y los rangos de fecha actuales y a comparar presiona el botón Graficar .		2. El sistema muestra los gráficos relacionados con los criterios escogidos donde se puede apreciar el comportamiento de los mismos en distintos rangos de fecha como se muestra a continuación.	
Reporte Gráfico			



3. La jefa de Servicio selecciona otra opción o cierra la aplicación	4. El sistema realiza la acción, cambia de pantalla en base a la opción elegida o no aparece.
Poscondiciones	Se visualiza el gráfico del reporte asociado a los criterios.
Cursos Alternos	Antes de realizar cualquier acción el sistema verifica si el servidor de BD está apagado o existe algún problema con la conexión, de ser así entonces muestra un mensaje de error. Antes de mostrar los datos, el sistema verifica que no existan errores o ausencia de los mismos, en caso contrario muestra un mensaje de error.

Tabla 18 Descripción Textual en formato real y expandido del CUS Mostrar Gráficos de Datos Comparativos por Rangos de Fecha

Descripción textual del CUS Validar Usuario

Nombre del caso de uso	Validar Usuario
Actores	Todos excepto Reloj.
Propósito	Validar que el usuario logueado tenga permisos para realizar las acciones que desea.
Resumen	El caso de uso se inicia cuando el Usuario se loguea y el

Anexos

	sistema verifica en los datos de la sesión creada el rol que tiene, de acuerdo con el mismo el sistema valida la información y le permite ver y hacer solo aquellas acciones asociadas al rol del usuario autenticado finalizando así el caso de uso.
Referencias	R2
Precondiciones	El Usuario debe estar autenticado satisfactoriamente. Debe haberse creado la sesión del mismo.
Poscondiciones	El usuario queda validado permitiéndole realizar sus acciones características.
Requisitos especiales	-

Tabla 19 Descripción textual del CUS Validar Usuario

Descripción textual del CUS Autenticar Usuario

Nombre del caso de uso	Autenticar Usuario
Actores	Usuario (inicia).
Propósito	Permitir que el usuario se pueda loguear en el sistema.
Resumen	El caso de uso se inicia cuando el Usuario se loguea y el sistema verifica si los datos nombre de usuario y contraseña se encuentran en la base de datos, crea una sesión con el nombre de usuario y el rol finalizando así el caso de uso.
Referencias	R1
Precondiciones	El Usuario debe tener una cuenta en el sistema, debe escribir correctamente los datos nombre de usuario y contraseña.
Poscondiciones	El usuario queda autenticado.
Requisitos especiales	-

Anexos

Tabla 20 Descripción textual del CUS Autenticar Usuario
Descripción textual del CUS Realizar Backup automático y salva cruzada de la BD

Nombre del caso de uso	Realizar Backup automático y salva cruzada de la BD.
Actores	Reloj. (Inicia)
Propósito	Permite realizar automáticamente todos los días salvadas a la Base de Datos del sistema, y hacer copias de estas en otras computadoras.
Resumen	El caso de uso inicia determinados días a una hora señalada. Realiza el Backup de la Base de Datos automáticamente y hace una copia de la salva en otras máquinas, una vez culminado este proceso el caso de uso finaliza.
Referencias	R4.
Precondiciones	El reloj del sistema operativo debe funcionar correctamente.
Poscondiciones	La BD queda resguardada.
Requisitos especiales	Este caso de uso depende del reloj del sistema operativo.

Tabla 21 Descripción textual del CUS Realizar Backup automático y salva cruzada de la BD

ANEXO 4 Tablas y Fórmulas del COCOMO. Resultados

Estudio de factibilidad.

Salidas externas(EO)

Nombre de las salidas externas	Cantidad de ficheros	Cantidad de elementos de datos	Clasificación
Modificar FP	4	23	Alto
Búsqueda por parámetros	4	14	Alto
Modificar Usuario	2	3	bajo
Modificar Procedencia	1	1	bajo
Modificar Patología	1	1	bajo
Modificar Factor Riesgo	1	1	bajo
Modificar Tipo infecciones	1	1	bajo

Tabla 22 Salidas externas(EO)

Entradas externas(EI)

Nombre de las entradas externas	Cantidad de ficheros	Cantidad de elementos de datos	Clasificación
Adicionar FP	4	23	Alto
Adicionar Usuarios	2	3	Bajo
Adicionar Procedencia	1	1	Bajo
Adicionar Patología	1	1	Bajo
Adicionar Factor Riesgo	1	1	Bajo
Adicionar Tipo Infecciones	1	1	Bajo

Tabla 23 Entradas externas(EI)

Ficheros Lógicos Internos

Ficheros internos	Cantidad de ficheros	Cantidad de elementos de datos	Clasificación
tbpaciente	1	24	Bajo
tbpaciente_patologia	1	2	Bajo

Anexos

tbprocedencia	1	2	Bajo
tbcontrareferencia	1	2	Bajo
tbreporte	1	2	Bajo
tbpatologia	1	2	Bajo
tbventilado	1	7	Bajo
tbinfeccion_nosocomial	1	5	Bajo
tbfactor_riesgo	1	2	Bajo
tbtipo_infeccion	1	2	Bajo
tbinfeccion_factor_riesgo	1	2	Bajo
tbinfeccion_tipo_infeccion	1	2	Bajo
tbtrol	1	2	Bajo
tbusuario	1	4	Bajo

Tabla 24 Ficheros Lógicos Internos

El sistema (temporalmente) no posee peticiones ya que todas las consultas o salidas son con elementos de filtrajes.

El sistema no posee ficheros lógicos externos ya que no tiene asociada ninguna tabla de una Base de Datos externa.

Puntos de función desajustados

Elementos	Bajos	x Peso	Medios	x Peso	Altos	x Peso	Subtotal
ILF	14	7		10	0	15	98
ELF	0	5		7	0	10	0
EI	5	3		4	1	6	21
EO	5	4		5	2	7	34
EQ	0	3		4	0	6	0
						Total	153

Tabla 25 Puntos de función desajustados

Características

Características	Valor
Puntos de función desajustados	153

Anexos

Lenguajes % de utilización	Java (60%), SQL (40%)
Ratio x lenguaje	Java:60, SQL:39
Instrucciones fuente x lenguaje	Java: 5508 SQL: 2386.8
Total de instrucciones fuente(KSLOC)	7.8948

Tabla 26 Características

El esfuerzo asociado al desarrollo un sistema se denomina PM. Para calcular PM se utilizan los multiplicadores de esfuerzo, los factores de escala y una serie de valores calibrados (A, B, C, D).

Tabla Multiplicadores de esfuerzo

Multiplicador	Descripción	Valor
RELY	La confiabilidad es nominal	1
DATA	El tamaño de la Base de Datos es bajo	0.83
CPLX	La complejidad del producto es simple	0.60
RUSE	La reutilización es nominal a través del proyecto	1
DOCU	La necesidad de documentación es poca	0.60
STOR	El almacenamiento es nominal	1
PVOL	La plataforma es estable.	1
TIME	El tiempo de ejecución es nominal	1

Tabla 27 Tabla Multiplicadores de esfuerzo

Tabla de Factores de escala

Factores	Descripción	Valor
PREC	Muy diferente	3.72
FLEX	Muy rigurosa	5.07
RESL	Se identifican algunos de los riesgos críticos y se establecen hitos para resolverlos. Pueden presentarse algunos riesgos.	4.24
TEAM	Interacciones básicas cooperativas, objetivos y	3.29

	culturas de accionistas básicamente consistentes.	
PMAT	Relación con el proceso de madurez del software. Nivel 1 superior.	6.24

Tabla 28 Tabla de Factores de escala

Tabla de Valores calibrados

Constante	Valor
A	2.94
B	0.91
C	3.67
D	0.28

Tabla 29 Tabla de Valores calibrados

$$PM = A * Size^E \prod_{i=1}^5 EM_i = 3.37$$

$$PM = 2.94 * (7.8948)^{1.133} * 0.2988$$

$$PM = 2.94 * (10.3275) * 0.2988$$

$$PM = 9.0724$$

E depende de los factores de escala (SF):

$$E = B + 0.01 * \sum_{j=1}^5 SF_j = 1.13$$

El esfuerzo asociado al módulo es de aproximadamente 9 hombre x mes.

Cálculo del tiempo de desarrollo (TDEV):

$$TDEV = C * PM^F$$

$$\text{Donde } F = D + 0.2 * (E - B) = 0.012$$

$$TDEV = 3.7684$$

Cantidad de hombres (CH):

$$CH = \frac{PM}{TDEV} = 2.4074 \approx 2 \text{ Hombres}$$

Lo que muestra que el proyecto necesitaría 2 hombres para realizarlo en aproximadamente 4 meses.

Anexos

Costo por Mes (CHM):

$$CHM = \text{SalarioMedio} * CH$$

$$CHM = 200.00 * 2$$

$$CHM = \$400.00$$

Costo General (CG):

$$CG = CHM * TDEV$$

$$CG = 400.00 * 4$$

$$CG = \$1600.00$$

CHR: cantidad de hombres reales=1x salario promedio=200

ANEXO 5 Diagramas de Clases del Diseño Web.

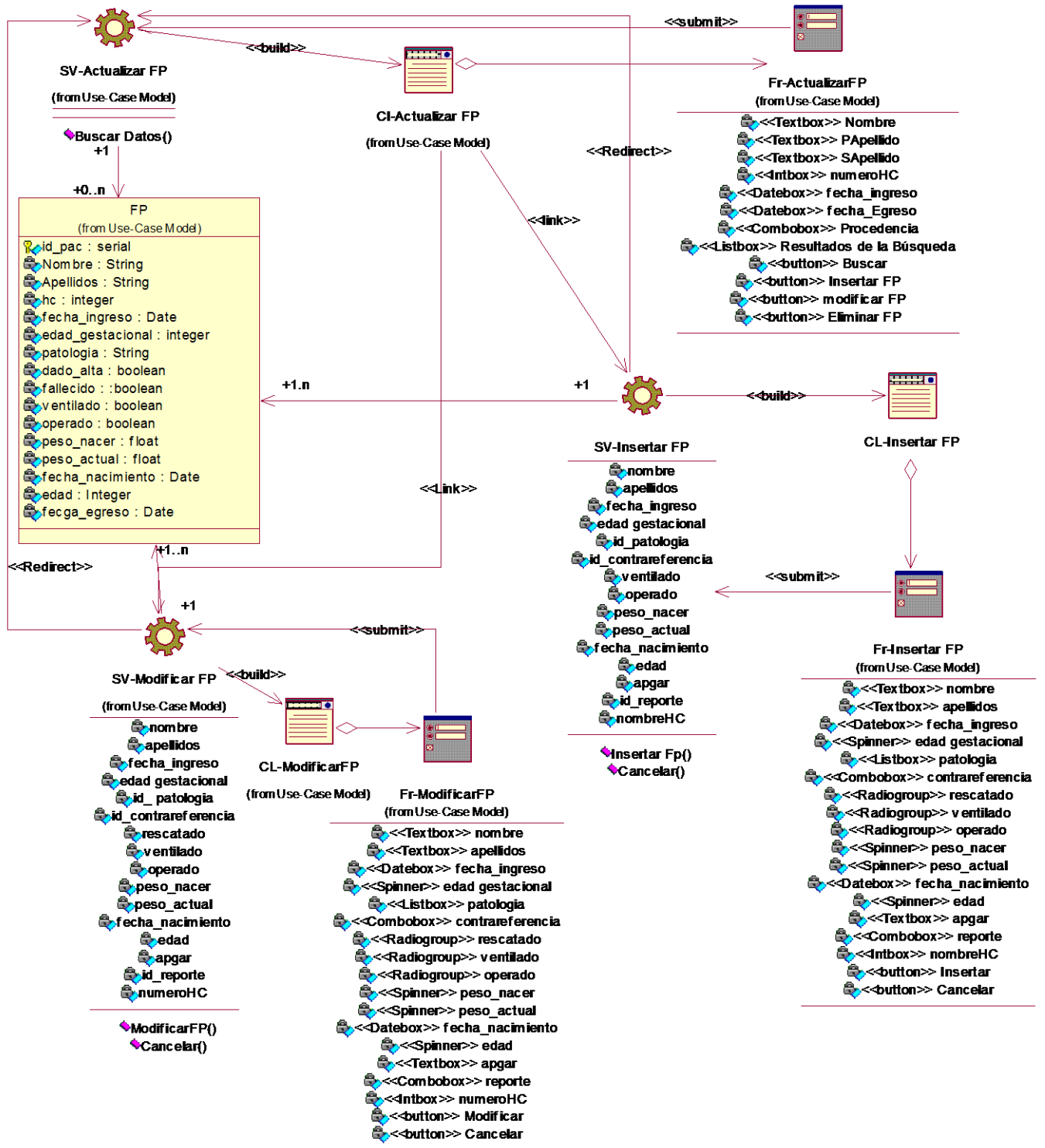


Figura 32 Diagrama de clases del diseño del paquete1 (Gestionar Ficha Paciente). Inserción y modificación

Diagrama de clases del diseño del paquete 2 Realizar Búsquedas por Criterios

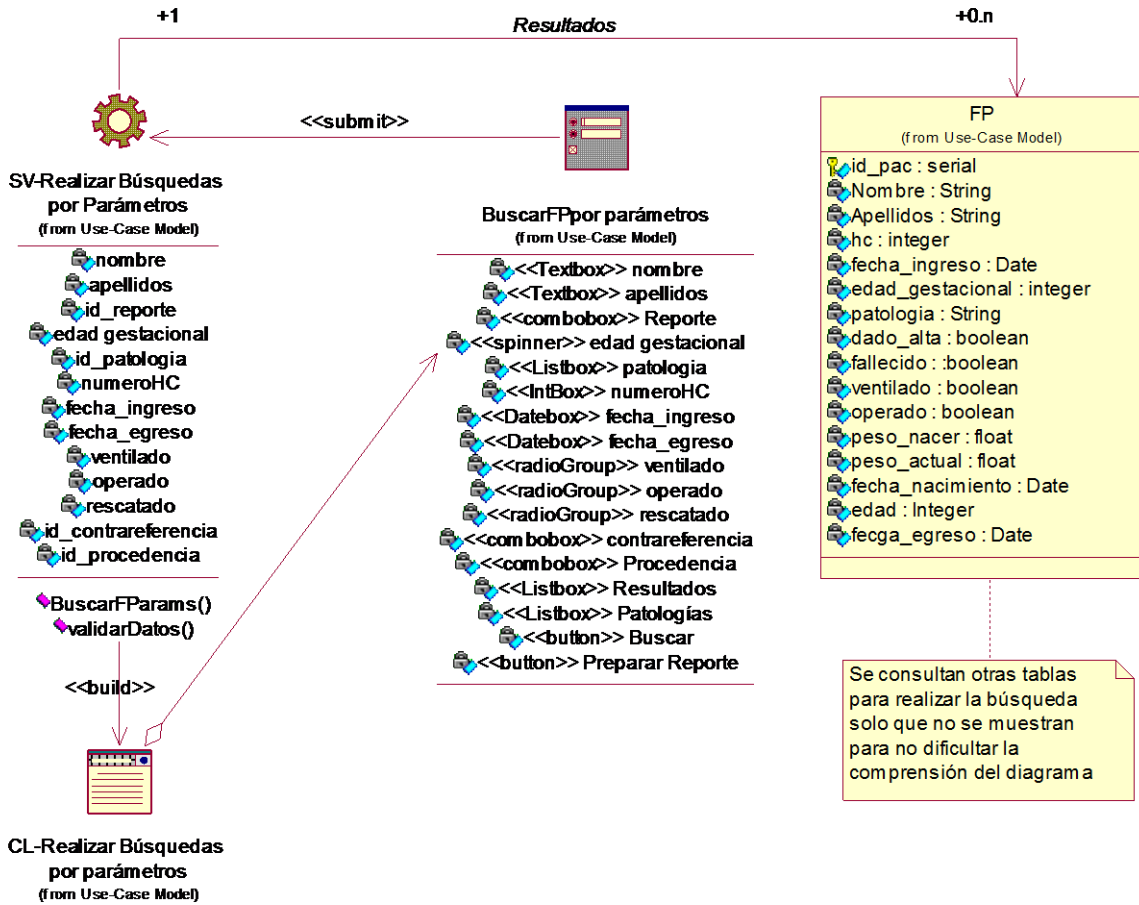


Figura 33 Diagrama de clases del diseño del paquete 2 Realizar Búsquedas por Criterios

Diagrama de clases del diseño del paquete 3 Visualizar Reportes

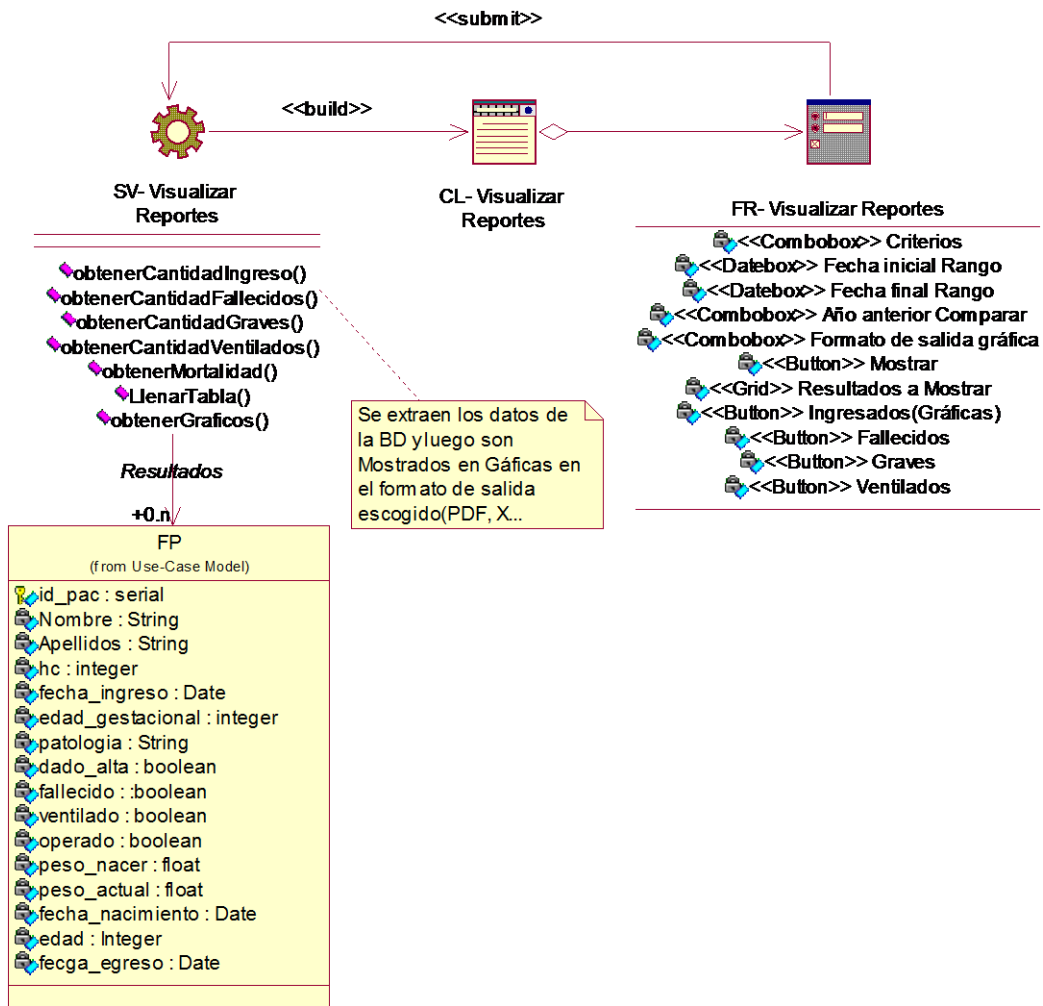


Figura 34 Diagrama de clases del diseño del paquete 3 Visualizar Reportes

Diagrama de clases del diseño del paquete 4 Gestionar Usuarios

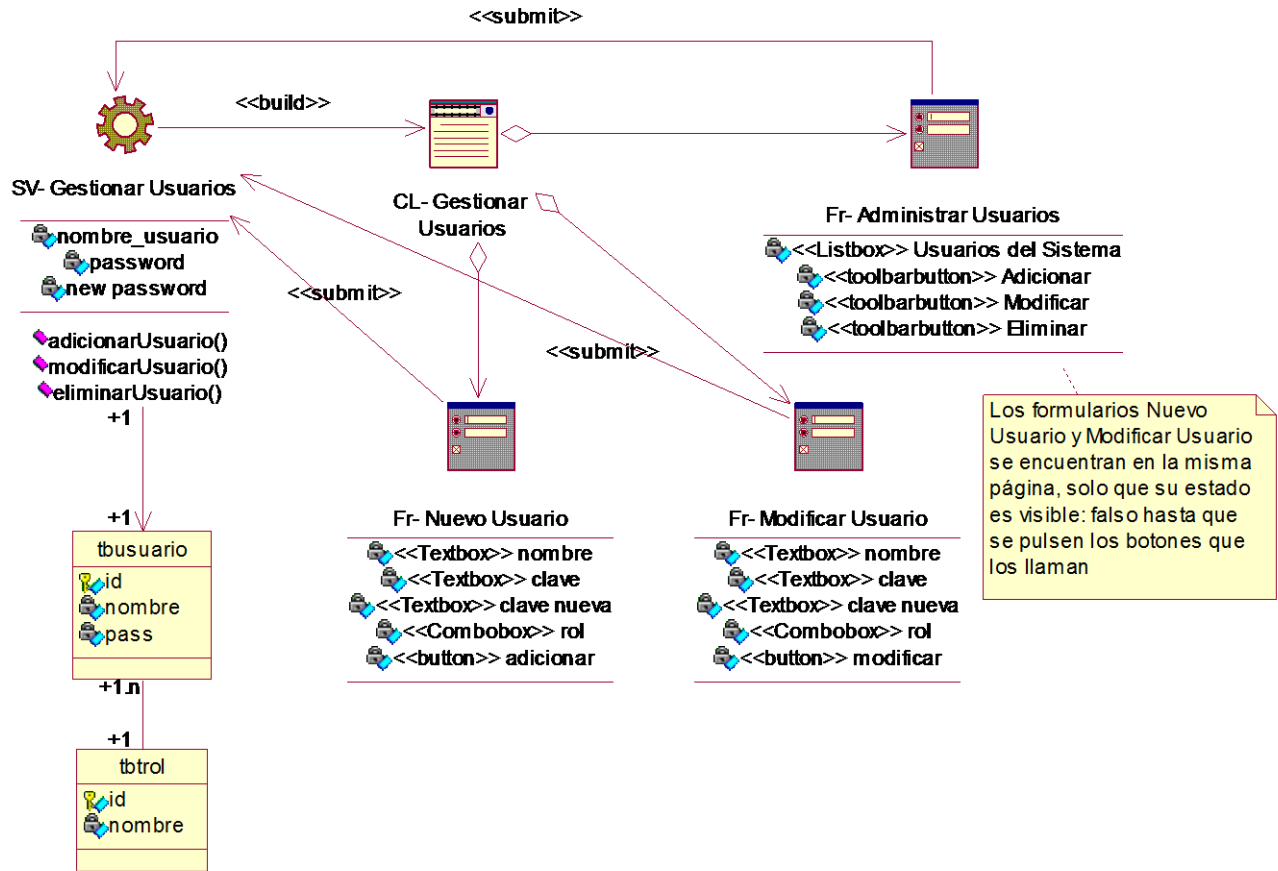


Figura 35 Diagrama de clases del diseño del paquete 4 Gestionar Usuarios

ANEXO 6 Diagrama de Clases Persistentes del Sistema NeoGest

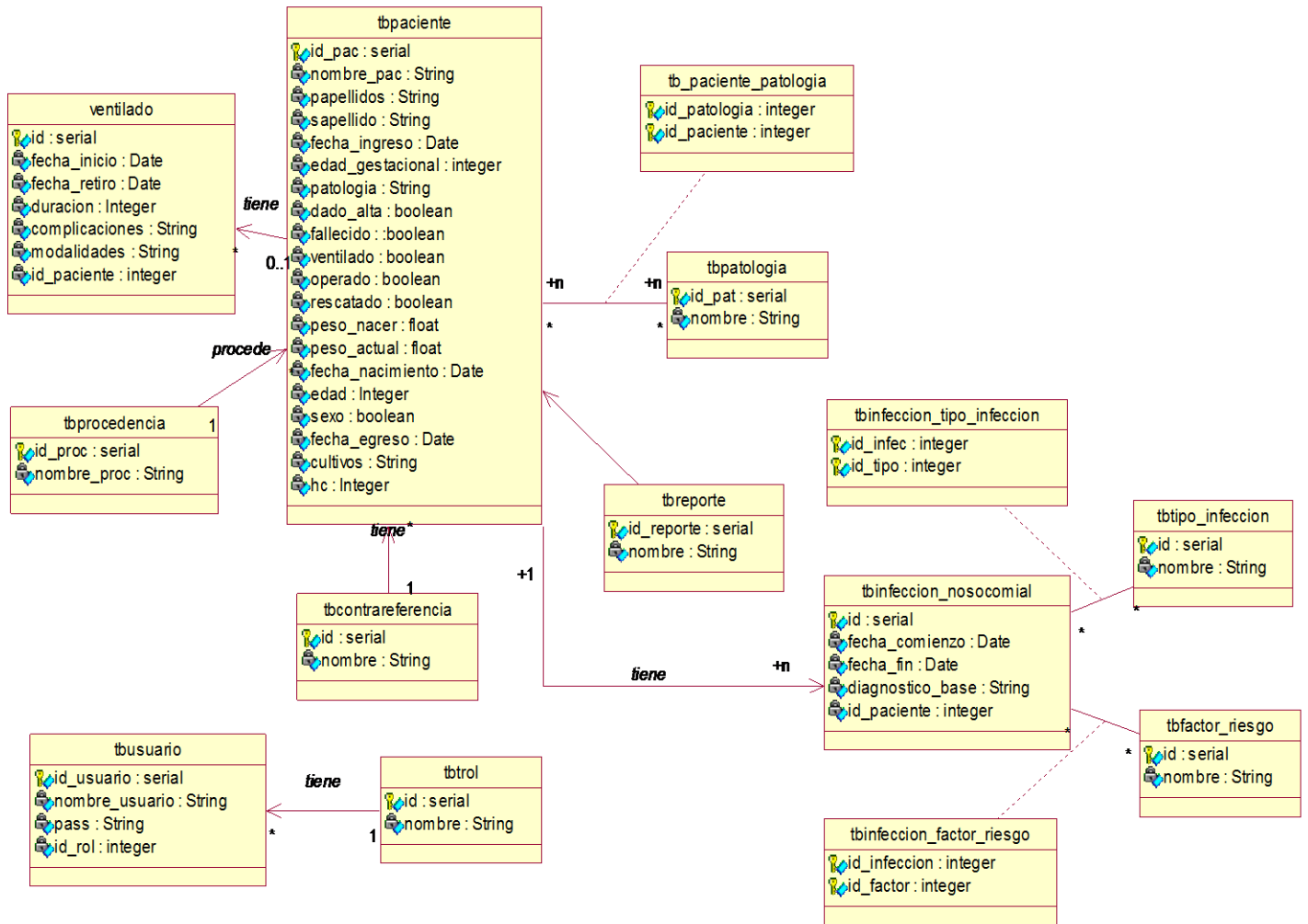


Figura 36 Diagrama de Clases Persistentes del Sistema NeoGest

ANEXO 7 Diagrama de Componente del Sistema NeoGest (paquete paciente y reporte)

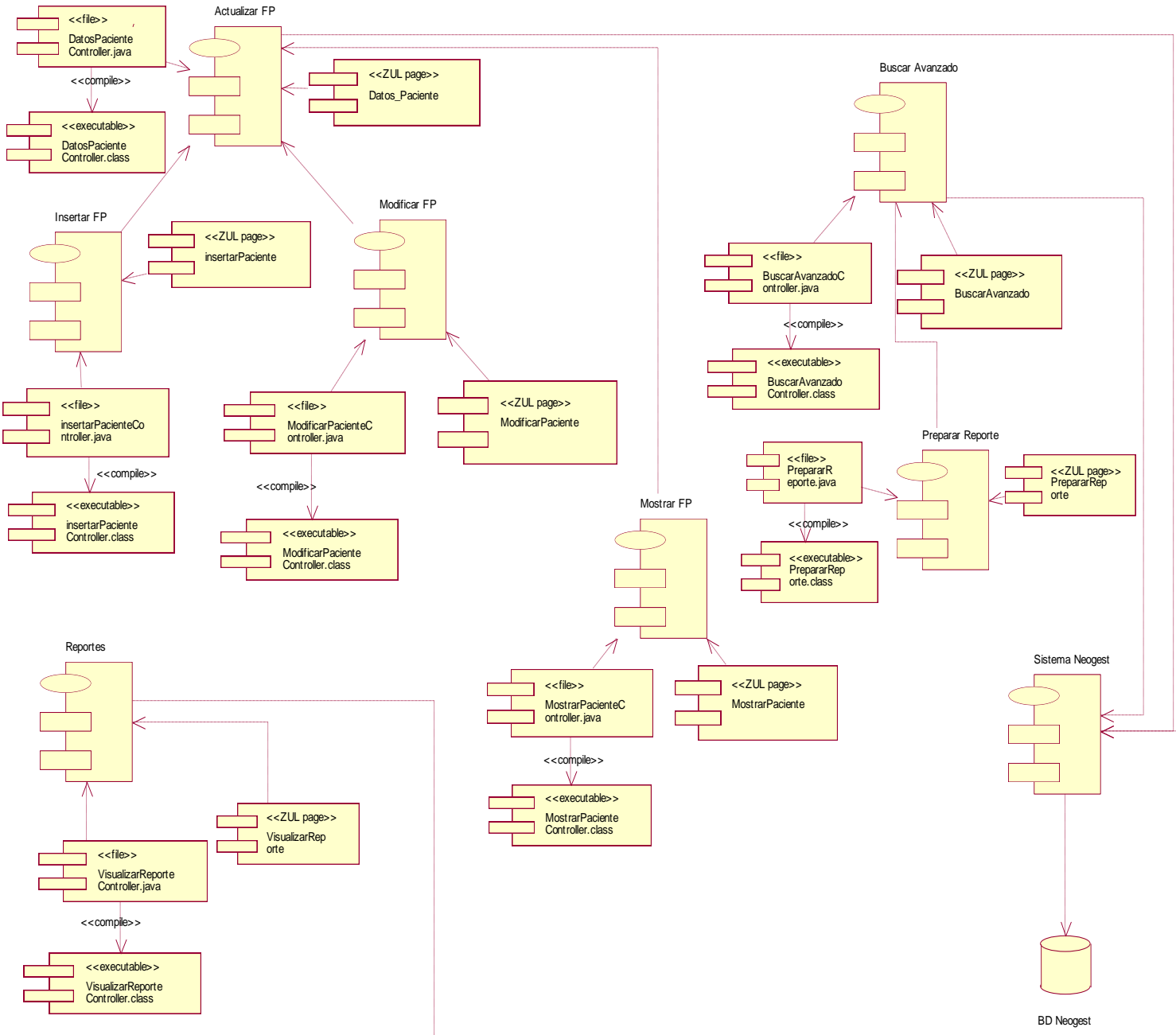


Figura 37 Diagrama de Componente del Sistema NeoGest (paquete Paciente y Reporte)

Diagrama de Componentes Sistema NeoGest (paquete Seguridad)

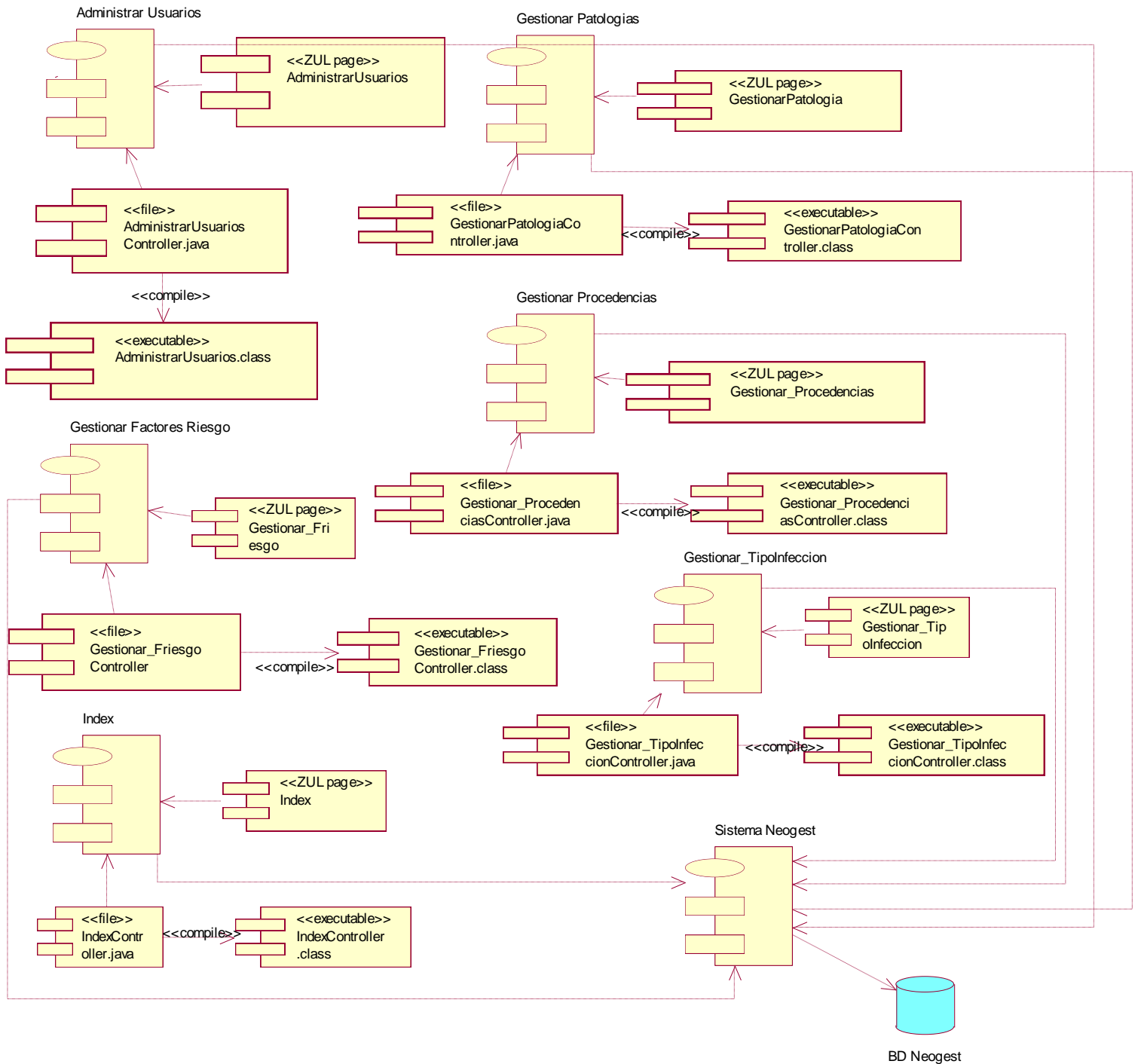


Figura 38 Diagrama de Componentes Sistema NeoGest (paquete Seguridad)

ANEXO 8 Tablas de los Casos de Pruebas y los Procedimientos de Pruebas.

Casos de Pruebas

Caso de uso: Realizar búsqueda por criterios	
Caso de prueba:	Realizar búsqueda por Parámetros con número de Historia Clínica existente en la BD.
Entrada:	El Médico decide realizar la búsqueda de un paciente, especificando el número de Historia Clínica "869145", encontrándose este numero de HC en la BD del sistema.
Resultado:	El sistema muestra una lista con los resultados de la búsqueda correspondiente a los datos del paciente con número de HC igual al especificado.

Tabla 30 Caso de prueba Realizar búsqueda por Criterios (caso1)

Caso de uso: Realizar búsqueda por criterios	
Caso de prueba:	Realizar búsqueda por Parámetros con número de Historia Clínica no existente en la BD.
Entrada:	El Médico decide realizar la búsqueda de un paciente, especificando el número de Historia Clínica "869145", no encontrándose este número de HC en la BD del sistema.
Resultado:	El sistema muestra un mensaje al usuario indicando que no existe ningún paciente en la BD con número de HC igual al especificado.

Tabla 31 Caso de prueba Realizar búsqueda por criterios (caso 2)

Caso de uso: Actualizar ficha del paciente (escenario insertar)	
Caso de prueba:	Insertar Ficha del paciente especificando todos los datos de la misma de forma correcta.
Entrada:	El Médico decide Insertar una ficha de un paciente, especificando los datos de entrada Nombre: "José

Anexos

	Luis”, apellidos: “Pérez Céspedes”, reporte: “regular”, edad gestacional: “38”, Diagnóstico Inicial: “Bronconeumonía”, No de Historia Clínica: “869145”, fecha de Ingreso: “06/02/10”, fecha de Nacimiento: “03/03/87”, ventilado: “sí”, operado: “sí”, dado de alta: “no” y fallecido: “no”, la información está correcta.
Resultado:	El sistema muestra un mensaje indicándole al usuario que la ficha del paciente ha sido insertada en la BD correctamente.

Tabla 32 Caso de prueba Actualizar ficha del paciente (caso1)

Caso de uso: Actualizar ficha del paciente (escenario insertar)	
Caso de prueba:	Insertar Ficha del paciente con todos los datos correctos, no especificando el número de HC del mismo.
Entrada:	El Médico decide Insertar una ficha de un paciente, especificando los datos Nombre: “José Luis”, apellidos: “Pérez Céspedes”, estado de Ingreso: “regular”, edad gestacional: “38”, Diagnóstico Inicial: “Bronconeumonía”, No de Historia Clínica: “ ”, fecha de Ingreso: “06/02/10”, fecha de Nacimiento: “03/03/87”, ventilado: “sí”, operado: “sí”, dado de alta: “no” y fallecido: “no”, la información está correcta pero el número de HC se encuentra en blanco.
Resultado:	El sistema muestra un mensaje indicándole al usuario que no puede dejar campos vacíos y que al paciente se le debe ingresar un número de HC, una vez confirmado este mensaje el sistema regresa al formulario de entrada de datos de la ficha del paciente.

Tabla 33 Caso de prueba Actualizar ficha del paciente (caso2)

Caso de uso: Actualizar ficha del paciente (escenario insertar)	
Caso de prueba:	Insertar Ficha del paciente, encontrándose en la BD.
Entrada:	El Médico decide Insertar una ficha de un paciente, especificando los datos Nombre: “José Luis”, apellidos:

Anexos

	<p>“Pérez Céspedes”, estado de Ingreso: “Grave”, edad gestacional: “38”, Diagnóstico Inicial: “Bronconeumonía”, No de Historia Clínica: “869145”, fecha de Ingreso: “06/02/10”, fecha de Nacimiento: “03/03/87”, ventilado: “si”, operado: “si”, dado de alta: “no” y fallecido: “no”, encontrándose almacenada la misma.</p>
Resultado:	<p>El sistema muestra un mensaje indicándole al usuario que ya existe una instancia de la HC en la BD y que en caso de insertar se crearía un reingreso o nueva FP.</p>

Tabla 34 Caso de prueba Actualizar ficha del paciente (caso3)

Caso de uso: Actualizar ficha del paciente (escenario modificar)	
Caso de prueba:	<p>Modificar Ficha del paciente que existe</p>
Entrada:	<p>El Médico decide modificar una ficha de un paciente, con los siguientes datos: Nombre: “José Luis”, apellidos: “Pérez Céspedes”, estado de Ingreso: “regular”, edad gestacional: “38”, patología: “Bronconeumonía”, No de Historia Clínica: “869145”, fecha de Ingreso: “06/02/10”, fecha de Nacimiento: “03/03/87”, ventilado: “si”, operado: “si”, dado de alta: “no” y fallecido: “no”, especificando los nuevos datos: fecha de ingreso: “05/02/10”, dado de alta: “si” . Los mismos se encuentran correctos.</p>
Resultado:	<p>El sistema muestra un mensaje indicándole al usuario que la ficha del paciente ha sido modificada correctamente.</p>

Tabla 35 Caso de prueba Actualizar ficha del paciente (caso 4)

Caso de uso: Actualizar ficha del paciente (escenario modificar)	
Caso de prueba:	<p>Modificar Ficha del paciente, entrando números en la caja de texto del nombre del paciente.</p>
Entrada:	<p>El Médico decide modificar una ficha de un paciente, especificando los datos Nombre: “José Luis”, apellidos: “Pérez Césp3des”, estado de Ingreso: “regular”, edad gestacional: “38”, patología: “asma”, No de Historia Clínica: “869145”, fecha de Ingreso: “06/02/10”, fecha de Nacimiento:</p>

Anexos

	“03/03/87”, ventilado: “si”, operado: “si”, dado de alta: “no” y fallecido: “no”. Todos los datos están correctos excepto el Segundo apellido del paciente:”José Luis Pérez Césp3des” que cuenta con errores.
Resultado:	El sistema muestra un mensaje indicándole al usuario que es un campo en el que solo se admiten letras y no números. Una vez confirmado este mensaje el sistema regresa al formulario de Modificar ficha del paciente.

Tabla 36 Caso de prueba Actualizar ficha del paciente (caso 5)

Procedimientos de Prueba.

Procedimiento de prueba: Realizar búsqueda por criterios	
Acción del actor	Respuesta del sistema
	1- El sistema se encuentra en la ventana correspondiente, permitiendo especificar el número de HC que servirá para realizar la búsqueda.
2- El Médico especifica el número de H.C [869145] que servirá de base para la búsqueda.	
3- El Médico decide que va a realizar la búsqueda.	4 El sistema valida la existencia de este número en la BD.
	5-El Sistema muestra los datos del paciente.
Cursos alternos	
<p>"Curso normal principal": Línea 4</p> <p>En caso de haber determinado el sistema que no había ningún paciente en la BD que tuviera igual número de H.C que el especificado, el sistema muestra un mensaje indicando que no existe ese paciente, y se devuelve el control a la línea 1 del curso normal principal.</p>	

Anexos

Tabla 37 Procedimiento de prueba Realizar búsqueda por criterios

Procedimiento de prueba:		Insertar Ficha del paciente	
Acción del actor		Respuesta del sistema	
		1- El sistema se encuentra en la ventana correspondiente, permitiendo especificar los datos para insertar la nueva ficha.	
2- El Médico especifica los datos que servirán de base para la inserción. Nombre: [José Luis], apellidos: [Pérez Céspedes], estado de Ingreso: [regular], edad gestacional: [38], patología: [asma], No de Historia Clínica: [869], fecha de Ingreso: [06/02/10], fecha de Nacimiento: [03/03/87], ventilado: [si], operado: [si], dado de alta: [no] y fallecido: [no],.			
3- El Médico decide que va a insertar la ficha.		4 El sistema valida que no existan errores en la introducción de los datos, campos vacíos así como otra instancia de la ficha en la BD.	
		5-El Sistema muestra un mensaje que indica que la FP ha sido insertada de forma correcta.	
Cursos alternos			
<p>"Curso normal principal": Línea 4</p> <p>En caso de determinar el sistema que habían errores a la hora de entrar los datos, campos vacíos (requeridos) o ya instanciada en la BD la FP, el sistema muestra un mensaje indicando que existen errores en la entrada de datos, no se pueden dejar campos vacíos y ya existe la ficha del paciente en la BD</p>			

Anexos

respectivamente, y se devuelve el control a la línea 1 del curso normal principal.

Tabla 38 Procedimiento de prueba insertar ficha del paciente

Procedimiento de prueba:	Actualizar ficha del paciente(escenario modificar)
Acción del actor	Respuesta del sistema
	1- El sistema se encuentra en la ventana correspondiente, permitiendo especificar los datos que servirán para modificar la FP.
2- El Médico especifica el nombre del paciente[José Luis Pérez] que servirá para modificar la FP.	
3- El Médico decide que va a modificar la ficha del paciente.	4 El sistema valida que el dato se encuentre correcto.
	5-El Sistema muestra un mensaje al usuario indicando que la ficha del paciente ha sido modificada satisfactoriamente.
Cursos alternos	
<p>"Curso normal principal": Línea 4</p> <p>En caso de haber determinado el sistema que existían problemas al modificar el nombre del paciente producto a errores cometido como introducir números en este campo, el sistema muestra un mensaje indicándole al usuario que es un campo en el que solo se admiten letras y no números y se devuelve el control a la línea 1</p>	

Tabla 39 Procedimiento de prueba Actualizar Ficha del paciente (escenario modificar)

ANEXO 9 Encuesta de opinión de los usuarios del sistema NeoGest

Nombre y apellidos: _____

Institución a la que pertenece: _____

Cargo actual: _____

Dada la importancia que tiene la neonatología y los servicios que se prestan en dicha sala así como el proceso de gestión de información de los pacientes ingresados, se realizó una investigación, dando como resultado la creación del sistema NeoGest para gestionar información y procesarla de forma correcta en el menor tiempo posible.

Evalúe y valore los aspectos siguientes comparando el sistema NeoGest con la forma actual de gestión de información, mediante la escala TA: totalmente de acuerdo, MA: muy de acuerdo, A: de acuerdo, PA: parcialmente de acuerdo, D: en desacuerdo. Marque con una **X** la escala que considera adecuada para cada aspecto.

Aspectos	TA	MA	A	PA	D
1. Las funcionalidades de la aplicación Web cubren las necesidades de los usuarios.					
2. El sistema actual permite que el usuario se sienta familiarizado con la aplicación Web.					
3. El producto informático permite el ahorro de recursos como el tiempo y materiales de oficina.					
4. Agiliza el proceso de gestión de información de los pacientes del servicio Neonatal.					
5. Se logró disminuir la probabilidad de errores al manipular la información así como aumentar su seguridad.					

Anexos

6. El diseño de la aplicación Web resulta más intuitivo y facilita la navegabilidad.					
7. El sistema actual facilita el acceso a la información referente al proceso de gestión de información de los pacientes del servicio de Neonatología.					
8. El tiempo de respuesta de la aplicación Web en la generación de búsquedas y reporte es más rápido.					
9. El sistema actual está acorde con el Procedimiento Específico que rige al proceso de gestión de información de los pacientes del servicio Neonatal.					

Tabla 40 Aspectos a medir en encuesta de opinión de los usuarios del sistema NeoGest

Por favor emita una valoración general del sistema actual y señale cualquier cambio o sugerencia que considere necesario.

Muchas gracias.

ANEXO 10 Procesamiento de la encuesta de opinión de los usuarios aplicando el método Delphi

TABLA DE FRECUENCIA ABSOLUTA						
Aspectos	TA	MA	A	PA	D	TOTAL
Las funcionalidades de la aplicación Web cubren las necesidades de los usuarios.	12	3	0	0	0	15
El sistema actual permite que el usuario se sienta familiarizado con la aplicación Web.	14	1	0	0	0	15
El producto informático permite el ahorro de recursos como el tiempo y materiales de oficina.	15	0	0	0	0	15
Agiliza el proceso de gestión de información de los pacientes del servicio Neonatal.	15	0	0	0	0	15
Se logró disminuir la probabilidad de errores al manipular la información así como aumentar su seguridad.	13	1	1	0	0	15
El diseño de la aplicación Web resulta más intuitivo y facilita la navegabilidad.	15	0	0	0	0	15
El sistema actual facilita el acceso a la información referente al proceso de gestión de información de los pacientes del servicio de Neonatología.	13	2	0	0	0	15
El tiempo de respuesta de la aplicación Web en la generación de búsquedas y reporte es más rápido.	15	0	0	0	0	15
El sistema actual está acorde con el Procedimiento Específico que rige al proceso de gestión de información de los pacientes del servicio Neonatal.	15	0	0	0	0	15

Tabla 41 Tabla de frecuencia absoluta

TABLA DE FRECUENCIA ABSOLUTA ACUMULADA					
Aspectos	TA	MA	A	PA	D
Las funcionalidades de la aplicación Web cubren las necesidades de los usuarios.	12	15	15	15	15
El sistema actual permite que el usuario se sienta familiarizado con la aplicación Web.	14	15	15	15	15
El producto informático permite el ahorro de recursos como el tiempo y materiales de oficina.	15	15	15	15	15
Agiliza el proceso de gestión de información de los pacientes del servicio Neonatal.	15	15	15	15	15
Se logró disminuir la probabilidad de errores al manipular la información así como aumentar su seguridad.	13	14	15	15	15
El diseño de la aplicación Web resulta más intuitivo y facilita la navegabilidad.	15	15	15	15	15
El sistema actual facilita el acceso a la información referente al proceso de gestión de información de los pacientes del servicio de Neonatología.	13	15	15	15	15
El tiempo de repuesta de la aplicación Web en la generación de búsquedas y reporte es más rápido.	15	15	15	15	15
El sistema actual está acorde con el Procedimiento Específico que rige al proceso de gestión de información de los pacientes del servicio Neonatal.	15	15	15	15	15

Tabla 42 Tabla de frecuencia absoluta acumulada

TABLA DEL INVERSO DE LA FRECUENCIA ABSOLUTA ACUMULADA				
Aspectos	TA	MA	A	PA
Las funcionalidades de la aplicación Web cubren las necesidades de los usuarios.	0,8	1	1	1
El sistema actual permite que el usuario se sienta familiarizado con la aplicación Web.	0,9333	1	1	1
El producto informático permite el ahorro de recursos	1	1	1	1

Anexos

como el tiempo y materiales de oficina.				
Agiliza el proceso de gestión de información de los pacientes del servicio Neonatal.	1	1	1	1
Se logró disminuir la probabilidad de errores al manipular la información así como aumentar su seguridad.	0,8667	0,9333	1	1
El diseño de la aplicación Web resulta más intuitivo y facilita la navegabilidad.	1	1	1	1
El sistema actual facilita el acceso a la información referente al proceso de gestión de información de los pacientes del servicio de Neonatología.	0,8667	1	1	1
El tiempo de repuesta de la aplicación Web en la generación de búsquedas y reporte es más rápido.	1	1	1	1
El sistema actual está acorde con el Procedimiento Especifico que rige al proceso de gestión de información de los pacientes del servicio Neonatal.	1	1	1	1

Tabla 43 Tabla del inverso de la frecuencia absoluta acumulada

TABLA DE DETERMINACIÓN DE LOS PUNTOS DE CORTES							
Aspectos	TA	MA	A	PA	Suma	Promedio	N - Prom.
Las funcionalidades de la aplicación Web cubren las necesidades de los usuarios.	0,84	3,49	3,49	3,49	11,31	2,83	0,34
El sistema actual permite que el usuario se sienta familiarizado con la aplicación Web.	1,5	3,49	3,49	3,49	11,97	2,99	0,18
El producto informático permite el ahorro de recursos como el tiempo y materiales de oficina.	3,49	3,49	3,49	3,49	13,96	3,49	-0,32
Agiliza el proceso de gestión de información de los pacientes del servicio Neonatal.	3,49	3,49	3,49	3,49	13,96	3,49	-0,32

Anexos

Se logró disminuir la probabilidad de errores al manipular la información así como aumentar su seguridad.	1,11	1,5	3,49	3,49	9,59	2,4	0,77
El diseño de la aplicación Web resulta más intuitivo y facilita la navegabilidad.	3,49	3,49	3,49	3,49	13,96	3,49	-0,32
El sistema actual facilita el acceso a la información referente al proceso de gestión de información de los pacientes del servicio de Neonatología.	1,11	3,49	3,49	3,49	11,58	2,9	0,27
El tiempo de repuesta de la aplicación Web en la generación de búsquedas y reporte es más rápido.	3,49	3,49	3,49	3,49	13,96	3,49	-0,32
El sistema actual está acorde con el Procedimiento Específico que rige al proceso de gestión de información de los pacientes del servicio Neonatal.	3,49	3,49	3,49	3,49	13,96	3,49	-0,32
Suma	22,01	29,42	31,41	31,41	114,3		
Punto de corte	2,45	3,27	3,49	3,49	12,69	3,17	= N (Prom. Gen.)

Tabla 44 Tabla de determinación de los puntos de cortes

Anexos

CONCLUSIONES GENERALES					
Aspectos	TA	MA	A	PA	D
Las funcionalidades de la aplicación Web cubren las necesidades de los usuarios.	Sí	-	-	-	-
El sistema actual permite que el usuario se sienta familiarizado con la aplicación Web.	Sí	-	-	-	-
El producto informático permite el ahorro de recursos como el tiempo y materiales de oficina.	Sí	-	-	-	-
Agiliza el proceso de gestión de información de los pacientes del servicio Neonatal.	Sí	-	-	-	-
Se logró disminuir la probabilidad de errores al manipular la información así como aumentar su seguridad.	Sí	-	-	-	-
El diseño de la aplicación Web resulta más intuitivo y facilita la navegabilidad.	Sí	-	-	-	-
El sistema actual facilita el acceso a la información referente al proceso de gestión de información de los pacientes del servicio de Neonatología.	Sí	-	-	-	-
El tiempo de respuesta de la aplicación Web en la generación de búsquedas y reporte es más rápido.	Sí	-	-	-	-
El sistema actual está acorde con el Procedimiento Específico que rige al proceso de gestión de información de los pacientes del servicio Neonatal.	Sí	-	-	-	-

Tabla 45 Tabla conclusiones generales (Método Delphi)

ANEXO 11 Entrevista realizada al cliente

ENTREVISTA AL CLIENTE

Fecha: __/__/____

Nombre: _____

Cargo: _____

Puntos a analizar:

- ¿Usted considera que el sistema actual cumple con todo lo establecido en el Procedimiento Específico de gestión de información?
- ¿Considera que se logró aumentar la seguridad del sistema?
- ¿Considera que con el sistema actual fueron cumplidas todas sus expectativas?
- ¿En qué medida usted piensa que el sistema NeoGest favorece al proceso de gestión de información de los pacientes del servicio?
- ¿Considera que se mejoró la eficiencia de los procesos respecto a la forma inicial?

Respuestas del cliente:
