

**UNIVERSIDAD DE ORIENTE**  
**NÚCLEO DE ANZOÁTEGUI**  
**ESCUELA DE INGENIERÍA Y CIENCIAS APLICADAS**  
**DEPARTAMENTO DE COMPUTACIÓN Y SISTEMAS**



**“DESARROLLO DE UN SISTEMA DE INFORMACIÓN PARA EL  
CONTROL DE REGISTROS MÉDICOS Y ESTADÍSTICA DEL HOSPITAL  
DE NIÑOS “DR. RAFAEL TOBIÁS GUEVARA” BARCELONA ESTADO  
ANZOATEGUI.”**

**REALIZADO POR:**

**LUIS CARLOS MARTÍNEZ CARVAJAL**  
**HÉCTOR LUIS MÁRQUEZ MÁRQUEZ**

**Trabajo de grado presentado como requisito parcial para optar al Título de:**

**INGENIERO DE SISTEMAS E INGENIERO EN COMPUTACIÓN**

**Puerto la Cruz, Agosto de 2010**

**UNIVERSIDAD DE ORIENTE**  
**NÚCLEO DE ANZOÁTEGUI**  
**ESCUELA DE INGENIERÍA Y CIENCIAS APLICADAS**  
**DEPARTAMENTO DE COMPUTACIÓN Y SISTEMAS**



**“DESARROLLO DE UN SISTEMA DE INFORMACIÓN PARA EL  
CONTROL DE REGISTROS MÉDICOS Y ESTADÍSTICA DEL HOSPITAL  
DE NIÑOS “DR. RAFAEL TOBÍAS GUEVARA” BARCELONA ESTADO  
ANZOÁTEGUI.”**

---

**ING. MANUEL CARRASQUERO**  
**Tutor Académico**

**Puerto la Cruz, Agosto de 2010**

**UNIVERSIDAD DE ORIENTE**  
**NÚCLEO DE ANZOÁTEGUI**  
**ESCUELA DE INGENIERÍA Y CIENCIAS APLICADAS**  
**DEPARTAMENTO DE COMPUTACIÓN Y SISTEMAS**



**“DESARROLLO DE UN SISTEMA DE INFORMACIÓN PARA EL  
CONTROL DE REGISTROS MÉDICOS Y ESTADÍSTICA DEL HOSPITAL  
DE NIÑOS “DR. RAFAEL TOBIÁS GUEVARA” BARCELONA ESTADO  
ANZOÁTEGUI.”**

**JURADO CALIFICADOR:**

---

**Ing. RHONALD RODRÍGUEZ**

**Jurado Principal**

---

**Ing. AIDA CARABALLO**

**Jurado Principal**

---

**ING. MANUEL CARRASQUERO**

**Tutor Académico**

**Puerto la Cruz, Agosto de 2010**

## **RESOLUCIÓN**

De acuerdo al Artículo 41 del Reglamento de Trabajos de Grado:

“Los Trabajos de Grado son de exclusiva propiedad de la Universidad de Oriente y sólo podrán ser utilizados a otros fines con el consentimiento del Consejo de Núcleo respectivo, quien lo participará al Consejo Universitario, para su autorización”

## **DEDICATORIA**

Dios por haber nos dado la oportunidad de existir, y darme fuerzas para seguir adelante con nuestras metas, gracias por estar en esos momentos difíciles y llenarme de esperanza para continuar creciendo como persona.

A mi madre Luisa Márquez en que enmarco mi vida en un sendero de alegría entusiasmo y ganas de vivir todo a plenitud.

A mi padre Héctor Guarache, el fiel ejemplo de humildad, responsabilidad y perseverancia.

A mis hermanos y hermanas, quienes me ayudaron mucho dándome inspiración para alcanzar las metas que hoy he logrado.

A todas mis amistades que de alguna u otra manera me ayudaron mucho.

A todas aquellas personas que tengan un sueño, que por mas difícil que sea, deben seguir la luz de su corazón que con humildad y esfuerzo, logran alcanzar el éxito en la vida, no lo olviden ¡Si se Puede!.

**Héctor L. Márquez .M.**

## **DEDICATORIA**

Dios por haber nos dado la oportunidad de existir, y darme fuerzas para seguir adelante con nuestras metas, gracias por estar en esos momentos difíciles y llenarme de esperanza para continuar creciendo como persona.

A mi esposa María Cabello en que enmarco mi vida en un sendero de alegría entusiasmo y ganas de vivir todo a plenitud.

A mi hijo Luis Carlos, que es mi fortaleza para seguir adelante  
A mi padres y mi madre que Dios los tenga en su gloria.

A mis hermanos y hermanas, lo cuales me ayudaron muchos dándome inspiración a alcanzar las metas que hoy en día he logrado.

A todas mis amistades que de alguna u otra manera me ayudaron mucho.

A todas aquellas personas que tengan un sueño, que por mas difícil que sea, deben seguir la luz de su corazón que con humildad y esfuerzo, logran alcanzar el éxito en la vida, no lo olviden ¡Si se Puede!.

**Luis C. Martínez C.**

## **AGRADECIMIENTOS**

Le damos gracias a mi Dios, a los angelito lindo y al Dr. José Gregorio Hernández, y San Miguel Arcángel por nuestra luz y guía espiritual.

Nuestra familia tuvo gran influencia en todos los resultados de nuestra vida, que son el pilar fundamental de ejemplo a seguir.

Nuestras vidas desde pequeño estaba influenciada por los consejos de nuestra madre, que fueron guía en todo, solo continué el camino que con amor y valores ella nos han enseñó gracias.

Llegamos a la universidad, con muchos deseos y expectativas, en primer lugar le damos gracias a las profesoras y profesores por su apoyo.

Tengo que agradecer al mundo a Dios, ya que me regalo tantos amigos (Hermanos) en esta vida universitaria, no se que decir para expresar lo especial que son en mi vida.

## **RESUMEN**

El siguiente proyecto de investigación se basó en el desarrollo de un sistema de información para el control de registros médicos y estadística del Hospital de niños “Dr. Rafael Tobías Guevara”, que se encuentra ubicado en la ciudad de Barcelona, Estado Anzoátegui, el cual lleva el nombre de (SIREM), dicho proyecto se encarga de proporcionar una interfaz agradable y de fácil manejo en entorno Web que permite a los usuarios realizar operaciones necesarias en las diferentes áreas en estudio como son: hospitalización, quirófano, emergencia y consulta externa. El sistema se encargará de registrar pacientes que ingresen al hospital y los doctores que laboran en él. Para luego así llevar un control de todas las estadísticas que sean necesarias. Este sistema garantiza información confiable ya que uno de sus principales objetivos es validar los datos ingresados. El software se elaboró utilizando el lenguaje de programación PHP versión 5.2.0, el servidor Apache versión 5.0 y el motor de base de datos MySQL versión 5.0. Este proyecto se construyó siguiendo el Proceso Unificado de Desarrollo de Software y la técnica de diagramación.



# CONTENIDO

	Pag.
Resolución.....	IV
Dedicatoria .....	V
Dedicatoria .....	VI
Agradecimientos .....	VII
Resumen .....	VIII
Contenido .....	IX
Lista de figuras.....	XV
Lista de tablas.....	<b>¡Error! Marcador no definido.</b>
<b>Capítulo I El Problema</b>	
1. 1. Planteamiento del problema.....	20
1.2 Objetivos .....	24
1.2.1 Objetivo General .....	24
1.2.2 Objetivos Específicos.....	24
<b>Capítulo II Marco Teórico</b>	
2.1. Antecedentes .....	26
2.2 Concepto de sistemas .....	28
2.2.1 Características de los Sistemas .....	28
2.2.2 Tipos de Sistemas.....	29
2.3 Sistemas de información (S.I).....	30
2.3.1 Características de los Sistemas de Información.....	30
2.3.2 Importancia de los Sistemas de Información .....	31
2.4 Datos e Información.....	31

2.4.1	Procesamiento de Datos .....	32
2.5	Componentes de un Sistema de Información.....	33
2.5.1	Componentes Físicos .....	33
2.5.2	Componentes Funcionales .....	33
2.6	Funciones de un Sistema de Información .....	33
2.7	Ciclo de Vida de un Sistema de Información .....	34
2.8	Ciclo de Desarrollo de un Sistema de Información .....	34
2.9	Desarrollo de software .....	36
2.9.1	Ingeniería de Software .....	36
2.9.2	Fases de la Ingeniería de Software.....	37
2.10	Paradigma de programación orientado a objetos .....	38
2.10.1	Modelado de Objetos .....	40
2.11	El Proceso Unificado de Desarrollo de Software.....	41
2.11.1	La vida del Proceso Unificado de Desarrollo de Software .....	42
2.11.2	Fases del Proceso Unificado de Desarrollo de Software .....	43
2.11.2.1.	Fase de Inicio .....	43
2.11.2.2	Fase Elaboración .....	44
2.11.2.3.	Fase de Construcción .....	44
2.11.2.4.	Fase de Transición.....	45
2.12	Base de datos.....	45
2.12.1.	Ventajas de las Bases de Datos .....	46
2.12.2	Sistema de Manejador de Base de Datos .....	47
2.12.2.1.	Objetivos del SMBD .....	47
2.13	Diseño de Una Base de Datos .....	47
2.14	PHP .....	48
2.15	Tecnología Web .....	49
2.15.1	Concepto .....	49

2.16 Protocolo http.....	49
2.16.1 Concepto .....	49
2.17 Servidor Apache Web .....	50
2.17.1 Concepto .....	50
2.17.2 Ventajas del Servidor HTTP Apache.....	50
2.18 Navegador Web.....	51
2.18.1 Concepto .....	51
2.18.2 Estructura de las Aplicaciones Web.....	51
2.19 MySQL.....	52
2.20 WAMP .....	52
<b>Capítulo III Marco Metodológico</b>	
3.1 Tipos de Investigación. ....	55
3.1.1 Nivel de la Investigación.....	55
3.1.2 Diseño de la investigación. ....	56
3.2 Técnicas a utilizar. ....	56
<b>Capítulo IV Resultados</b>	
4.1 Fase de Inicio .....	60
4.2 Contexto del Sistema de información de Registro y Estadístico Médico SIREM. ....	60
4.2.1. Modelo del Dominio .....	62
4.2.2 Glosario de Términos.....	63
4.2.3 Riesgos críticos .....	64
4.2.3.1 Identificación de los actores.....	64
4.2.3.2 Identificación de los casos de uso .....	64
4.2.4 Descripción de los casos de uso.....	67
4.2.4.1 Caso de uso de la realización del caso de uso procesar doctor .....	67
4.2.4.2 Caso de uso de la realización del Caso de Uso Procesar Consulta	

Externa .....	69
4.2.4.3 Caso de uso de la realización del Caso de Uso Procesar Emergencia	70
4.2.4.4 Caso de uso de la realización del Caso de Uso Procesar Hospitalización.....	71
4.2.4.5 Caso de uso de la realización del Caso de Uso Procesar Quirófano..	72
4.3 Análisis.....	72
4.3.1 Diagrama de clases de análisis de la realización del caso de uso Procesar Doctor.....	73
4.3.2 Diagrama de clases de colaboración de la realización del Caso de Uso Procesar Doctor.....	74
4.3.3 Diagrama de clases de análisis de la realización del caso de uso Procesar Consulta Externa .....	75
4.3.4 Diagrama de clases de colaboración de la realización del Caso de Uso Procesar Consulta externa.....	76
4.3.5 Diagrama de clases de análisis de la realización del caso de uso Procesar Emergencia.....	77
4.3.6 Diagrama de clases de colaboración de la realización del Caso de Uso Procesar Emergencia.....	78
4.3.7 Diagrama de clases de análisis de la realización del caso de uso Procesar Hospitalización.....	79
4.3.8 Diagrama de clases de colaboración de la realización del Caso de Uso Procesar Hospitalización.....	80
4.3.9 Diagrama de clases de colaboración de la realización del caso de uso Procesar Quirófano. ....	81
4.3.10 Diagrama de clases de colaboración de la realización del caso de uso Procesar Quirófano .....	82
4.4 Diagrama de Paquete .....	83

4.5 Conclusión de la fase de inicio .....	84
4.6 Fase de Elaboración .....	85
4.6.1 Requisitos.....	86
4.6.2 Caso de uso de la realización del Caso de Uso procesar reportes.....	86
4.7 Análisis.....	88
4.7.1 Diagrama de clases de análisis de la realización del caso de uso Procesar Reportes .....	88
4.7.2 Diagrama de colaboración de la realización del caso de uso generar reportes.....	88
4.8 Diseño .....	89
4.8.1 Diagrama de Clases de Diseño.....	91
4.8.2 Diagrama de Secuencia de la realización del Caso de Uso Procesar Paciente del Sistema SIREM .....	92
4.8.3 Diseño de la arquitectura.....	94
4.8.4 Diseño de las Bases de Datos DB SIREM.....	94
4.8.4.1 Tablas de la Base de Datos DB SIREM.....	97
4.9 Implementación.....	102
4.9.1 Identificación de Componentes de la Arquitectura.....	102
4.9.2 Implementación de la Arquitectura.....	103
4.9.3 Conclusión de la Fase de Elaboración .....	104
4.10 Fase de construcción .....	105
4.10.1 Requisitos.....	105
4.10.2 Diseño de la interfaz de Usuario .....	107
4.11 Flujo de Trabajo: Implementación .....	109
4.11.1 Flujo de Trabajo: Pruebas .....	118
4.11.2 Pruebas de Integración.....	119
4.11.3 Pruebas de Caja Negra .....	119
4.11.4 Conclusión de la Fase de Construcción .....	121

4.12 Fase de Transición.....	121
4.12.1 Evaluación de la Fase de Transición.....	123
4.12.2 Flujo de Trabajo Pruebas .....	123
4.12.2.1 Aplicación de las Pruebas .....	123
4.12.3 Conclusión de la Fase de Transición.....	123
Conclusiones .....	125
Recomendaciones.....	127
Bibliografía .....	128
Anexos .....	130

## LISTA DE FIGURAS

	Pag.
Figura 1.1. Organigrama Estructural del Departamento de Registros Médicos y Estadísticas.....	22
Figura 2.1 Modelo Dinámico de un Sistemas de Información.....	31
Figura 2.3 Estructura General de una Aplicación Web.....	52
Figura 4.1 Modelo De Dominio. ....	63
Figura 4 .2 Diagrama de casos de uso Principal del Sistema SIREM.....	68
Figura 4.3 Diagrama de casos de uso de Procesar Doctor .....	69
Figura 4.4 Diagrama de casos de uso de Procesar Consulta Externa.....	70
Figura. 4.5 Diagrama de casos de uso Procesar Emergencia.....	71
Figura 4.6 Diagrama de casos de uso de Procesar Hospitalización .....	72
Figura. 4.7 Diagrama de casos de uso de Procesar Quirófano.....	73
Figura 4.8 Diagrama de clases de análisis de la realización del Caso de Uso Procesar Doctor.....	74
Figura 4.9 Diagrama de clases de colaboración de la realización del Caso de Uso Procesar Doctor.....	75
Figura 4.10 Diagrama de clases de análisis de la realización del Caso de Uso Procesar Consulta Externa .....	76
Figura 4.11 Diagrama de clases de colaboración de la realización del Caso de Uso Procesar Consulta Externa .....	77
Figura 4.12 Diagrama de clases de análisis de la realización del Caso de Uso	

Procesar Emergencia.....	78
Figura 4.13 Diagrama de clases de Colaboración de la realización del Caso de Uso Procesar Emergencia.....	79
Figura 4.14 Diagrama de clases de análisis de la realización del Caso de Uso Procesar Hospitalización.....	80
Figura 4.15 Diagrama de clases de Colaboración de la realización del Caso de Uso Procesar Hospitalización.....	81
Figura 4.16 Diagrama de clases de análisis de la realización del Caso de Uso Procesar Quirófano .....	82
Figura 4.17 Diagrama de clases de Colaboración de la realización del Caso de Uso Procesar Quirófano .....	83
Figura 4.18 Diagrama de paquete de análisis a partir de los casos de uso del sistema SIREM .....	85
Figura 4.19 Flujo de trabajo de la fase de Elaboración del Sistema SIREM.....	87
Figura 4.20 Diagrama de casos de uso de procesar reportes.....	87
Figura 4.21 Diagrama de clases de análisis de la realización del Caso de Uso Procesar Reportes.....	89
Uso Generar Reportes Estadísticos .....	90
Figura 4.23 Diagrama general de la Clase de Diseño .....	92
Figura 4.24 Diagrama de secuencia de la realización del caso de uso Procesar Paciente del sistema SIREM .....	95
Figura 25 Diseño de la arquitectura del sistema .....	96
Figura 4.26 Relación de Tablas de la Base de Datos .....	97
Figura 4.27 Diagrama de Despliegue del Sistema. ....	103



Figura 4.28 Pantalla de Inicio de Sesión.....	104
Figura 4.29 Flujo de trabajo de la fase de Elaboración del Sistema SIREM.....	106
Figura 4.30 Pantalla de Inicio de Sesión.....	112
Figura 4.31 Pantalla Ingresar los datos del paciente.....	113
Figura 4.32 Reporte de Quirófano .....	115

## LISTA DE TABLA

	Pag.
Tabla 4.1 Terminología utilizada .....	65
Tabla 4.2 Riesgos Críticos del Sistema SIREM .....	66
Tabla 4.3 Actores del sistema SIREM .....	67
Uso Generar Reportes Estadísticos .....	90
Tabla 4.4 Acceso de Usuarios.....	98
Tabla 4.5 Quirófano .....	99
Tabla 4.6 Emergencia.....	99
Tabla 4.7 Consulta externa.....	100
Tabla 4.8 Hospitalización .....	101
Tabla 4.9 Paciente .....	101
Tabla 4.10 Paciente .....	102
Tabla 4.11 Clase de Equivalencia para la Operación: Crear paciente .....	120
Tabla 4.12 Clase de Equivalencia para la Operación: Crear Doctor.....	121
Tabla 4.13 Clase de Equivalencia para la Operación: Modificar Paciente .....	121

Hospital de Niños Dr. Tobías Guevara  
Barcelona Edo. Anzoátegui



# CAPÍTULO I

## EL PROBLEMA

# CAPÍTULO I

## EL PROBLEMA

### 1. 1. Planteamiento del problema

Gracias al incesante avance de la tecnología, hoy en día es posible delegar las gestiones de proceso administrativo a sistemas computarizado que faciliten el manejo de grandes volúmenes de información y que aumenten la eficiencia y la productividad, optimizando el tiempo y la calidad de respuesta a las solicitudes de servicio.

El Hospital de Niños “Dr. Rafael Tobías Guevara”, ubicado en la ciudad de Barcelona Estado. Anzoátegui, representa un sistema esencial en las entidades locales de salud. Ésto se debe no sólo al papel que juega en la atención de pacientes que requieren servicios más especializados que los disponibles en los ambulatorios y centros de salud, sino también por el apoyo que ofrecen a este establecimiento de mayor complejidad en cuanto a sistema de información, funciones gerenciales, desarrollo de recursos humanos, logística e investigación. A causa de su naturaleza pública el Hospital de Niños “Dr. Rafael Tobías Guevara”, recibe un porcentaje de recursos, tanto económicos como tecnológicos, del gobierno central, siendo encausado a nivel regional a través de SALUDANZ.

Dentro de la estructura organizacional del Hospital de Niños “Dr. Rafael Tobías Guevara”, se encuentra el Departamento de Registros Médicos y Estadística. Este departamento está constituido por secciones de Archivos, Admisión, Consulta Externa y Estadísticas.

El Departamento de Registros Médicos y Estadística de una institución de salud, garantiza en forma idónea, la información necesaria para optimizar el procesamiento de los registros asistenciales que puedan permitir el estudio cuantitativo y cualitativo de los problemas de salud de la comunidad, con base en la toma de decisiones, redirección de programas y evaluación de recursos y funciones.

Los datos proporcionados por el Departamento de Registros Médicos y Estadística son utilizados en todas las etapas de la administración del hospital, desde el momento de la planificación hasta la evaluación de las acciones desarrolladas.

El Departamento de Registros Médicos y Estadística, es responsable de la recopilación, tabulación, presentación y análisis descriptivo de toda la información que se produce en el establecimiento de salud, así como: el análisis cuantitativo de la historia clínica, la codificación y registro de la morbilidad atendida en el área de hospitalización y el control y atención a las solicitudes para los trabajos de investigación y especiales, lo que permitirá evaluar cualitativa y cuantitativamente los servicios integrales que se presentan a la comunidad a través de la promoción, defensa y restitución de la salud.

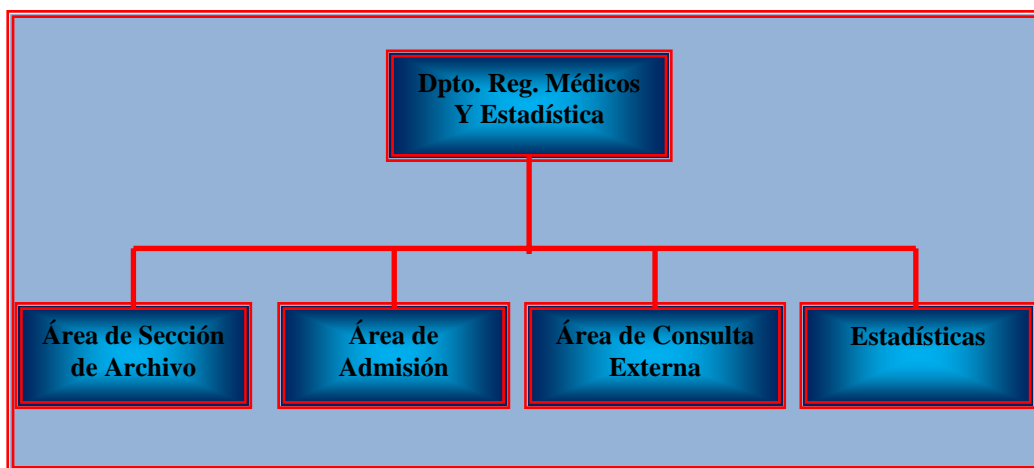
El Hospital de Niños de Oriente, abrió sus puertas el 01 de Abril del 2000. Se inició con la apertura de la consulta externa, debidamente apoyada por enfermería, trabajo social, registro médico y laboratorio, para ese entonces sólo se atendían un promedio de 50 niños diarios.

Actualmente los datos que maneja el Departamento de Registros Médicos y Estadística del Hospital de Niños “Dr. Rafael Tobías Guevara”, son suministrados por los doctores, pacientes y el personal encargado de las diferentes áreas de trabajo, los cuales son manejados en forma manual ya que estos son elaborados en formato. Todo esto, origina un continuo uso de cualquier papel o formato codificador, para escribir

los registros de los datos sociales y médicos del paciente, corriendo el riesgo que estos papeles o información se pierda. Estos archivos son almacenados en áreas inadecuadas (estantes, carpeta, sitios húmedos y en sitios sin ningún tipo de seguridad), dificultando de esta manera la obtención de información por parte de la dirección del Hospital.

El organigrama estructural del Departamento de Registros Médicos y Estadística están representados (figura 1.1)

El propósito del trabajo de investigación será el desarrollo de un Sistema de Información, seguro y confiable que contribuya a satisfacer los requerimientos de reducción de tiempo al momento de requerir información de los registros estadísticos médicos del paciente que son atendidos en el Hospital de Niño. Para su desarrollo se utilizará una moderna metodología que sobresale en el desarrollo de *Software* denominada Proceso Unificado la cual usa una novedosa herramienta para el modelaje de proyectos conocida como U.M.L (Lenguaje Unificado del Modelado) y para el diseño de la base de datos se utilizará el Modelo Relacional



**Figura 1.1. Organigrama Estructural del Departamento de Registros Médicos y Estadísticas.**

Fuente: Hospital del niño “Dr. Rafael Tobías Guevara”.

La importancia de éste trabajo de investigación es que permitirá estar a la par con la tecnología la cual define condiciones para lograr la prestación de un servicio de calidad.

La originalidad de este proyecto, estará dado por ser el primer trabajo que se elabora en el Departamento de Registros Médicos y Estadística en el Hospital de Niños “Dr. Rafael Tobías Guevara”, utilizando metodología de desarrollo de software como es el Proceso Unificado de desarrollo.

El propósito de este software es minimizar el tiempo de repuesta y agilizar la manipulación de la información, beneficiando tanto a los usuarios como a los empleados, doctores que trabajan en las diferentes áreas del hospital de niños. Conjuntamente se estima almacenar toda la información relacionada con los usuarios procesando los datos almacenados para realizar consultas, reportes, estadísticas, etc.; de importancia para la actividad que se lleva a cabo en el recinto hospitalario.

Para la elaboración del software, es necesario el uso de una metodología que garantice el correcto análisis y diseño de todo los procedimientos que integran el sistema. La metodología a utilizar es Proceso Unificado de Desarrollo (RUP), el cual está orientado a objetos, dirigido por casos de uso, es interactivo y está basado en la arquitectura del software. Esta metodología es de gran aceptación y utilización a nivel mundial. Por tal motivo, el interés de emplear este proceso, ya que es una tarea clave para modelar un sistema de software de grande dimensiones, permitiendo dividir y categorizar el trabajo. En la elaboración del código del sistema se utilizará lenguaje de programación PHP y MySQL para la elaboración de la base de datos.

La implementación del software pretende mejorar la gestión de estadística del hospital relacionadas con la atención de paciente en las áreas de: emergencia, consulta externa, hospitalización y quirófano.



El desarrollo del software abarca la ejecución de la fase inicio, elaboración, construcción y transición correspondiente a la Metodología del RUP.

## **1.2 Objetivos**

### **1.2.1 Objetivo General**

Desarrollar un sistema de información para el control de registros médicos y estadística del Hospital de Niños “Dr. Rafael Tobías Guevara” Barcelona Estado Anzoátegui.

### **1.2.2 Objetivos Específicos**

- Describir los procedimientos para el procesamiento de datos de los registros médicos y estadísticas utilizado actualmente.
- Determinar los requisitos necesarios para el diseño del sistema.
- Modular la estructura del software.
- Diseñar la interfaz del sistema.
- Diseñar la estructura de los reportes y consultas.
- Construir el software del nuevo sistema.
- Realizar pruebas de integración del nuevo sistema.



*Hospital de Niños “Dr. Rafael Tobías Guevara”.  
Barcelona. Estado Anzoátegui.*

## CAPÍTULO II

### MARCO TEÓRICO

## **CAPÍTULO II**

### **MARCO TEÓRICO**

#### **2.1. Antecedentes**

Es la primera vez que se inicia el desarrollo de una investigación utilizando el proceso de desarrollo de software Rup, como metodología para el Departamento de Registros Médicos y Estadística en el Hospital de Niños “Dr. Rafael Tobías Guevara”, sin embargo, en la Universidad de Oriente se han llevado a cabo otros trabajos de investigación dirigidos a la automatización de procesos dentro de empresas e instituciones, en los cuales se ha empleado la metodología y las técnicas necesarias para realizar dichos estudios, a continuación se citan algunos que se han hecho recientemente:

Rodríguez, A. y López, C. (2001), desarrollaron un trabajo de grado titulado “Estudio del clima organizacional para mejorar el rendimiento en el área operativa y administrativa de los Departamentos Críticos del Hospital de Veteranos “Dr. Julio Rodríguez” esta investigación se origina con la finalidad de analizar el funcionamiento del Hospital de Veterano “Dr. Julio Rodríguez”, para así ofrecer un servicio médico eficiente que cubra la necesidades de la población para llevar a cabo el estudio. Se realizó una evaluación en el ámbito gerencial de la instalación a fin de diagnosticar el sistema actual a través de la primera fase de la metodología de Checkland, conocido el sistema se aplico el análisis estratégico para determinar el departamento más crítico.

Gotilla, M. y Guacaran, G. (2001), desarrollaron un trabajo de grado titulado “Diseño de un Plan Estratégico para el Departamento de Gineco-obstetrica del

Hospital Universitario “Dr. Luis Razetti” del Estado Anzoátegui” dentro de la estructura del hospital Universitario “Dr. Luis Razetti” del Estado Anzoátegui se encuentra el Departamento de Gineco-obstetrica, el cual se encarga de brindar asistencia médica preventiva y curativa a la población femenina del Estado Anzoátegui. El Departamento desde su creación en 1963 ha contado con una gestión administrativa basado en la experiencia, sin tomar en cuenta premisa estrategia. De esto nace la necesidad de realizar un estudio que genere un plan estratégico para el Departamento. La metodología utilizada para elaborar la presente investigación incluye distintas áreas, tales como: Enfoque de sistemas el cual emplea los modelos general de los sistemas abierto para conocer el funcionamiento del Departamento, planificación estratégica, para determinar factores tanto interno (debilidades y fortaleza) como externa (de manera, oportuna).

Alvarez, W. y Brito, M. (2001), desarrollaron un trabajo de grado titulado “Estudio sistémico del servicio médico de salud ocupacional de una empresa fabricante de tubos de acero, para el Diseño de un sistema de control de gestión de información”. Debido a la poca gestión de flujo de información en la realización de las tareas internas, procesos, entre otras actividades, y manejos de las información entre las unidades, y dentro de la unidad de servicios médicos de Salud ocupacional de HELVESA, se hizo imperativo la realización de un estudio logístico que permitiera determinar los diferentes factores, que estaba ocasionando la situación problema. Para la realización del estudio se utilizó el enfoque de sistema aplicando como herramienta principal la metodología de Checkland, para identificar las situaciones que alteraban su óptimo desempeño. Y por último, el diseño de un sistema de control de gestión de información, logrando integrar así los diferentes procesos que se lleven a cabo en la unidad y la relación con los niveles gerenciales y departamentales de la empresa.

Urbin, B. y Cedeño, E. (2001), desarrollaron un trabajo de grado titulado “Diseño de un plan estratégico para el departamento de Registro Médicos y Estadística del hospital Universitario “Dr. Luis Razetti” del estado Anzoátegui”. La metodología utilizada para realizar el presente plan estratégico incluye diversas áreas, enfoque sistemático, se utilizó los tres primeros estudios de la metodología para sistemas blandos de Peter.

## **2.2 Concepto de sistemas**

Conjunto de elementos dinámicamente relacionados formando una actividad para alcanzar un objetivo operando sobre datos/energía/materia para proveer información/energía/materia.

### **2.2.1 Características de los Sistemas**

Sistema es un todo organizado y complejo; un conjunto o combinación de cosas o partes que forman un todo complejo o unitario. Es un conjunto de objetos unidos por alguna forma de interacción o interdependencia. Los límites o fronteras entre el sistema y su ambiente admiten cierta arbitrariedad.

Según Bertalanffy, sistema es un conjunto de unidades recíprocamente relacionadas. De ahí se deducen dos conceptos: propósito (u objetivo) y globalismo (o totalidad).

- **Propósito u objetivo:** Todo sistema tiene uno o algunos propósitos. Los elementos (u objetos), como también las relaciones, definen una distribución que trata siempre de alcanzar un objetivo.

- **Globalismo o totalidad:** Un cambio en una de las unidades del sistema, con probabilidad producirá cambios en las otras. El efecto total se presenta como un ajuste a todo el sistema. Hay una relación de causa/efecto. De estos cambios y ajustes, se derivan dos fenómenos: entropía y homeostasia.
- **Entropía:** Es la tendencia de los sistemas a desgastarse, a desintegrarse, para el relajamiento de los estándares y un aumento de la aleatoriedad. La entropía aumenta con el correr del tiempo. Si aumenta la información, disminuye la entropía, pues la información es la base de la configuración y del orden.
- **Homeostasia:** Es el equilibrio dinámico entre las partes del sistema. Los sistemas tienen una tendencia a adaptarse con el fin de alcanzar un equilibrio interno frente a los cambios externos del entorno.

Una organización podrá ser entendida como un sistema o subsistema o un supersistema, dependiendo del enfoque. El sistema total es aquel representado por todos los componentes y relaciones necesarios para la realización de un objetivo, dado un cierto número de restricciones. Los sistemas pueden operar, tanto en serio como en paralelo.

### 2.2.2 Tipos de Sistemas

En cuanto a su constitución, pueden ser físicos o abstractos y en cuanto a su naturaleza pueden ser cerrados o abiertos:

- **Sistemas físicos o concretos:** Compuestos por equipos, maquinaria, objetos y cosas reales. El hardware.

- **Sistemas abstractos:** Compuestos por conceptos, planes, hipótesis e ideas. Muchas veces sólo existen en el pensamiento de la personas. Es el software.
- **Sistemas cerrados:** No presentan intercambio con el medio ambiente que los rodea, son herméticos a cualquier influencia ambiente.
- **Sistemas abiertos:** Presentan intercambio con el ambiente, a través de entradas y salidas. Intercambian energía y materia con el ambiente.

## 2.3 Sistemas de información (S.I)

**Definición:** son sistemas con la capacidad de transformación de un conjunto de datos (hechos o eventos sin significado) de entrada al sistema relevante, como salida; es decir en hechos y eventos con significado.

**Organización:** es el proceso mediante el cual se crean estructuras y se mantienen, es decir, es un proceso que se usa para instituir, crear, modificar, expandir y mantener la organización de trabajo. También puede decirse que organización es un medio de obtener, localizar o asignar y mantener ayuda en una empresa.

### 2.3.1 Características de los Sistemas de Información

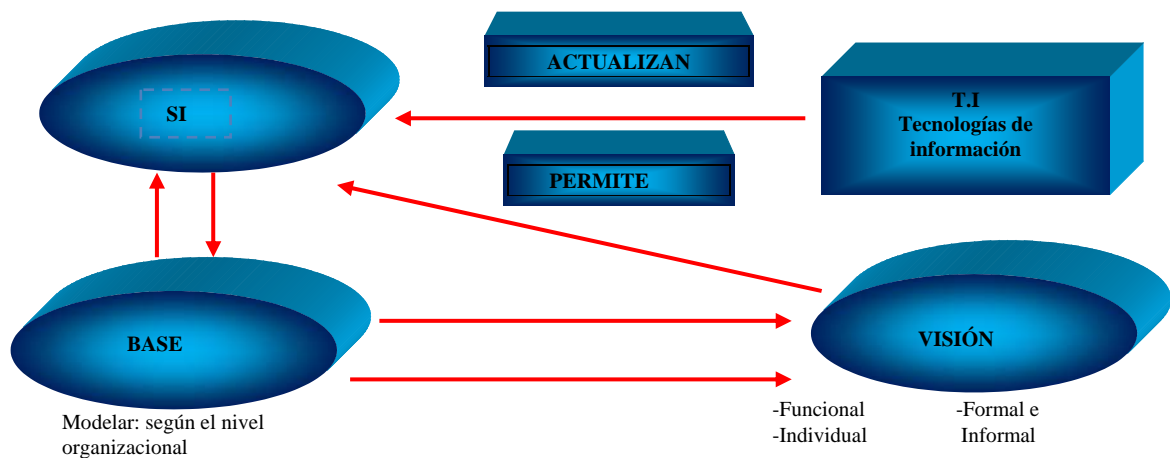
Son sistemas caracterizados por:

- Son **abiertos** porque interactúan con su ambiente mediante el intercambio de información y se **adaptan** a las necesidades del ambiente que los contiene.

- Los sistemas hombre-máquina permite el uso de máquinas (computadoras) que automaticen los procesos rutinarios de transformación los cuales son controlados y dirigidos por el hombre.
- Su entrada está constituida por datos y salida por información, el proceso de transformación de datos en información constituye la función principal del sistema de información.

### 2.3.2 Importancia de los Sistemas de Información

Radica en proporcionar información para la toma de decisiones y solución de problemas, actividades fundamentales de toda organización, a continuación se muestra como interactúan los sistemas de información figura 2.1.



**Figura 2.1** Modelo Dinámico de un Sistema de Información.

**Fuente:** Microsoft Corporation, 2001.

### 2.4 Datos e Información

Los datos son representaciones abstractas de los hechos (evento, ocurrencias o transacciones) u objetos (personas lugares etc.). Cuando estos se ordenan en un



contexto adecuado por medio de procesamiento, adquieren significado y proporcionan conocimiento sobre los hechos u objetos que los originan, transformándose en lo que se denomina información.

La información constituye el recurso esencial en proceso de toma de decisiones y en la solución de problemas en una organización.

La información posee las siguientes características:

- Exactitud.
- Forma.
- Frecuencia.
- Alcance.
- Horizonte.
- Relevancia.
- Entereza.
- Oportunidad.

#### **2.4.1 Procesamiento de Datos**

El procesamiento de datos es la actividad fundamental de un sistema de información. Consiste en transformar un conjunto de datos de entrada en información de salida y almacenar tales datos para su uso posterior. Se pueden decir que el procesamiento de datos posee dos actividades diferentes pero estrechamente relacionadas como lo son:

- **Procesamiento de transacciones:** Se encarga de realizar funciones como la calcificación de datos, ordenamiento de datos, cálculo, resumen, almacenamiento y captura.

- **Procesamiento de información:** Se encarga de realizar funciones como selección, reducción, conversión, presentación y distribución.

## **2.5 Componentes de un Sistema de Información**

Los sistemas de información poseen componentes físicos y funcionales.

### **2.5.1 Componentes Físicos**

- Sub sistema del computador.
- Sub sistema de personal-Recurso Humano.
- Sub sistema programado y aplicaciones.
- Sub sistema de datos.

### **2.5.2 Componentes Funcionales**

- Sub sistema de procesamiento de transacciones.
- Sub sistema de procesamiento datos.
- Sub sistema de información.

## **2.6 Funciones de un Sistema de Información**

Los sistemas de información difieren en sus tipos de entradas y salidas, en el tipo de procesamiento y en su estructura. Estos elementos están determinados con el propósito u objetivo del sistema, el cual es establecido a su vez, por la organización.

A pesar de las diferencias que puedan existir entre distintos sistemas de información, en todos ellos podemos encontrar un conjunto de funciones:

1. Procesamiento de transacciones.
2. Definición de archivos.
3. Mantenimiento de archivos.
4. Generación de reportes.
5. Procesamiento de consultas.
6. Mantenimiento de la integridad de los datos.

### **2.7 Ciclo de Vida de un Sistema de Información**

Un sistema de información al igual seres vivientes muestra un proceso de nacimiento y muerte bien definido.

Un sistema da información tiene un origen (nacimiento), generalmente ocasionado por necesidades, a partir del cual se emprende su desarrollo que va desde la definición del proyecto hasta la puesta en operación (crecimiento); seguidamente se inicia su operación y mantenimiento por un periodo mayor a los demás durante el cual alcanza el máximo rendimiento posible (maduración); luego, factores tales como la dinámica de organización, los avances tecnológicos y las regiones externas o internas vuelven obsoleto e ineficaz al sistema (decaimiento); lo cual origina su paralización (muerte). En este último periodo se toma la decisión de renovar el sistema, lo cual origina un nuevo ciclo de vida, o desecharlo por completo, lo cual marca su fin definitivo.

### **2.8 Ciclo de Desarrollo de un Sistema de Información**

A continuación se presenta un resumen de la metodología utilizada por IBM/AD Cycle para el desarrollo de sus sistemas de información.

### **Fase I Requerimientos.**

Comprende actividades de modelaje del negocio y planificación de sistemas. Esta fase se puede realizar para uno o todos los subsistemas del negocio. Permite identificar los procesos del sistema y como se realizan. Esta fase también comprende lo referente a la información que se necesita, las reglas, políticas y prácticas del negocio a seguir.

#### **El plan del sistema debe contener:**

1. Los sistemas a considerar.
2. La descripción de cada sistema.
3. Tecnología a emplear.
4. Decisiones referente a sí el sistema se comprará o se diseñará.
5. Problemas detectados y las oportunidades de mejorarlo.
6. Un análisis de los beneficios del nuevo sistema de información.

### **Fase II: Análisis.**

Esta fase comprende Actividades destinadas a analizar y diseñar los procesos, condicionado al plan de sistemas y modelado de datos.

¿Cómo utilizar los componentes físicos?

¿Cómo se diseñaran los componentes funcionales?

Evaluar paquetes.

### **Fase III: Construcción.**

1. Preparar la infraestructura.
2. Desarrollo de las unidades de diseño interactivo.

3. Desarrollo de las unidades de diseño.
4. Adaptar paquetes.
5. Procedimientos administrativos.

#### **Fase IV: Pruebas.**

1. De función.
2. De sistema.
3. De integración.
4. De aceptación.

#### **Fase V: Producción y Mantenimiento.**

1. Entrenamiento a usuarios.
2. Entonación del sistema.
3. Realización de los manuales definitivos.
4. Entrega de manera definitiva el sistema a la parte interesada.

## **2.9 Desarrollo de software**

La realización de cualquier sistema computarizado, que busca no solo solucionar una necesidad sino perdurar y evolucionar, requiere del uso de ciertos conocimientos teóricos, así como de técnicas de diseño, metodologías y herramientas.

### **2.9.1 Ingeniería de Software**

Es la disciplina tecnológica y administrativa dedicada a la producción sistemática de productos de programación, que son desarrollados y modificados a tiempo y dentro de un presupuesto definido.

## 2.9.2 Fases de la Ingeniería de Software

El desarrollo de toda aplicación informática, normalmente se hace cumpliendo un ciclo de desarrollo del software, el cual está dividido en las siguientes fases (Kendall y Kendall, 1997).

**Análisis y Especificación de los Requerimientos:** Esta fase consiste en analizar, entender y registrar el problema que el patrocinante está tratando de resolver. Los requerimientos son una descripción de las necesidades o deseos de un producto. El reto consiste en definirlos de manera inequívoca, de modo que se detecten los riesgos y no se presenten sorpresas al momento de entregarlo. Un conjunto de requerimientos en estado de madurez, deben presentar una serie de características tanto individualmente como en grupo, así tenemos que un requerimiento debe ser:

**Necesario:** Si su omisión provoca una deficiencia en el sistema a construir, y además su capacidad, características físicas o factor de calidad no pueden ser reemplazados por otras capacidades del producto o proceso.

**Conciso:** Si es fácil de leer y entender. Su redacción debe ser simple y clara para aquellos que vayan a consultarlo en un futuro.

**Completo:** Si no necesita ampliar detalles en su redacción, es decir, si se proporciona la información suficiente para su comprensión.

**Consistente:** Si no es contradictorio con otro requerimiento.

**No ambiguo:** Cuando tiene una sola interpretación.

**Diseño:** Consiste en crear una solución que satisfaga las especificaciones definidas en la fase de análisis. En esta fase se crea la interfaz de usuario, estructura del software.

**Codificación:** Esta fase corresponde con la implementación física de las bases de datos, de los programas (codificación), prueba de unidad o procedimientos separados y prueba de subsistemas o integración.

**Prueba del sistema:** Los grupos de procedimientos de unidades probadas como elementos separados en la fase previa son comprobados como sistema durante la integración.

**Instalación:** Corresponde a la fase de puesta en marcha de la aplicación.

**Mantenimiento del sistema y mejoras:** Esta fase se corresponde al mantenimiento y mejoras futuras que deben contemplarse a la hora de implantar una aplicación informática, una vez que el sistema está en funcionamiento.

## 2.10 Paradigma de programación orientado a objetos

Para entender claramente el paradigma de programación orientado a objetos debe tenerse conocimiento previo de los siguientes conceptos:

Un paradigma de programación es una colección de modelos conceptuales que juntos modelan el proceso de diseño y determinan la estructura de un programa.

La orientación a Objetos (OO), puede describirse como el conjunto de disciplinas (ingeniería) que desarrollan y modelan software; que facilita la construcción de sistemas complejos a partir de componentes, llamados objetos.

Los sistemas basados en modelos de datos OO se originaron con el paradigma de programación orientada a objeto:

En la programación Orientada a Objetos (OO), todo se organiza en torno a los objetos. Un objeto tiene un nombre y posee datos y procedimientos. En cierto sentido, un objeto es un programa virtual. Dado ciertos datos de entrada, el objeto aplica los procedimientos sobre dichos datos y genera una salida.

Por consiguiente se tiene que: un objeto es una abstracción con límites bien definidos y con significado a efectos del problema que se tenga entre manos.

Los objetos poseen las siguientes características básicas:

Interactúan a través de mensajes. Un mensaje estimula la ocurrencia de cierto comportamiento en el objeto receptor.

Encapsulan datos y los algoritmos que procesan estos datos. Estos algoritmos son llamados **operaciones, métodos o servicios** y definen el comportamiento de un objeto y cambian, de alguna manera, los atributos de dicho objeto.

Más correctamente, una operación cambia valores de uno o más atributos contenidos en el objeto.

Aunque existen muchos tipos diferentes de operaciones, estas pueden clasificarse en tres grandes categorías:

1. Operaciones que manipulan, de alguna manera, datos.
2. Operaciones que realizan algún cálculo
3. Operaciones que motorizan un objeto frente a la ocurrencia de un suceso de control.



La programación orientada a objeto OO: es la disciplina de ingeniería de desarrollo y modelado de software que permite construir más fácilmente sistemas complejos a partir de componentes individuales. Objetos + Mensajes = Programa.

El paradigma de programación OO incluye el concepto de tipos de datos abstractos en lenguajes de programación. Las declaraciones de tipos de datos abstractos explícitamente se definen públicos y privadas en algunas porciones de la estructura de datos, u objetos. Los tipos de datos abstractos en lenguaje orientado a objeto, son llamados clases, es decir encapsula porciones privadas de datos del objeto con procedimientos públicos, llamados métodos. El argumento para encapsulación es uno de los más simples en la construcción y mantenimiento de programas a través de modularización.

Un objeto es como una caja negra, que puede ser construida y modificada independientemente del resto del sistema, tan grande como una interfaz pública (método) en la cual las definiciones no cambian.

### 2.10.1 Modelado de Objetos

**Concepto de modelo:** Esquema teórico abstracto de un sistema extraído de la realidad compleja, elaborado para facilitar su comprensión y estudio. En el modelado existen cuatro características fundamentales que ayudan a definir un objeto:

**Abstracción:** Denota las características esenciales de un objeto que lo distinguen de todos los demás tipos objeto, y proporciona así fronteras conceptuales nítidamente definidas respecto a la perspectiva del observador". Una abstracción se centra en la visión externa de un objeto, y, por tanto sirve para separar el comportamiento esencial de un objeto de su implantación.

**Modularidad:** Se basa en el concepto de fragmentación de los programas en componentes individuales para reducir su complejidad en algún grado, y para crear además una serie de fronteras bien definidas y documentadas dentro del programa, dónde estas fronteras o interfaces tienen un incalculable valor cara a la comprensión del programa.

**El polimorfismo:** Significa que una misma operación puede comportarse de modos distintos en distintas clases. El polimorfismo permite que un número de operaciones diferentes tengan el mismo nombre. Esto reduce el acoplamiento entre objetos, haciendo a cada uno más independiente.

**El encapsulamiento:** Significa que toda la información de un objeto se encuentra empaquetada bajo un nombre y puede reutilizarse como una especificación o componente de un programa. El encapsulado es el resultado (o acto) de ocultar los detalles de implantación de un objeto respecto de su usuario.

## 2.11 El Proceso Unificado de Desarrollo de Software

El Proceso Unificado como su nombre lo indica es un proceso que unifica el desarrollo de software mediante un conjunto de actividades necesarias para transformar los requisitos de un usuario o de un grupo de estos en un sistema de software.

No obstante es bueno tener claro que el proceso unificado posee la virtud de poder especializarse y adaptarse a diferentes sistemas de software, a diferentes tipos y modelos de organizaciones y distintos proyectos.

Por otro lado el Proceso Unificado para su aplicación se sustenta o fundamenta en una herramienta que es esencial para tal fin, el *Lenguaje Unificado de Modelado* (Unified Modeling Language, UML), ésta es sumamente versátil y poderosa para la

elaboración y diseño de los diferentes planos, trazas o perspectivas del sistema software.

Pero, a decir verdad la naturaleza única e integral del Proceso Unificado se sostiene sólidamente en sus tres principios básicos, que le dan sencillamente un carácter metódico y organizado, estos paradigmas son:

**Dirigidos por casos de uso:** Los casos de uso dirigen y controlan el proceso de desarrollo en su totalidad.

**Centrado en la arquitectura:** Es la pieza clave que permite comprender el sistema, organizar el desarrollo y hacer evolucionar el software.

**Proceso iterativo e incremental:** El desarrollo se plantea de manera progresiva, de tal modo que se atenúen los riesgos y se planteen las cuestiones en el instante que estamos capacitados para resolverlas. Cada avance realizado en el proyecto se cuenta como una iteración, esto trae como consecuencias que una gran aplicación se descomponga en piezas más pequeñas, lo que permite visualizar la mayor parte de los detalles inherentes al desarrollo y crecimiento de la aplicación, garantizando así, que es trabaje en forma más segura y eficiente.

### **2.11.1 La vida del Proceso Unificado de Desarrollo de Software**

Al igual que la vida de cualquier ser vivo o ente que recorre una serie de etapas o ciclos como nacimiento, desarrollo y muerte asimismo sucede con el Proceso Unificado que recorre una serie de ciclos que constituyen la vida de un sistema. Precisamente cada uno de estos ciclos se constituye por cuatro fases a saber: inicio, elaboración, construcción y transición. Luego cada fase se subdivide en iteraciones, y

finalmente cada iteración no es más que la ejecución de los cinco flujos de trabajo (requisitos, análisis, diseño, implementación y prueba).

Entonces así, cada ciclo finaliza con una versión del sistema o producto.

### **2.11.2 Fases del Proceso Unificado de Desarrollo de Software**

Como se dijo anteriormente cada ciclo de la vida del software se subdivide en cuatro fases (inicio, elaboración, construcción y transición). Cada fase luego se descompone en iteraciones con sus respectivos incrementos. Cada fase concluye con la disponibilidad de un conjunto de modelos o documentos desarrollados hasta conseguir un estado predefinido, llamado hito, que permite la toma de decisiones importantes para así continuar la fase siguiente, permite además controlar el progreso del trabajo según pasa por cada fase. A través del seguimiento del tiempo y esfuerzo aplicado en cada fase se consigue un conjunto de datos útiles para la estimación de tiempo y recursos para otros proyectos, además de su asignación tomando en cuenta el tiempo de duración.

#### **2.11.2.1. Fase de Inicio**

El objetivo general de esta fase es poner en marcha el proyecto de desarrollo, pero antes se debe desarrollar el análisis del negocio lo cual lleva a delimitar el alcance (ámbito) del sistema, el ámbito es necesario para comprender lo que debe cubrir la arquitectura y así ubicar y enfrentar tempranamente los riesgos dentro de los límites del sistema.

Se debe establecer una arquitectura que soporte el ámbito del sistema, lo que se conoce por “arquitectura candidata”. En esta fase se determinan las principales funciones del sistema mediante un modelo de casos de uso que comprende los casos

de uso más críticos, además de identificar y dar prioridad a los riesgos más importantes, estas últimas son las actividades más relevantes de la fase de inicio.

### **2.11.2.2 Fase Elaboración**

Para esta fase se recopilan la mayor parte de los requisitos funcionales, aun pendientes, mediante los casos de uso y se diseña la arquitectura del sistema. Dicha arquitectura se expresa a través de las vistas de todos los modelos del sistema, es decir, vistas arquitectónicas del modelo de casos de uso, del modelo de análisis, del modelo de diseño, del modelo de implementación y modelo de despliegue. La resultante de esta fase es la línea base de la arquitectura. Al término de esta fase se tiene la ventaja de que es posible planificar las actividades y estimar los recursos necesarios para ultimar el proyecto. Consecuentemente se aclaran dudas acerca de la suficiente estabilidad de los casos de uso, de la arquitectura y el plan; además de discernir sobre si están completamente controlados los riesgos como para seguir en el esfuerzo de desarrollo.

### **2.11.2.3. Fase de Construcción**

La idea fundamental de esta fase es culminar y entregar un producto software en su versión operativa inicial conocida como “versión beta”. El producto debería poseer la calidad adecuada y la seguridad de cumplir los requisitos para así emprender su aplicación en la comunidad de usuarios. Para alcanzar todo lo anterior se deben detallar todos los casos de uso y escenarios restantes, modificar si es pertinente la descripción de la arquitectura, dejar cerrados los modelos de análisis, diseño e implementación continuados en los flujos de trabajo de las iteraciones adicionales. Además integrar los subsistemas y someterlos a prueba y hacer lo propio con todo el sistema.

Finalmente en esta fase el producto contiene todos los casos de uso que la dirección y el cliente han convenido en desarrollar para la versión beta. Pero es aquí donde se establece y aclara la expectativa acerca de si cubre suficientemente el producto desarrollado, las necesidades de algunos usuarios como para hacer realidad la entrega de la versión inicial.

#### **2.11.2.4. Fase de Transición**

Esta fase tiene como objetivos fundamentales, cumplir con todos los requisitos establecidos en fases anteriores, hasta satisfacer a todos los usuarios y preparar todos los aspectos relativos a la operación del sistema en el contexto del usuario, lo que implica la corrección de los posibles defectos acusados por estos en la versión beta.

La fase de transición abarca actividades como la fabricación, formación de cliente, entrega de una línea de ayuda y asistencia, y la corrección de defectos encontrados posterior a la entrega del software.

### **2.12 Base de datos**

Es un conjunto de datos almacenados de forma integrada y compartida.

Integrada ya que es un conjunto de archivos independientes, donde se elimina y se reduce la redundancia.

Compartida ya que varios usuarios diferentes pueden acceder a la misma fracción de la BD, incluso al mismo tiempo y utilizarla con fines diferentes.

Las bases de datos proporcionan la infraestructura requerida para los Sistemas de Apoyo a la Toma de Decisiones y para los Sistemas de Información Estratégicos,

ya que estos sistemas explotan la información contenida en las bases de datos de la organización para apoyar el proceso de toma de decisiones o para lograr ventajas competitivas.

### **2.12.1. Ventajas de las Bases de Datos**

La utilización de bases de datos como plataforma para el desarrollo de sistemas de aplicación en las organizaciones se ha incrementado notablemente en los últimos años, se debe a las ventajas que ofrece su utilización, algunas de las cuales se comentarán a continuación:

**Globalización de la información:** Permite a los diferentes usuarios considerar la información como un recurso corporativo que carece de dueños específicos.

**Eliminación de información inconsistente:** Debe haber coherencia, si existen dos o más archivos con la misma información, los cambios que se hagan a éstos deberán hacerse a todas las copias del archivo de facturas.

**Permite compartir información:** La información almacenada puede ser compartida por un gran número de usuarios.

**Permite mantener la integridad en la información:** La integridad de la información es una de sus cualidades altamente deseable y tiene por objetivo que sólo se almacena la información correcta.

**Independencia de datos:** El concepto de independencia de datos es quizás el que más ha ayudado a la rápida proliferación del desarrollo de Sistemas de Bases de Datos. La independencia de datos implica un divorcio entre programas y datos.

## **2.12.2 Sistema de Manejador de Base de Datos**

Es un conjunto de programas que se encargan de manejar la creación y todos los accesos a las bases de datos. Se compone de un lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta.

### **2.12.2.1. Objetivos del SMBD**

- Independencia de los datos.
- Seguridad.
- Reserva.
- Integridad.
- Respaldo.
- Recuperación.
- Redundancia mínima.
- Consistencia.
- Capacidad e auditoria.
- Control de concurrencia.
- Desempeño.

## **2.13 Diseño de Una Base de Datos**

Existen distintos modos de organizar la información y representar las relaciones entre los datos en una base de datos. Los Sistemas administradores de bases de datos convencionales usan uno de los tres modelos lógicos de bases de datos para hacer seguimiento de las entidades, atributos y relaciones. Los tres modelos lógicos principalmente de bases de datos son el jerárquico, de redes y el relacional. Cada modelo lógico tiene ciertas ventajas de procesamiento y también ciertas ventajas de negocios.



**Modelo Jerárquico de Datos:** Es una clase de modelo lógico de bases de datos que tiene una estructura arborescente. Un registro subdivide en segmentos que se interconectan en relaciones padre e hijo y muchos más. Los primeros sistemas administradores de bases de datos eran jerárquicos. Puede representar dos tipos de relaciones entre los datos: relaciones de uno a uno y relaciones de uno a muchos.

**Modelo de Datos en Red:** Es una variación del modelo de datos jerárquico. De hecho las bases de datos pueden traducirse de jerárquicas a en redes y viceversa con el objeto de optimizar la velocidad y la conveniencia del procesamiento. Mientras que las estructuras jerárquicas describen relaciones de muchos a muchos.

**Modelo Relacional de Datos:** Es el más reciente de estos modelos, supera algunas de las limitaciones de los otros dos anteriores. El modelo relacional de datos representa todos los datos en la base de datos como sencillas tablas de dos dimensiones llamadas relaciones.

Las tablas son semejantes a los archivos planos, pero la información en más de un archivo puede ser fácilmente extraída y combinada.

## 2.14 PHP

PHP es un lenguaje de scripting que permite la generación dinámica de contenido en un servidor Web. Su nombre oficial es: Hypertext Processo (PHP). Entre sus principales característica se puede destacar su potencia, alto rendimiento y facilidad de aprendizaje. PHP es una eficaz herramienta de desarrollo para los programadores Web, ya que permite la creación de sitio Web dinámico.

## **2.15 Tecnología Web**

### **2.15.1 Concepto**

Red diseñada para el procesamiento de información dentro de una compañía u organización. Entre sus usos se incluyen servicios tales como distribución de documentos, distribución de software, acceso a bases de datos y aprendizaje. Las intranets deben su nombre a que en ellas se utilizan a menudo aplicaciones asociadas a Internet, tales como páginas Web, sitios FTP, correo electrónico, grupos de noticias y listas de distribución, a las cuales únicamente se pueden tener acceso a los terminales de la propia compañía u organización.

## **2.16 Protocolo http**

### **2.16.1 Concepto**

El protocolo HTTP (*Hypertext Transfer Protocol*) es el protocolo base de la W.W.W. Se trata de un protocolo simple, orientado a conexión y sin estado.

La razón de que esté orientado a conexión es que emplea para su funcionamiento un protocolo de comunicaciones (TCP, *Transport Control Protocol*) de modo conectado, un protocolo que establece un canal de comunicaciones de extremo a extremo (entre el cliente y el servidor) por el que pasa el flujo de bytes que constituyen los datos que hay que transferir, en contraposición a los protocolos de datagrama o no orientados a conexión que dividen los datos en pequeños paquetes (datagramas) y los envían, pudiendo llegar por vías diferentes del servidor al cliente.

El protocolo no mantiene estado; es decir, cada transferencia de datos es una conexión independiente de la anterior, sin relación alguna entre ellas, hasta el punto

de que para transferir una página Web tenemos que enviar el código HTML del texto, así como las imágenes que la componen, pues en la especificación inicial de HTTP, la 1.0, se abrían y usaban tantas conexiones como componentes tenía la página, transfiriéndose por cada conexión un componente (el texto de la página o cada una de las imágenes).

## **2.17 Servidor Apache Web**

### **2.17.1 Concepto**

El servidor Web Apache es un esfuerzo por desarrollar y mantener un servidor HTTP de código abierto para sistemas operativos modernos. El objetivo del proyecto es proveer un servidor Web seguro, eficiente y extensible que proporcione servicios HTTP en sincronización con los estándares HTTP actuales.

El servidor Web Apache es un proyecto de la fundación de software Apache (*Apache Software Foundation*) y ha sido el servidor Web más popular en Internet desde Abril de 1996.

### **2.17.2 Ventajas del Servidor HTTP Apache**

- Modular.
- Open source.
- Multi-plataforma.
- Extensible.
- Popular (facilidad para conseguir ayuda/suporte).

## **2.18 Navegador Web**

### **2.18.1 Concepto**

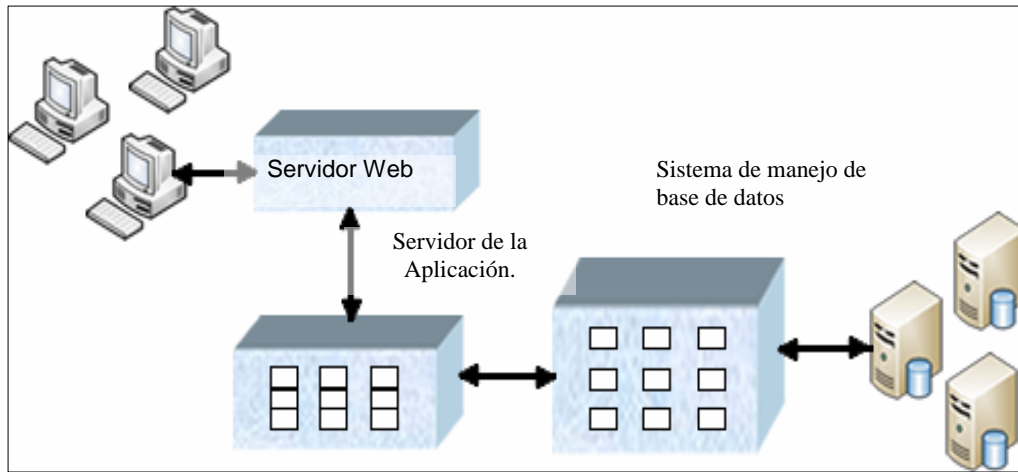
Un navegador Web, es una aplicación que permite al usuario recuperar y visualizar documentos de hipertexto, comúnmente descritos en HTML, desde servidores Web de todo el mundo a través de Internet o localmente a través de una Intranet. Esta red de documentos es denominada (WWW, *World Wide Web*) o Telaraña Mundial. Los navegadores actuales permiten mostrar y/o ejecutar: gráficos, secuencias de vídeo, sonido, animaciones y programas diversos además del texto y los hipervínculos o enlaces.

### **2.18.2 Estructura de las Aplicaciones Web**

Una aplicación Web está normalmente estructurada como una aplicación de tres (3) capas. En su forma más común, el navegador Web ofrece la primera capa y un motor de ejecución de programas capaz de usar alguna tecnología Web dinámica (ejemplo: PHP, Java Servlets o ASP, ASP.NET, entre muchos otros).

El servidor de aplicación es el requerido para la ejecución de la aplicación. Por último, una base de datos instalado en los servidores físicos constituye el último bloque del proceso, ver Figura 2.3.

El navegador Web manda peticiones al servidor de la aplicación, este ofrece servicios valiéndose de consultas y actualizaciones a la base de datos y a su vez proporciona una interfaz de usuario.



**Figura 2.3** Estructura General de una Aplicación Web.

**Fuente:** Microsoft Corporation, 2001.

## 2.19 MySQL

Es un sistema de gestión de base de dato relacional. Su diseño multihilo le permite soportar una gran carga de forma muy eficiente.

MySQL fue creada por la empresa sueca MaySQLLAB que mantiene el copyright del código fuente del servidor SQL, así como también la marca.

## 2.20 WAMP

Es el acrónimo usado para describir un sistema de infraestructura de internet que usa las siguientes herramientas:

- **Windows**, como sistema operativo;

- **A**pache, como servidor Web;
- **M**ySQL, como gestor de bases de datos;
- **P**HP (generalmente), **P**erl, o Python, como lenguajes de programación.

El uso de un WAMP permite servir páginas html a internet, además de poder gestionar datos en ellas, al mismo tiempo un WAMP, proporciona lenguajes de programación para desarrollar aplicaciones Web.



*Hospital de Niños “Dr. Rafael Tobías Guevara”.  
Barcelona. Estado Anzoátegui.*

## CAPÍTULO III

### MARCO METODÓLOGICO

## **CAPÍTULO III**

### **MARCO METODOLÓGICO**

#### **3.1 Tipos de Investigación.**

Existen muchos modelos y formas de clasificarlos. Sin embargo, lo importante es precisar los criterios de clasificación (Arias, 2006).

En este sentido se identifican:

- Tipos de investigación según el Nivel.
- Tipos de investigación según el Diseño.
- Tipos de investigación según el Propósito.

Sin embargo, independientemente de su clasificación, todos son tipos de investigación, y al no ser excluyentes, un estudio puede ubicarse en más de una clase.

##### **3.1.1 Nivel de la Investigación.**

El nivel de investigación se refiere al grado de profundidad con que se aborda un fenómeno u objeto de estudio.

El nivel de investigación de este proyecto es del tipo Descriptivo.

La investigación descriptiva consiste en la caracterización de un hecho, fenómeno, individuo o grupo, con el fin de establecer su estructura o



comportamiento. Los resultados de este tipo de investigación se ubican en un nivel intermedio en cuanto a la profundidad de los conocimientos se refiere.

### **3.1.2 Diseño de la investigación.**

El diseño de la investigación es la estrategia general que adopta el investigador para responder al problema planteado. En atención al diseño la investigación se clasifica en:

Documental.

De campo.

Experimental.

El diseño de investigación de este proyecto es de campo.

La investigación de campo es aquella que consiste en la recolección de datos directamente de los sujetos investigados, o de la realidad donde ocurren los hechos (datos primarios), sin manipular o controlar variable alguna, es decir, el investigador obtiene información pero no altera las condiciones existentes.

### **3.2 Técnicas a utilizar.**

Se entenderá por técnica, el procedimiento o forma particular de obtener datos o información.

Como el diseño de esta investigación es de campo, las técnicas a utilizar serán las siguientes:

**La observación:** Es una técnica que consiste en visualizar o captar mediante la vista, en forma sistemática, cualquier hecho, fenómeno o situación que se produzca en la naturaleza o en la sociedad, en función de unos objetivos de investigación preestablecidos.

La observación puede ser estructurada o no estructurada:

**Estructurada:** Es aquella que además de realizarse en correspondencia con unos objetivos, utiliza guía diseñada previamente, en la que se especifican los elementos que serán observados.

**No estructurada:** Es la que se efectúa en función de un objeto, pero sin guía prediseñada que especifique cada uno de los aspectos que deben ser observados.

**La entrevista:** Más que un simple interrogatorio, es una técnica basada en un dialogo o conversación “cara a cara”, entre el entrevistador y el entrevistado acerca de un tema previamente determinado, de tal manera que el entrevistador pueda obtener la información requerida.

**La entrevista se clasifica en:**

- Entrevista Estructurada o Formal: Es la que se realiza a partir de una guía prediseñada que contiene preguntas que serán formuladas al entrevistado.
- Entrevista no Estructurada o Informal: En esta modalidad no se dispone de una guía de preguntas elaboradas previamente. Sin embargo, se orienta por unos objetivos preestablecidos, lo que permite definir el tema de la entrevista.

- Entrevista Semi-estructurada: Aun cuando existe una guía de preguntas, el entrevistador puede realizar otras no contempladas inicialmente. Esto se debe a que una respuesta puede dar origen a una pregunta adicional o extraordinaria.

**Técnica de diagramación:** Proceso Unificado de desarrollo para el desarrollo y el Lenguaje Unificado Modelado para la visualización y análisis de los diferentes escenarios contemplados en la metodología RUP.



*Hospital de Niños “Dr. Rafael Tobías Guevara”.  
Barcelona. Estado Anzoátegui.*

## CAPÍTULO IV

### RESULTADOS

## **CAPÍTULO IV**

### **RESULTADOS**

#### **4.1 Fase de Inicio**

La fase de inicio es la más importante del Proceso Unificado, en ella se establece una vista general de los objetivos del proyecto, determina si es viable y decide si se continuará el proyecto.

El objetivo de esta fase es conocer y analizar a fondo las actividades y procesos llevados a cabo en el hospital de niños “Dr. Tibias Guevara” en Barcelona estado Anzoátegui, a través la comunicación entre los investigadores y los usuarios.

#### **4.2 Contexto del Sistema de información de Registro y Estadístico Médico SIREM.**

La información presentada en este proyecto se obtuvo, haciendo una serie de observaciones directas al Departamento de Registros y Estadísticos Médico, específicamente en las áreas de emergencia, quirófano, hospitalización y consulta externa, las cuales son abarcadas en este proyecto.

También se realizaron entrevistas a los empleados y revisión de la documentación técnica manejada por los departamentos.

##### **❖ Requisitos Esenciales del Sistema**

- Automatizar el control del departamento de registros y estadísticos médicos del hospital de niños “Dr. Rafael Tobías Guevara”.

- Guardar en la Base de Datos toda la información referente a los pacientes y doctores, para así poder realizar reportes automatizados de cualquier índole, reduciendo y eliminando el almacenamiento de información en archivos.
- Establecer claves de acceso personal asociado, para mantener la seguridad del sistema.

### ❖ **Requisitos Funcionales**

En los requisitos funcionales se concentran aquellos que describen los servicios (funciones) que se esperan del sistema y que tengan que ver con algún proceso propiamente de gestión (crear, modificar, eliminar, consultar).

- Permitir el acceso a la base de datos a través del sistema: Inserción, modificación, eliminación y búsqueda de registros.
- Registrar, modificar y eliminar doctor.
- Registrar y modificar datos del paciente que llegan al hospital por emergencia y consulta externa.
- Actualizar datos de los pacientes que ingresen al hospital.
- Permite al usuario del sistema almacenar los datos necesarios para llevar el control y actualización en los datos del paciente en las diferentes áreas en estudio.
- Generar reportes de estadísticas asociadas con las áreas de emergencia, consulta externa, hospitalización y quirófano.

### ❖ **Requisitos no funcionales**

- La estructura y diseño del sistema debe ser escalable lo que significa que debe adaptarse fácilmente a cualquier cambio o mejora.

- Debe poseer una interfaz visual, fácil y cómoda que permite la interacción entre los usuarios y el sistema.

#### **4.2.1. Modelo del Dominio**

Un Modelo de Dominio es un diagrama de la disciplina de análisis, construido con las reglas de UML, representado como uno o más diagramas de clases y que contiene no conceptos propios de un sistema de software, sino de la propia realidad física.

Los modelos de dominio pueden utilizarse para capturar y expresar los conocimientos obtenidos en un área bajo análisis como paso previo al diseño de un sistema. Similares a los mapas mentales utilizados en el aprendizaje, el modelo de dominio es utilizado por el analista como un medio para comprender el sistema al cual se va a servir.

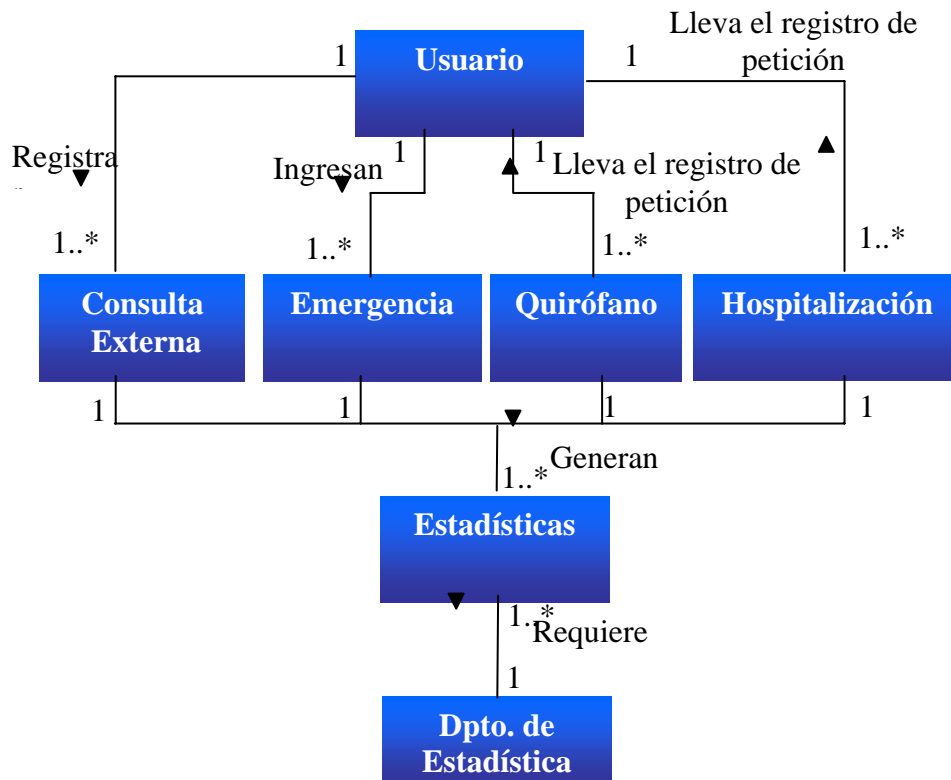
Es posible capturar un mayor grado de detalle en estos modelos de dominio, corresponde al analista decidir cuánto detalle va a ser necesario y hasta dónde llegar a modelar. El objetivo es capturar lo necesario para comprender donde va a funcionar el sistema que se está diseñando y esto demanda una cantidad distinta de detalles cada vez.

A través de esta descripción se pretende lograr una mayor comprensión del diagrama explicando paso a paso el proceso, el cual inicia de la siguiente manera: el usuario envía uno o muchos datos al área de emergencia, al área de consulta externa, al área de hospitalización y al área de quirófano. El área de emergencia envía uno o muchos datos procesado al usuario, envía uno o muchos datos a quirófano, envía uno o muchos datos a hospitalización. El área de hospitalización envía un dato a quirófano y un dato procedo al usuario. El área de quirófano envía dato al área de hospitalización y datos procesados al usuario. El área de consulta externa envía dato al área de hospitalización y dato procesado al usuario. Ver figura 4.1.

#### 4.2.2 Glosario de Términos

Para un mayor entendimiento del modelo de dominio se describirán cada uno de los términos utilizados en la figura 4.1.

- **Usuario:** es la persona encargada de ingresar, modificar los datos del paciente.
- **Consulta Externa:** es el área de consulta del paciente.
- **Hospitalización:** es el área de reclusión del paciente.
- **Emergencia:** es el área de emergencia del hospital
- **Quirófano:** es el área en donde operan a los pacientes.



**Figura 4.1** Modelo De Dominio.

**Fuente:** Elaboración propia.



El sistema de información se tituló "SIREM" (Sistema de Información para los Registro y Estadístico Médico), el cual tiene como función llevar el control de todas las actividades diarias que se realiza en dichos departamentos, para luego realizar las estadísticas que son necesarias para el manejo de la información.

En la tabla 4.1 se definen cada uno de los términos técnicos utilizados en la descripción del contexto del sistema.

### **4.2.3 Riesgos críticos**

Los riesgos críticos son aquellos que si no son mitigados pondrían en peligro el éxito del proyecto, es decir, si existe un riesgo de este tipo y no hallamos una forma de mitigarlo o un plan de emergencia para contenerlo, se debe considerar el abandono del proyecto. La tabla 4.2 muestra una lista de los posibles riesgos críticos que se podrían presentar en el desarrollo del proyecto SIREM y un plan de emergencia para mitigarlo.

#### **4.2.3.1 Identificación de los actores**

Los actores son terceros fuera del sistema que colaboran con el sistema. En la tabla 4.3 se describen todos los actores que interactúan con el sistema. SIREM.

#### **4.2.3.2 Identificación de los casos de uso**

**Procesar Doctor:** En el caso de uso que permite realizar el ingreso de los datos del doctor que laboran en el hospital.

**Procesar consulta externa:** Es el caso de uso que permite ingresar y modificar los datos del paciente, también nos permite ingresar las consulta del paciente.

**Procesar emergencia:** Es el caso de uso que nos permite ingresar los datos del paciente y también permite ingresar datos del paciente que entran por emergencia.

**Procesar hospitalización:** Es el caso de uso que nos permite ingresar datos de los paciente que ingresen al área de hospitalización.

**Tabla 4.1** Terminología utilizada

TÉRMINOS	DEFINICIONES
<b>Hospitalización</b>	Es el área que se encarga de recopila toda la información de los días de hospitalización, de ingresos, egresos de los muertos antes y después de 48 horas, egresos vivos, egresos contra opiniones medicas recluidos por diferentes servicios.
<b>Quirófano</b>	Es el área que se encarga de las intervenciones de emergencia y de los planes quirúrgicos, para luego elaborar sus estadísticas.
<b>Consulta externa</b>	Es el área que se encarga de los movimiento de consulta externa mensual, por servicios, tanto por diagnostico, emergencia, por especialidades, de paciente visto y no visto.
<b>Emergencia</b>	Es el área que se encarga de recibir todo aquellos pacientes que llegan por estados críticos, los cuales pueden ser remitidos a hospitalización o quirófano.

**Fuente:** Elaboración Propia

**Tabla 4.2** Riesgos Críticos del Sistema SIREM

IDENTIFICACION DEL RIESGO	PLAN DE EMERGENCIA
Realizar una fase de inicio que no pueda evolucionar en las fases siguientes	<ul style="list-style-type: none"> <li>• Identificar adecuadamente los casos de uso tanto funcionales como no funcionales.</li> <li>• Hacer una realización completa y detallada del flujo de trabajo de análisis en la fase de inicio.</li> <li>•</li> </ul>
No conseguir un sistema que satisfaga los requerimientos de los usuarios	<ul style="list-style-type: none"> <li>• Recopilar información confiable del sistema haciendo uso de las entrevistas y de las consultas a los usuarios.</li> <li>• Construir un prototipo del sistema que satisfaga a los usuarios.</li> </ul>

**Fuente:** Elaboración Propia

**Procesar quirófano:** Es el caso de uso que permite ingresar datos de los paciente que ingresen al área de quirófano.

**Generar Reportes:** Este caso de uso es utilizado para generar todos los reportes relacionados con las estadísticas que generan de las áreas de quirófano, hospitalización, emergencia y consulta externa.

**Tabla 4.3** Actores del sistema SIREM

ACTOR	DESCRIPCION
<b>Usuario del Sistema</b>	Llevar el control de las actividades que realiza las áreas de la misma.
<b>Manejador de la Base de Datos (MBD)</b>	Es un agente que interactúa directamente con el sistema. Representa la base de datos que contiene toda la información que maneja Sistema SIREM, la cual se encuentra distribuida en tablas.

**Fuente:** Elaboración Propia

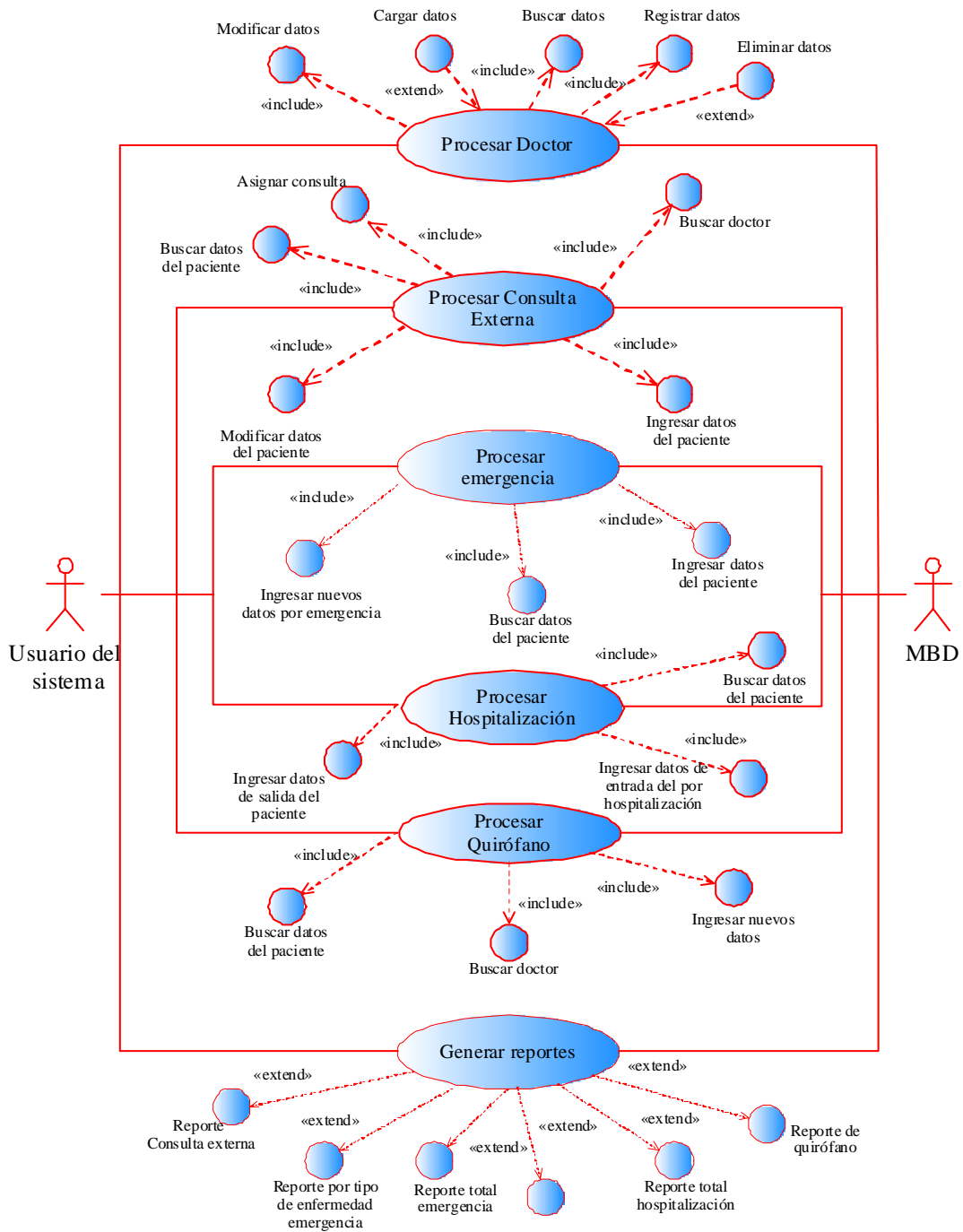
#### 4.2.4 Descripción de los casos de uso

El diagrama presentado en la figura 4.2 representa el diagrama principal de casos de uso del Sistema SIREM en el cual el usuario del sistema después de haberse instalado el software tendrá la opción de activar uno de los casos de uso descritos anteriormente, dependiendo de la actividad a realizar. El cual se va a detallar más adelante.

##### 4.2.4.1 Caso de uso de la realización del caso de uso procesar doctor

**Descripción:** Permite ingresar, modificar y eliminar los datos del doctor. Ver figura 4.3.

**Precondición:** Acceder al Sistema.



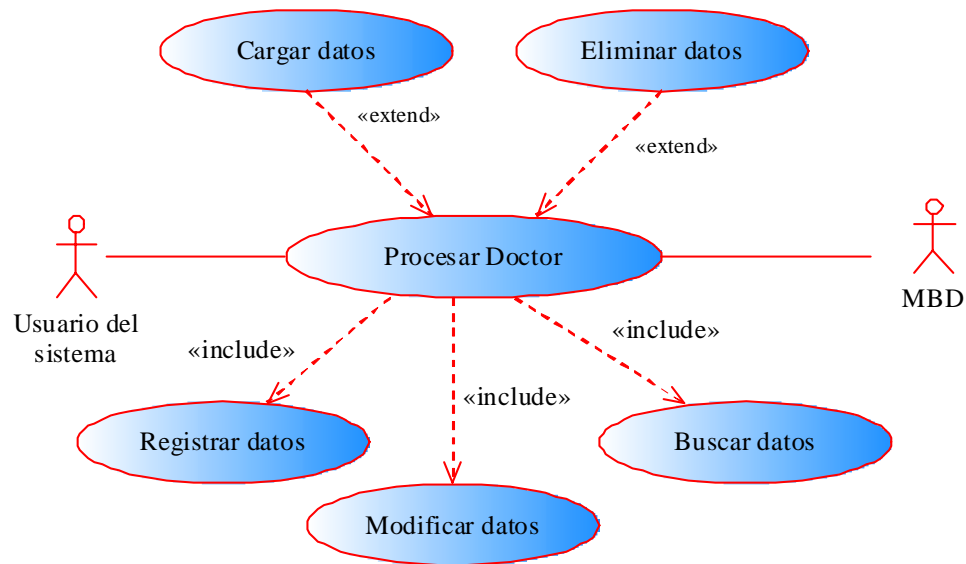
**Figura 4.2** Diagrama de casos de uso Principal del Sistema SIREM

**Fuente:** Elaboración propia.

### Flujo principal

1. El usuario ingresa al modulo del doctor.
2. Se ingresan, modifican o eliminan los datos del doctor.
3. Fin del caso de uso.

**Post condición:** Se registra, actualiza o elimina los datos del doctor.



**Figura 4.3** Diagrama de casos de uso de Procesar Doctor

**Fuente:** Elaboración propia.

#### 4.2.4.2 Caso de uso de la realización del Caso de Uso Procesar Consulta Externa

**Descripción:** Permite ingresar, modificar los datos del paciente o asignar una consulta. Ver figura 4.4.

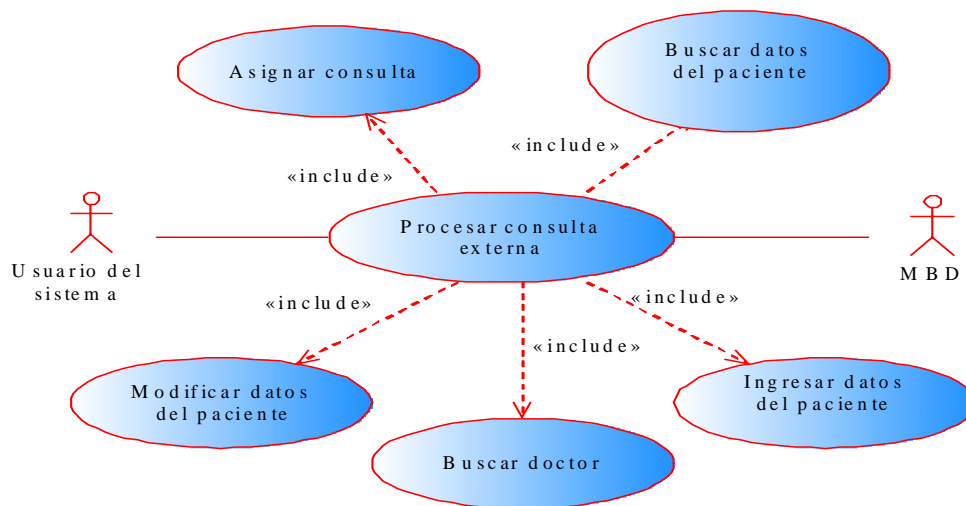
**Precondición:** Acceder al Sistema.

### Flujo principal

1. El usuario ingresa al modulo de consulta externa.
2. Registra o modifican los datos del paciente.

3. En el caso de asignar cita se busca la disponibilidad del médico y consulta.
4. Fin del caso de uso.

**Post condición:** Se registra o modifica los datos del paciente, también se asignan consulta.



**Figura 4.4** Diagrama de casos de uso de Procesar Consulta Externa

**Fuente:** Elaboración propia

#### 4.2.4.3 Caso de uso de la realización del Caso de Uso Procesar Emergencia

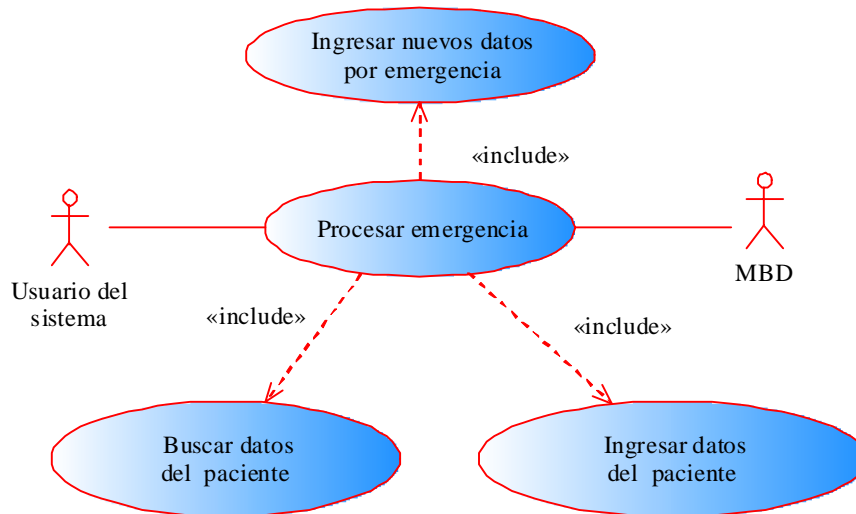
**Descripción:** Permite ingresar los datos del paciente por primera vez o nuevos datos requerido si el paciente existe. Ver figura 4.5.

**Precondición:** Acceder al Sistema.

##### Flujo principal

1. El usuario ingresa al modulo de emergencia.
2. Registra datos del paciente.
3. En el caso de que el paciente existe se ingresan nuevos datos requerido.
4. Fin del caso de uso.

**Post condición:** Se registra los datos del paciente o ingresan nuevos datos.



**Figura. 4.5** Diagrama de casos de uso Procesar Emergencia

**Fuente:** Elaboración propia.

#### 4.2.4.4 Caso de uso de la realización del Caso de Uso Procesar Hospitalización

**Descripción:** Permite ingresar nuevos datos requeridos ya sea por entrada o salida del paciente. Ver figura 4.6.

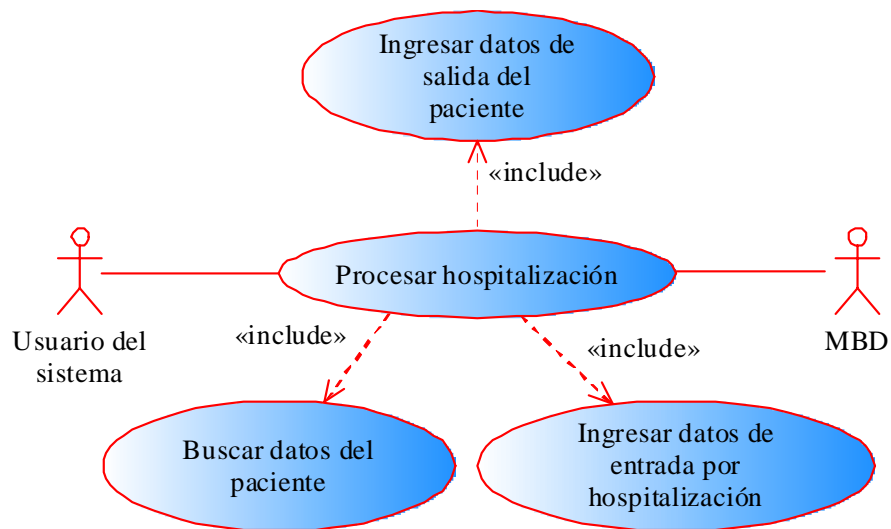
**Precondición:** Acceder al Sistema.

##### Flujo principal

1. El usuario ingresa al modulo de hospitalización.
2. Busca y registra los nuevos datos del paciente.
3. Fin del caso de uso.

**Post condición:** Se registra los datos de salida o entrada del paciente.





**Figura 4.6** Diagrama de casos de uso de Procesar Hospitalización

**Fuente:** Elaboración propia.

#### 4.2.4.5 Caso de uso de la realización del Caso de Uso Procesar Quirófano

**Descripción:** Permite ingresar nuevos datos requerido. Ver figura 4.7.

**Precondición:** Acceder al Sistema.

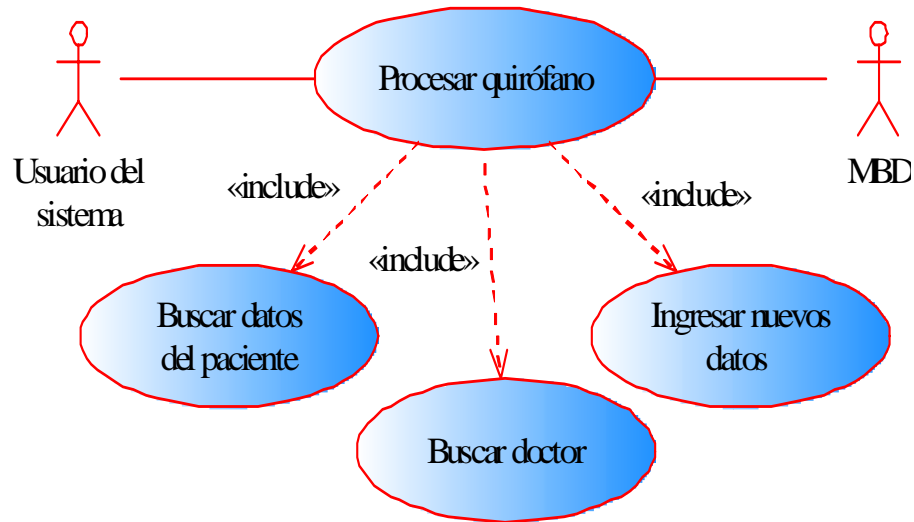
##### **Flujo principal**

1. El usuario ingresa al modulo de quirófano.
2. Registra nuevos datos del paciente.
3. Fin del caso de uso.

**Post condición:** Se registra los nuevos datos del paciente.

### 4.3 Análisis

En esta fase, se pretende hacer un estudio de una forma muy general de los casos de usos más importantes del sistema "SIREM", sin llegar al detalle profundo de estos casos de uso, utilizando un lenguaje formal como lo es UML.



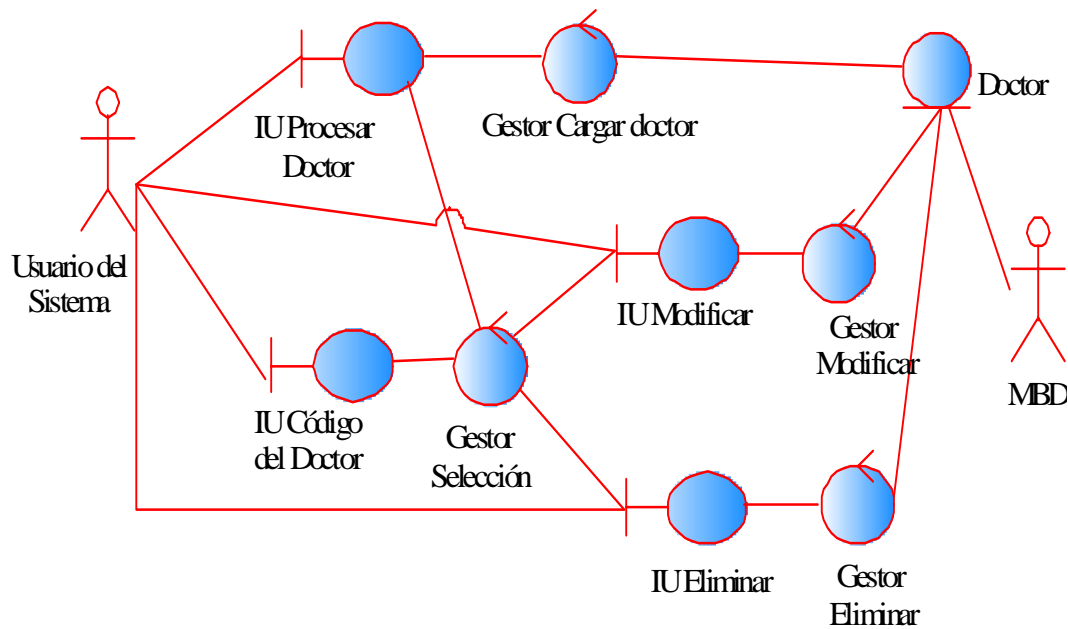
**Figura. 4.7** Diagrama de casos de uso de Procesar Quirófano

**Fuente:** Elaboración propia

El resultado de esta fase es un modelo inicial de análisis, el cual servirá de guía para estructurar al sistema en conjunto incluyendo su arquitectura. Dentro de este modelo, los casos de usos se describen mediante diagramas de clases de análisis y de colaboración, los cuales se representan utilizando tres estereotipos: clases de entidad, clase de interfaz y clase de control.

#### 4.3.1 Diagrama de clases de análisis de la realización del caso de uso Procesar Doctor

La figura 4.8 muestra el diagrama de clases de análisis de la realización del caso de uso Procesar Doctor, el cual puede ser activado por el actor usuario del sistema mediante la interfaz Doctor, que permite que el usuario del sistema pueda registrar, modificar y eliminar los datos del doctor.



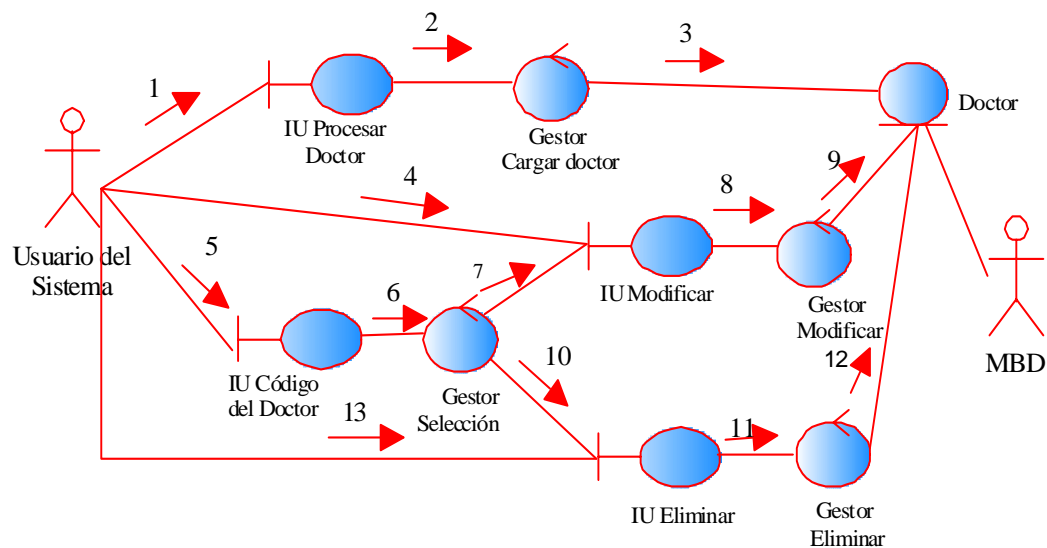
**Figura 4.8** Diagrama de clases de análisis de la realización del Caso de Uso  
Procesar Doctor

**Fuente:** Elaboración Propia

#### 4.3.2 Diagrama de clases de colaboración de la realización del Caso de Uso Procesar Doctor

Mediante la interfaz doctor el usuario del sistema podrá ingresar, modificar y eliminar los datos del doctor.

Los datos que deben ingresar son: nombre, apellido, cedula y la especialidad. Esto se hace a través del objeto 1. En caso de eliminar o modificar se pedirá el código del doctor, el único dato del doctor que puede modificar es la especialidad. Ver figura 9.



#### Leyenda

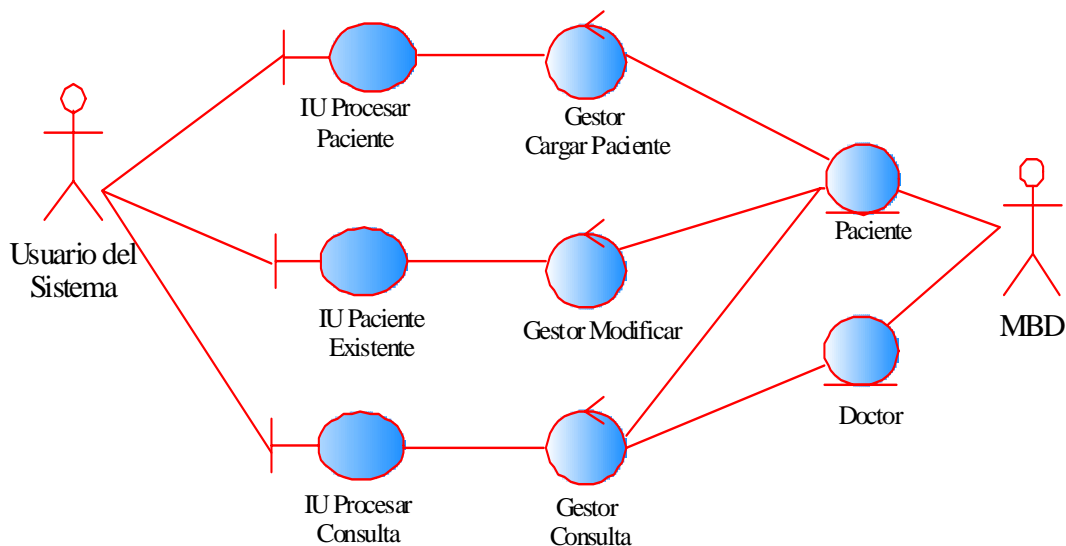
- |   |  |
|---|--|
| 1 Activa el proceso para ingresar los datos del doctor. | 8 Permite ingresar los datos del doctos a modificar. |
| 2 Procesa los datos.                                    | 9 Almacena los datos modificados.                    |
| 3 Almacena los datos.                                   | 10 Muestra todos los datos del doctor a eliminar.    |
| 4 Activa el proceso modificar doctor                    | 11 Elimina los datos del doctor                      |
| 5 Activa el proceso para ingresar el código del doctor. | 12 Actualiza la base de datos.                       |
| 6 Gestiona la selección a tomar.                        | 13 Activa el proceso para eliminar al doctor         |
| 7 Muestra los datos del doctor.                         |  |

**Figura 4.9** Diagrama de clases de colaboración de la realización del Caso de Uso Procesar Doctor

**Fuente:** Elaboración Propia

#### 4.3.3 Diagrama de clases de análisis de la realización del caso de uso Procesar Consulta Externa

La figura 4.10 muestra el diagrama de clases de análisis de la realización del caso de uso Procesar consulta externa, el cual puede ser activado por el actor usuario del sistema mediante la interfaz Consulta Externa, que permite que el usuario del sistema pueda registra, modificar los datos del paciente y asignar consulta.



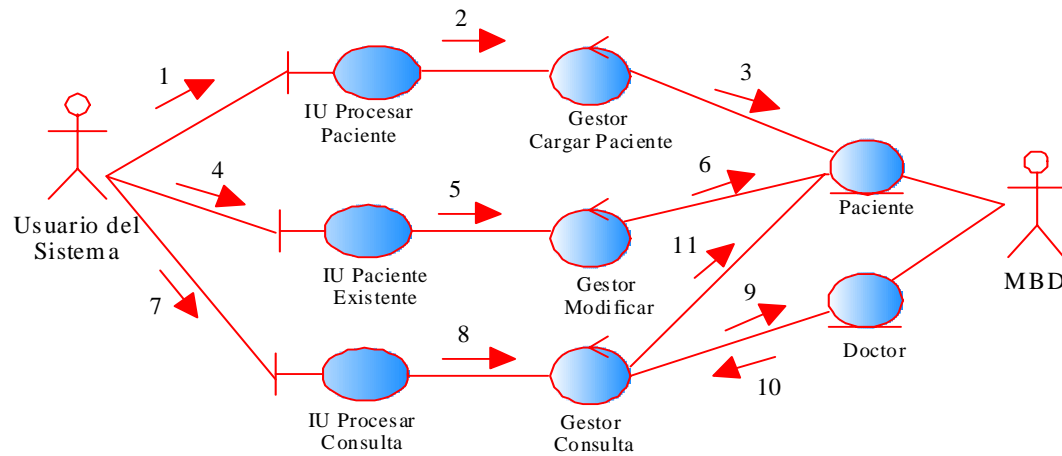
**Figura 4.10** Diagrama de clases de análisis de la realización del Caso de Uso  
Procesar Consulta Externa

**Fuente:** Elaboración Propia

#### 4.3.4 Diagrama de clases de colaboración de la realización del Caso de Uso Procesar Consulta externa

Mediante la interfaz consulta externa el usuario del sistema podrá ingresar y modificar los datos del paciente, también podrían asignar una consulta.

Los datos del paciente que deben ingresar son: nombre, apellido, edad, sexo, dirección de habitación. Esto se hace a través del objeto 1. En caso de modificar se pedirá el código del paciente, los datos del paciente que puede modificar es la edad y la dirección. Si desea una cita para una consulta se hace introduciendo el código del paciente y luego buscando al doctor disponible. Ver figura 4.11



#### Leyenda

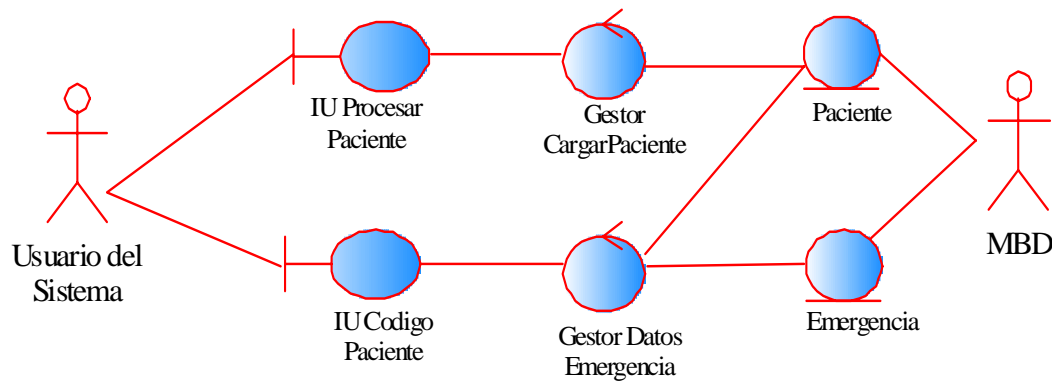
- |   |                                   |
|---|-----------------------------------|
| 1 Activa el proceso para ingresar los datos del paciente. | 8 Procesa la consulta. modificar. |
| 2 Procesa los datos.                                      | 9 Busca el doctor disponible.     |
| 3 Almacena los datos.                                     | 10 Regresa los datos del doctor   |
| 4 Activa el proceso para ingresar el código del paciente. | 11 Almacena la consulta.          |
| 5 Procesar los datos a modificar.                         |                                   |
| 6 Almacena los datos modificados.                         |                                   |
| 7 Activa el proceso para procesar la consulta.            |                                   |

**Figura 4.11** Diagrama de clases de colaboración de la realización del Caso de Uso Procesar Consulta Externa

**Fuente:** Elaboración Propia

#### 4.3.5 Diagrama de clases de análisis de la realización del caso de uso Procesar Emergencia

La figura 4.12 muestra el diagrama de clases de análisis de la realización del caso de uso Procesar Emergencia, el cual puede ser activado por el actor usuario del sistema mediante la interfaz de Emergencia, que permite que el usuario del sistemas introduzca los datos requeridos para realizar las respectivas estadísticas. La interfaz antes mencionada utiliza el gestor G.Emergencia para procesar los datos proveniente del area de emergencia.



**Figura 4.12** Diagrama de clases de análisis de la realización del Caso de Uso Procesar Emergencia

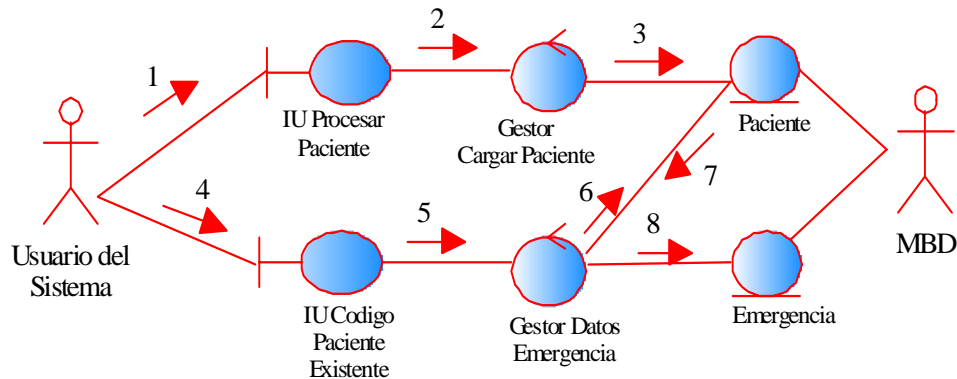
**Fuente:** Elaboración Propia

#### 4.3.6 Diagrama de clases de colaboración de la realización del Caso de Uso Procesar Emergencia.

El diagrama de la figura 4.13 muestra los objetos que conforman el diagrama de colaboración de la realización del caso de uso Procesar Emergencia.

A continuación se explica el flujo de activación para el usuario del sistema que desee ingresar datos provenientes de las áreas de emergencia mediante la activación de los objetos IU Emergencia, para realizar la solicitud de ingreso de datos de paciente que son asignados por el usuario del sistema, mediante el mensaje 1 activa el proceso para ingresar los datos del paciente, para llevar a cabo dicho propósito es de suma importancia contar con la información que suministra por el paciente, 2 se encarga de procesar dichos datos, una vez procesados los datos el mensaje 3 se encarga de almacenarlos. Para el mensaje 4 se activa el proceso mediante el código del paciente, luego se procesan los nuevos datos mediante el mensaje 5, luego se procede a buscar los datos del paciente a través del mensaje 6, una vez obtenidos los datos mediante el

mensaje 7, se procede a almacenar lo nuevos datos en la base de datos mediante el mensaje 8.



#### Leyenda

- 1 Activa el proceso para ingresar los datos del paciente.
- 2 Procesa los datos.
- 3 Almacena los datos.
- 4 Activa el proceso para ingresar el código del paciente.
- 5 Procesar los nuevos datos.
- 6 Busca datos del paciente.
- 7 Regresar datos del paciente .
- 8 Almacenar datos del paciente.

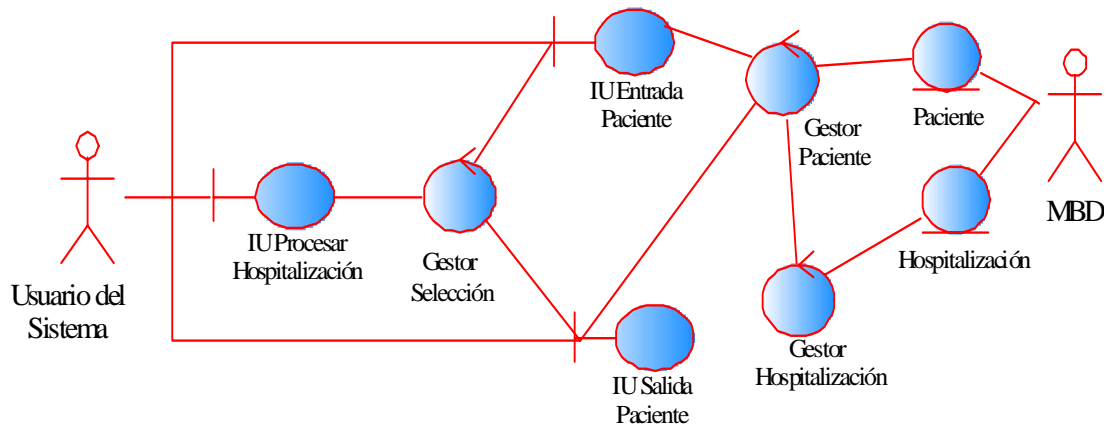
**Figura 4.13** Diagrama de clases de Colaboración de la realización del Caso de Uso Procesar Emergencia

**Fuente:** Elaboración Propia

#### 4.3.7 Diagrama de clases de análisis de la realización del caso de uso Procesar Hospitalización

La figura 4.14 muestra el diagrama de clases de análisis de la realización del caso de uso Procesar Hospitalización, el cual puede ser activado por el actor usuario del sistema mediante la interfaz de IU Hospitalización, que permite que el usuario del sistemas introduzca los datos requeridos.





**Figura 4.14** Diagrama de clases de análisis de la realización del Caso de Uso  
Procesar Hospitalización

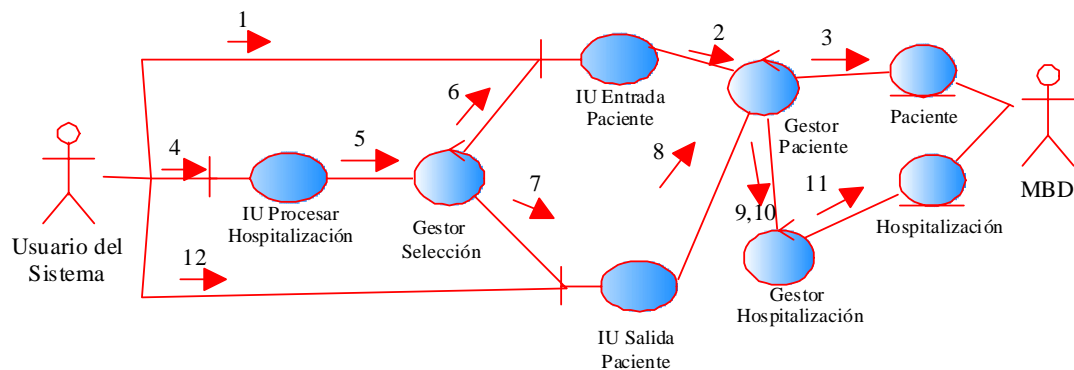
**Fuente:** Elaboración Propia

#### 4.3.8 Diagrama de clases de colaboración de la realización del Caso de Uso Procesar Hospitalización

El diagrama de la figura 4.15 muestra los objetos que conforman el diagrama de colaboración de la realización del caso de uso Procesar Hospitalización, los cuales fueron descritos en el análisis del diagrama anterior. El cual está formado por objetos relacionados entre sí, donde cada objeto tiene una función específica dentro del diagrama, como fue descrito.

A continuación se explica el flujo de activación para el usuario del sistema que desee ingresar datos provenientes del áreas Hospitalización mediante la activación del objeto IU Procesar Hospitalización, para realizar la solicitud de ingreso de datos que son asignado por el usuario del sistema, mediante el mensaje 1, por medio del mensaje 2 se hace una selección para ver hacia donde de los que se va a realizar si es entrada o salida del paciente, en el mensaje 3 se hace una activación de entrada del paciente en el área de hospitalización, en el mensaje 4 se gestionan los datos

requeridos, el mensaje 5 se encarga de buscar los datos, el mensaje 6 se encarga de activar el proceso de salida del paciente, el mensaje 7 gestiona los datos del paciente, el mensaje 8 lleva los datos entrada requerido en hospitalización, el mensaje 9 lleva los datos salida requerido en hospitalización y por último el mensaje 10 se encarga de almacenar los datos .



#### Leyenda

- 1,6 Activa el proceso para ingresar la entrada del paciente.
- 2 Gestiona datos del paciente.
- 3 Busca los datos.
- 4 Activa el proceso para ingresar los datos del paciente.
- 5 Selecciona el proceso a realizar .
- 7,12 Activa el proceso de salida del paciente.
- 8 Gestiona los datos del paciente .
- 9 Lleva los datos de la entrada.
- 10 Lleva los datos de la salida
- 11 Almacena los datos

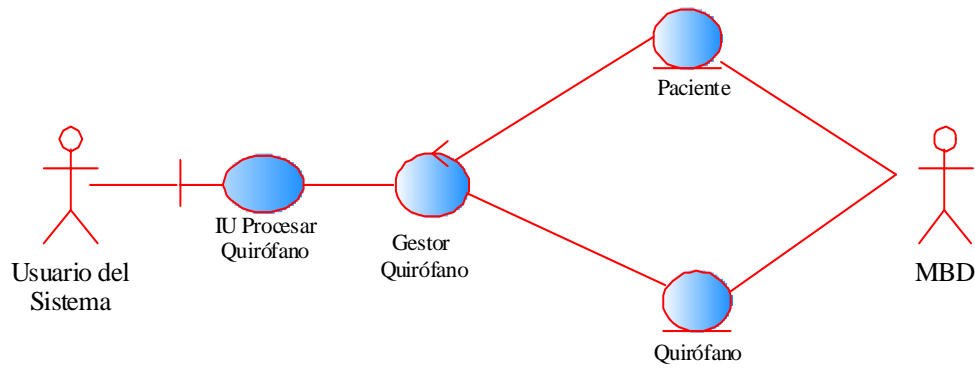
**Figura 4.15** Diagrama de clases de Colaboración de la realización del Caso de Uso Procesar Hospitalización

Fuente: Elaboración Propia

#### 4.3.9 Diagrama de clases de colaboración de la realización del caso de uso Procesar Quirófano.

La figura 4.16 muestra el diagrama de clases de análisis de la realización del caso de uso Procesar Quirófano, el cual puede ser activado por el actor usuario del sistema

mediante la interfaz de IU Quirófano, que permite que el usuario del sistemas introduzca los datos requeridos.



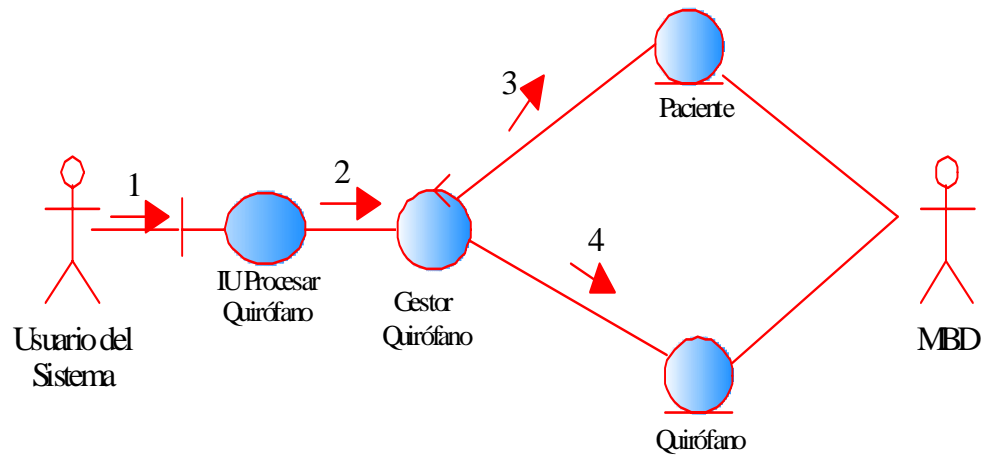
**Figura 4.16** Diagrama de clases de análisis de la realización del Caso de Uso Procesar Quirófano

**Fuente:** Elaboración Propia

#### 4.3.10 Diagrama de clases de colaboración de la realización del caso de uso Procesar Quirófano

La figura 4.17 muestra el diagrama de clases de colaboración de la realización del caso de uso Procesar Quirófano, el cual puede ser activado por el actor usuario del sistema mediante la interfaz de IU Quirófano, que permite que el usuario del sistemas introduzca los datos requeridos.

A continuación se explica el flujo de activación para el usuario del sistema que desee ingresar datos provenientes del áreas Quirófano mediante la activación del objeto: IU Quirófano, para realizar la solicitud de ingreso de datos que son asignado por el usuario del sistema, mediante el mensaje 1, luego por medio del mensaje 2 gestiona los datos de quirófano, en el mensaje 3 se buscan los datos del paciente para luego así poderlo almacenar con los nuevos datos del paciente mediante el mensaje 4.



### Leyenda

- 1 Activa el proceso para ingresar los datos del paciente.
- 2 Gestiona los datos de quirófano
- 3 Busca los datos del paciente
- 4 Almacena los datos.

**Figura 4.17** Diagrama de clases de Colaboración de la realización del Caso de Uso Procesar Quirófano

**Fuente:** Elaboración Propia

## 4.4 Diagrama de Paquete

El diagrama de figura 4.18 muestra los paquetes que encapsulan los diferentes casos de usos que fueron definidos al realizar el análisis del sistema SIREM, esto con la finalidad de esbozar dichos casos de usos en piezas más pequeñas que puedan ser fáciles de manejar.

De esta forma, resulta más fácil la tarea de diseñar una arquitectura que responda a los requerimientos del sistema, ya que se basa en los casos de usos.

Para llevar a cabo la identificación de estos paquetes de análisis asignamos un cierto número de casos de usos a un paquete concreto y luego realizamos la funcionalidad de dicho paquete, esto nos ayuda a asignar casos de uso a paquetes.

En la figura 4.18 se representa gráficamente las funcionalidades del sistema encapsulados en paquetes de análisis, según los requerimientos determinados que estos poseen. En la figura podemos observar tres paquetes que representan los casos de uso del sistema como son: Datos, áreas y reportes. Estos paquetes se asocian a sus funcionalidades mediante *trace* que parten de un paquete a otro paquete. En el diagrama podemos apreciar, por ejemplo como el paquete de análisis Datos se relaciona con todos los demás paquetes involucrados en el diagrama, debido a que estos soportan los requerimientos funcionales entre un paquete y otro, es decir, el paciente puede solicitar el ingreso por las diferentes áreas, lo cual generarían los reportes estadísticos, es por esto la relación del paquete datos con los demás paquetes del diagrama.

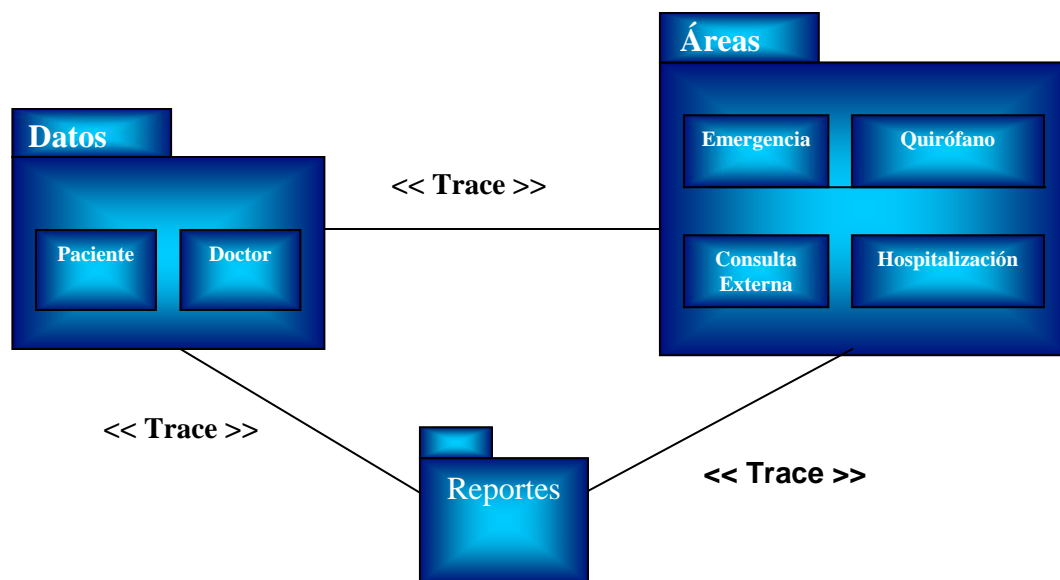
El paquete de análisis Reportes, agrupa las funcionalidades de: Reporte estadísticos de las áreas en estudio, y la información proveniente tanto del paciente como del doctor, de los cuales dan soporte al actor usuario del sistema. Otro ejemplo, es el paquete Áreas, que es donde se realizan toda la información de las estadísticas que son requeridas para realizar sus respectivos reportes.

#### **4.5 Conclusión de la fase de inicio**

Los objetivos principales de la fase de inicio son: establecer un análisis del sistema propuesto, describir el contexto del sistema, capturar los requerimientos funcionales, identificar los riesgos críticos que pondrían en peligro el desarrollo del proyecto y proponer una arquitectura candidata factible. Dichos objetivos se cumplieron de una

forma satisfactoria, ya que se obtuvo una primera versión del modelo del negocio que describe el contexto del sistema, una lista inicial de riesgos y un esbozo de los modelos que representan una primera versión del modelo de casos de uso y el modelo de análisis, los cuales describen una arquitectura candidata factible. En esta fase se obtuvo una buena comprensión del proyecto y la factibilidad de culminarlo.

Los resultados alcanzados en esta fase se refinarán en la fase de elaboración.



**Figura 4.18** Diagrama de paquete de análisis a partir de los casos de uso del sistema SIREM

**Fuente:** Elaboración Propia

#### 4.6 Fase de Elaboración

En esta fase se desarrolla la línea base para la arquitectura que cubra la funcionalidad del sistema y le sirva de guía a lo largo de su vida futura. Es de suma importancia en esta fase la captura y el análisis de los requisitos que no fueron abarcados en la fase

anterior para lograr un estudio más completo del Sistema SIREM. Así como también se elabora el modelo de diseño a través de los diagramas de clases y sus interfaces, el diseño de la Interfaz de Usuario y el diseño de las bases de datos.

La figura 4.19 ilustra los cinco flujos de trabajo que se realizan en esta fase, los cuales se distribuyen de la siguiente manera: 15% en la captura de los requisitos, 20% en el análisis y un 40% en el diseño del sistema.

#### **4.6.1 Requisitos**

En este flujo de trabajo se hizo una nueva evaluación del caso de uso principal del sistema SIREM, con el fin de analizar los nuevos casos de Uso que aparecen en esta fase, esto a razón de que se profundizó e indagó un poco más en el análisis del sistema, lo cual dio como resultado nuevas funcionalidades útiles para el sistema SIREM. Es por ello que ahora en la fase de elaboración se analizará y se detallará el caso de uso Generar Reportes.

#### **4.6.2 Caso de uso de la realización del Caso de Uso procesar reportes**

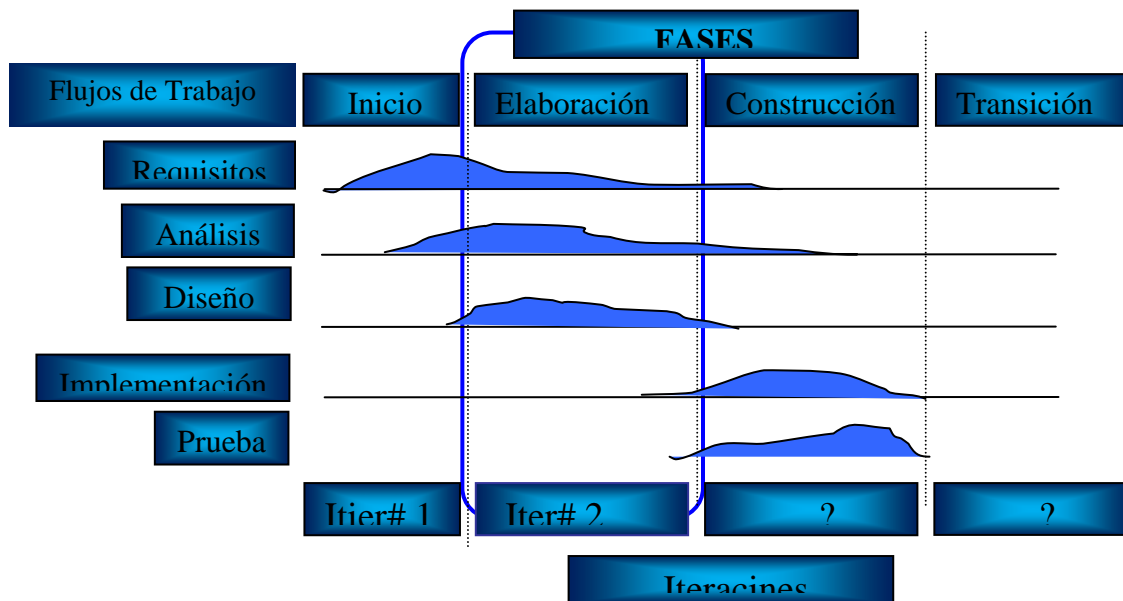
**Descripción:** Permite mostrar todos los reportes necesario del sistema. Ver figura 4.20

**Precondición:** Acceder al Sistema.

##### **Flujo principal**

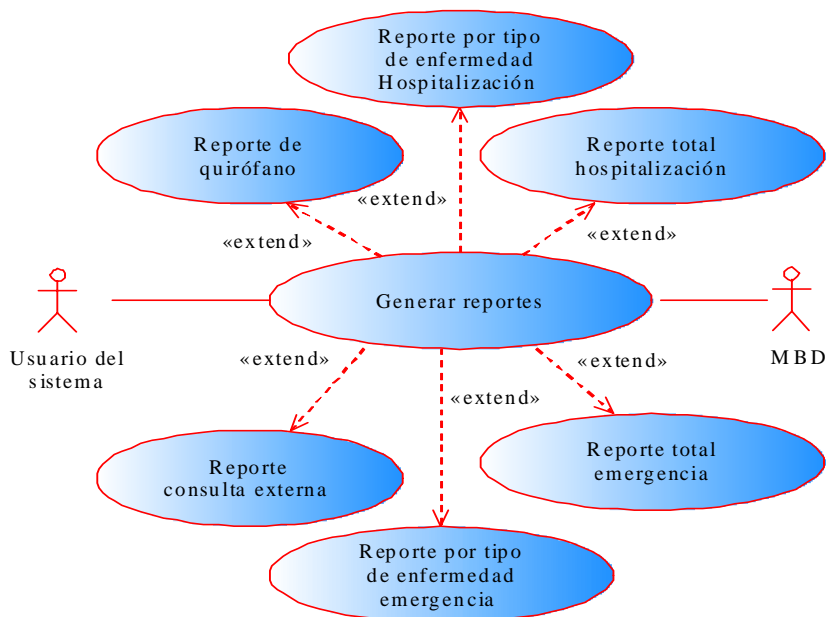
1. El usuario ingresa al modulo de reporte.
2. Ingresan el mes y año para poder imprimir los reportes.
3. Fin del caso de uso.

**Post condición:** Imprimir reportes.



**Figura 4.19** Flujo de trabajo de la fase de Elaboración del Sistema SIREM

**Fuente:** Elaboración Propia



**Figura 4.20** Diagrama de casos de uso de procesar reportes

**Fuente:** Elaboración propia



## **4.7 Análisis**

En esta fase de elaboración se hará un análisis de aquellos casos de uso que no fueron analizados en la fase anterior por no contar con la información necesaria para realizar dicho trabajo.

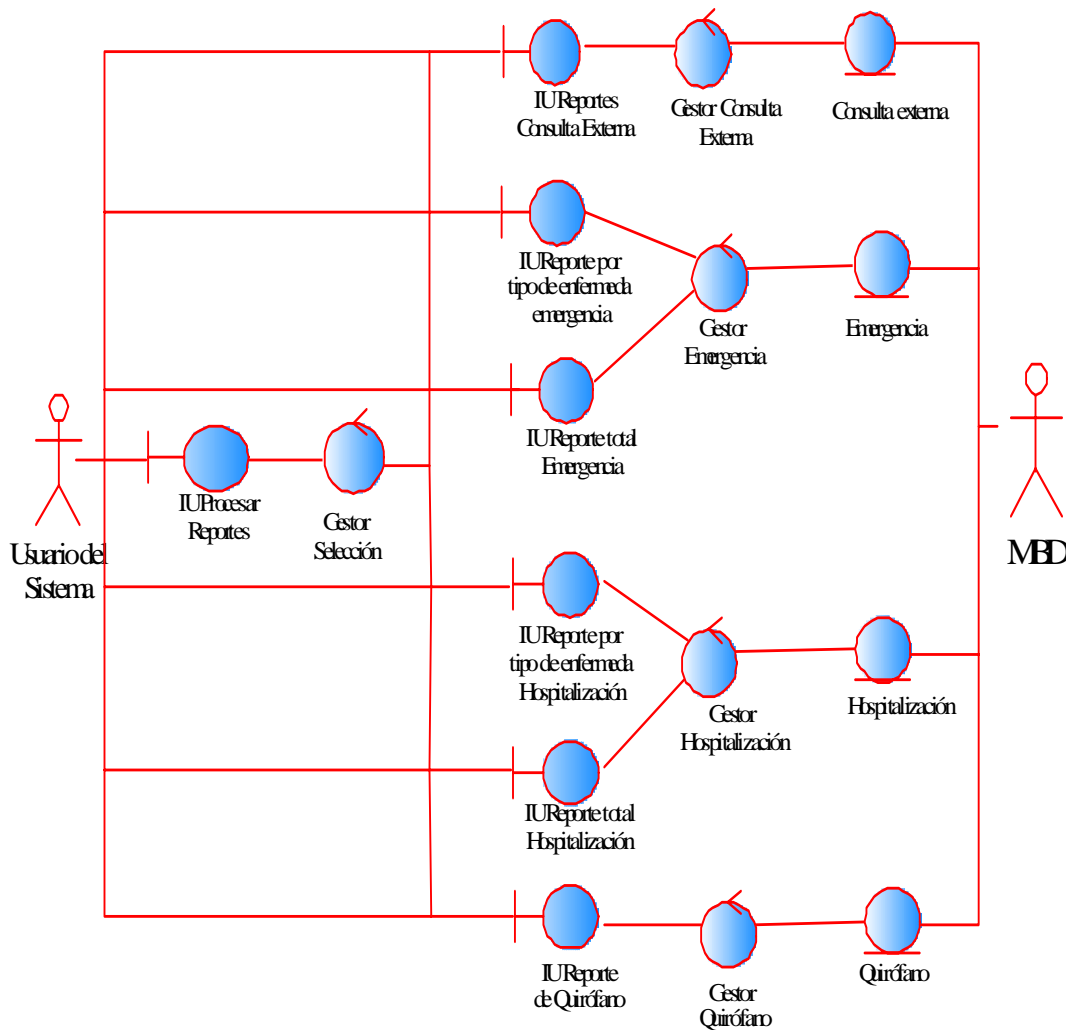
El flujo de trabajo de análisis en esta fase se representará por medio de los diagramas de clase y de colaboración de los casos de uso descritos en la captura de los requisitos.

### **4.7.1 Diagrama de clases de análisis de la realización del caso de uso Procesar Reportes**

La figura 4.21 muestra el diagrama de clases de análisis de la realización del caso de uso Procesar Reportes, el cual puede ser activado por el actor usuario del sistema mediante la interfaz de IU Reportes, que permite que el usuario del sistema imprima todo los reportes de la diferentes área en estudio.

### **4.7.2 Diagrama de colaboración de la realización del caso de uso generar reportes**

El diagrama de la figura 4.22 muestra los elementos que conforman el diagrama de Colaboración del Caso de Uso Generar Reporte. El cual se activa mediante el mensaje 1, Luego por medio del mensaje 2 se hace una selección dependiendo del tipo de reporte requerido por el usuario. Dicho reportes se obtienen introduciendo el mes y el año, ya que esto es requerido mensualmente.



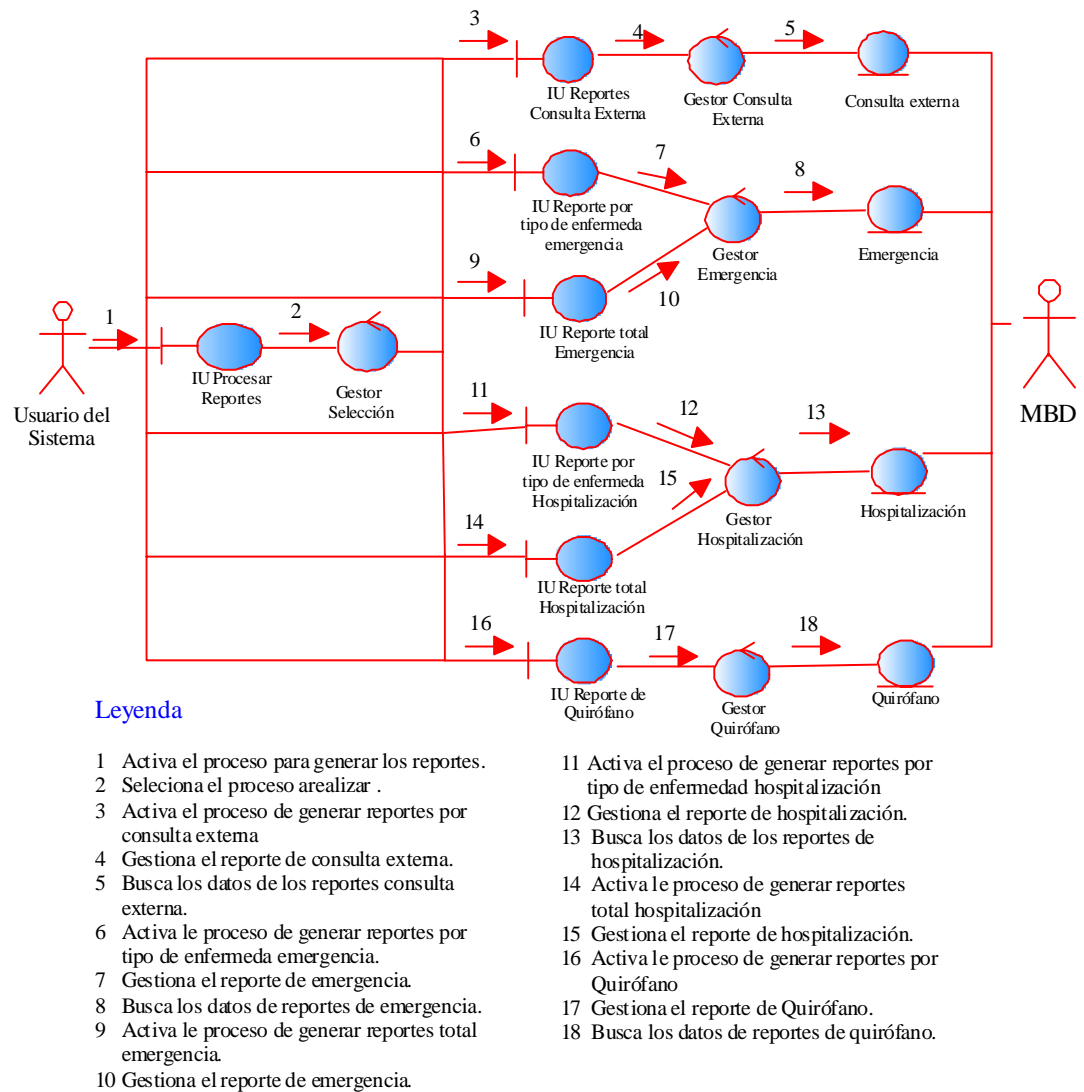
**Figura 4.21** Diagrama de clases de análisis de la realización del Caso de Uso  
Procesar Reportes

**Fuente:** Elaboración Propia

#### 4.8 Diseño

En el diseño modelamos el sistema y encontramos su arquitectura para que soporte todos los requisitos. Este flujo de trabajo tiene como entrada principal el resultado del análisis, que es el Modelo de Análisis, el cual proporciona una comprensión detallada

de los requisitos y una estructura del sistema que debemos conservar lo más fiel posible. El resultado de este flujo es el Modelo de Diseño, que es un modelo de objeto el cual describe la realización física de los casos de uso centrándose en como los requisitos tienen impacto en el sistema.



**Figura 4.22** Diagrama de colaboración de la realización del caso de Uso Generar Reportes Estadísticos

**Fuente:** Elaboración Propia

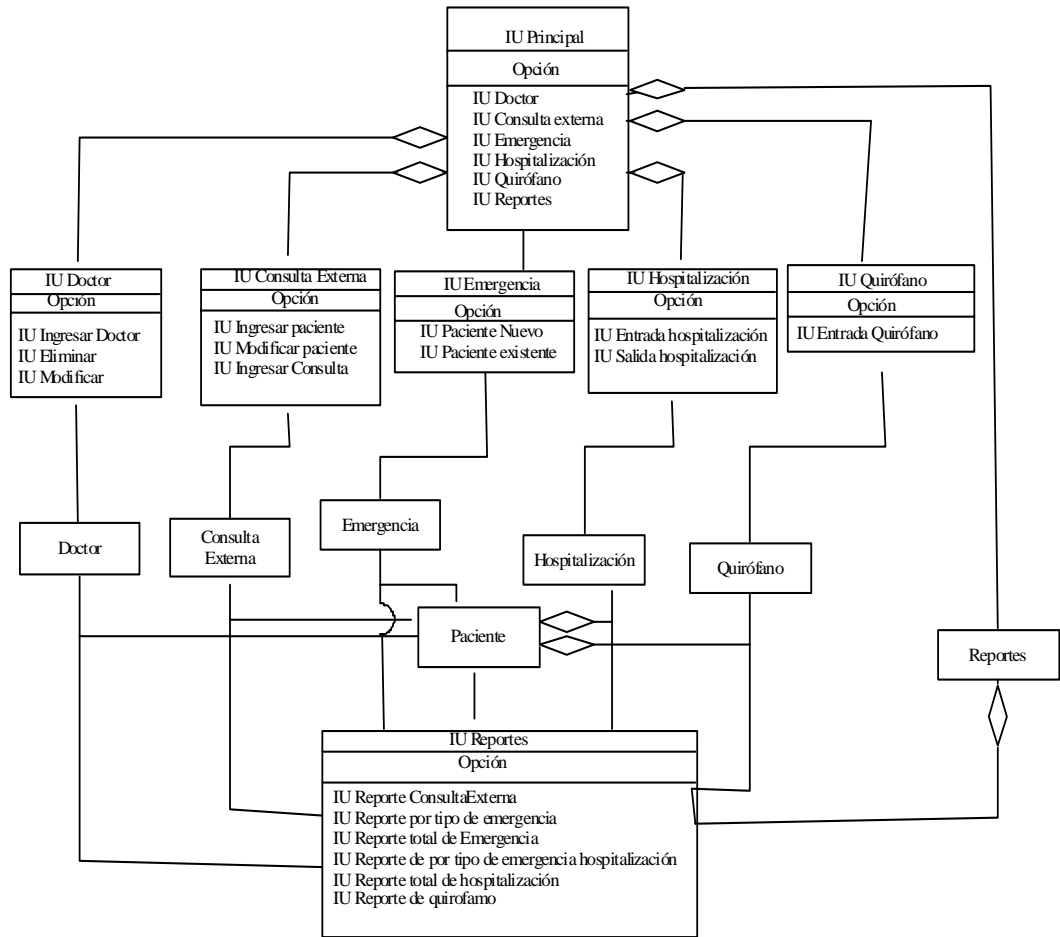
En esta fase se diseña desde el punto de vista de la arquitectura, es decir, diseñaremos las clases y subsistemas que sean arquitectónicamente significativos. Estos subsistemas de diseño y clases de diseño representan abstracciones del sistema y componentes de la implementación del sistema.

#### **4.8.1 Diagrama de Clases de Diseño**

En el Modelo de Diseño las clases de diseño y los subsistemas pueden participar en varias realizaciones de casos de uso. Por lo tanto es importante utilizar los diagramas de clases, ya que ellos nos muestran sus clases participantes, subsistemas y sus relaciones, y de esta forma se puede llevar la secuencia de los elementos participantes en una realización del caso de uso.

A continuación se presenta el análisis de la figura 4.23 el cual representa el Diagrama General de la Clases de Diseño del sistema SIREM. En el diagrama La Clase IU.Principal despliega el menú principal del sistema, y mantiene una relación de agregación (de todo y parte) con las interfaces IU.Ayuda, IU.Configuración, IU.Paciente, IU.Emergencia, I.Consulta Externa, IU.Hospitalización, IU.Quirofano, IU.Reportes, es decir, La interfaz IU.Principal es el todo y las otras interfaces son las partes, pero esta relación no es de total dependencia.

Continuando con el análisis del diagrama, observamos la clase Paciente la cual es agregada de la clase IU.Paciente, IU.Emergencia y IU.Consulta Externa, ya que los pacientes pueden ingresar por estas tres diferentes clases. De igual forma en él la clase Paciente tiene una relación de agregación con las clases IU.Hospitalización y IU.Quirofano, esta relación es de composición, es decir, hay una fuerte dependencia, ya que la clase IU.Hospitalización y IU.Quirofano se crea única y exclusivamente si existe un paciente.



**Figura 4.23** Diagrama general de la Clase de Diseño

**Fuente:** Elaboración propia

#### 4.8.2 Diagrama de Secuencia de la realización del Caso de Uso Procesar Paciente del Sistema SIREM

La figura 4.24 muestra el diagrama de secuencia con los objetos de diseño que se llevan a cabo la realización del caso de uso principal del sistema SIREM. La descripción del flujo de diseño para Procesar las estadísticas de los pacientes ingresados al hospital es la siguiente.

El usuario utiliza el sistema mediante el objeto: **IU.Paciente** y el objeto: **Paciente** para ingresar los datos del paciente. El paciente utiliza el objeto: **IU.Consulta Externa** para introducir sus datos y el objeto: **Consulta Externa** para procesar los datos, también utiliza el objeto: **IU.Emergencia** para ingresar sus datos y el objeto: **Emergencia** para procesar dichos datos, una vez procesado los datos del paciente por el objeto: **Consulta Externa**, el paciente puede ser remitido a hospitalización a través del objeto: **IU.Hospitalización**, también el paciente puede provenir de emergencia a hospitalización mediante el objeto: **Emergencia**. Una vez que el paciente ingresa a hospitalización a través del objeto: **IU.Hospitalización**, el objeto: **Hospitalización** se encarga de procesar los datos del paciente. El paciente también puede ser remitido de emergencia a quirófano a través del objeto: **Emergencia**, además el paciente puede provenir de hospitalización a quirófano por medio del objeto: **Hospitalización**, una vez que el paciente ingresa a quirófano a través del objeto: **IUQuirofono**, los datos son procesado por medio del objeto: **Quirófano**, el paciente puede ser enviado de quirófano a hospitalización a través del objeto: **Quirófano**. El usuario utiliza el sistema mediante el objeto: **IU.Paciente** y el objeto: **Paciente** para ingresar los datos del paciente. El paciente utiliza el objeto: **IU.Consulta Externa** para introducir sus datos y el objeto: **Consulta Externa** para procesar los datos, también utiliza el objeto: **IU.Emergencia** para ingresar sus datos y el objeto: **Emergencia** para procesar dichos datos, una vez procesado los datos del paciente por el objeto: **Consulta Externa**, el paciente puede ser remitido a hospitalización a través del objeto: **IU.Hospitalización**, también el paciente puede provenir de emergencia a hospitalización mediante el objeto: **Emergencia**. Una vez que el paciente ingresa a hospitalización a través del objeto: **IU.Hospitalización**, el objeto: **Hospitalización** se encarga de procesar los datos del paciente. El paciente también puede ser remitido de emergencia a quirófano a través del objeto: **Emergencia**, además el paciente puede provenir de hospitalización a quirófano por medio del objeto: **Hospitalización**, una vez que el paciente ingresa a quirófano a través del objeto: **IUQuirofono**, los datos son procesado por medio del objeto:

**Quirófano**, el paciente puede ser enviado de quirófano a hospitalización a través del objeto: **Quirófano**.

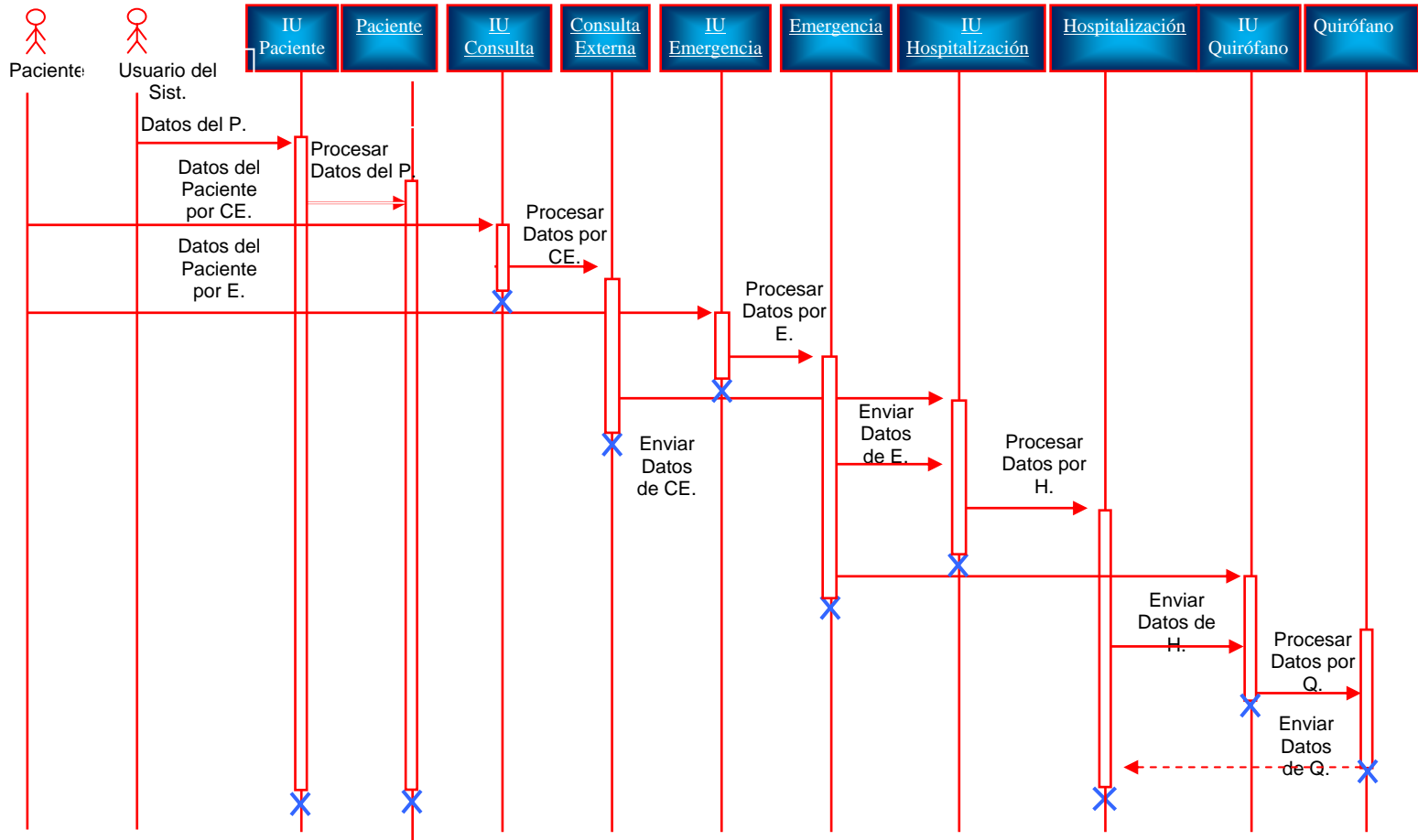
### **4.8.3 Diseño de la arquitectura**

En este flujo de trabajo se diseñan los aspectos arquitectónicamente significativos del sistema, y se hará mediante la identificación de la arquitectura en capas.

Un sistema con una arquitectura en capas ubica a los subsistemas de aplicación individuales en lo más alto. Estos se construyen a partir de subsistemas en las capas más bajas, como son los marcos de trabajo y las bibliotecas de clases. En la figura 4.25 la capa específica de la aplicación es la parte de un sistema que es específica de la aplicación, utiliza la capa general de la aplicación y no es compartida por otras partes. La capa general de la aplicación contiene los subsistemas que no son específicos de una sola aplicación, sino que pueden ser reutilizados por muchas aplicaciones diferentes dentro del mismo dominio o negocio. El software del sistema y la capa intermedia constituyen el cimiento del sistema, ya que toda la funcionalidad descansa sobre software como sistemas operativos, sistemas de gestión de bases de datos, software de comunicaciones, etc.

### **4.8.4 Diseño de las Bases de Datos DB SIREM**

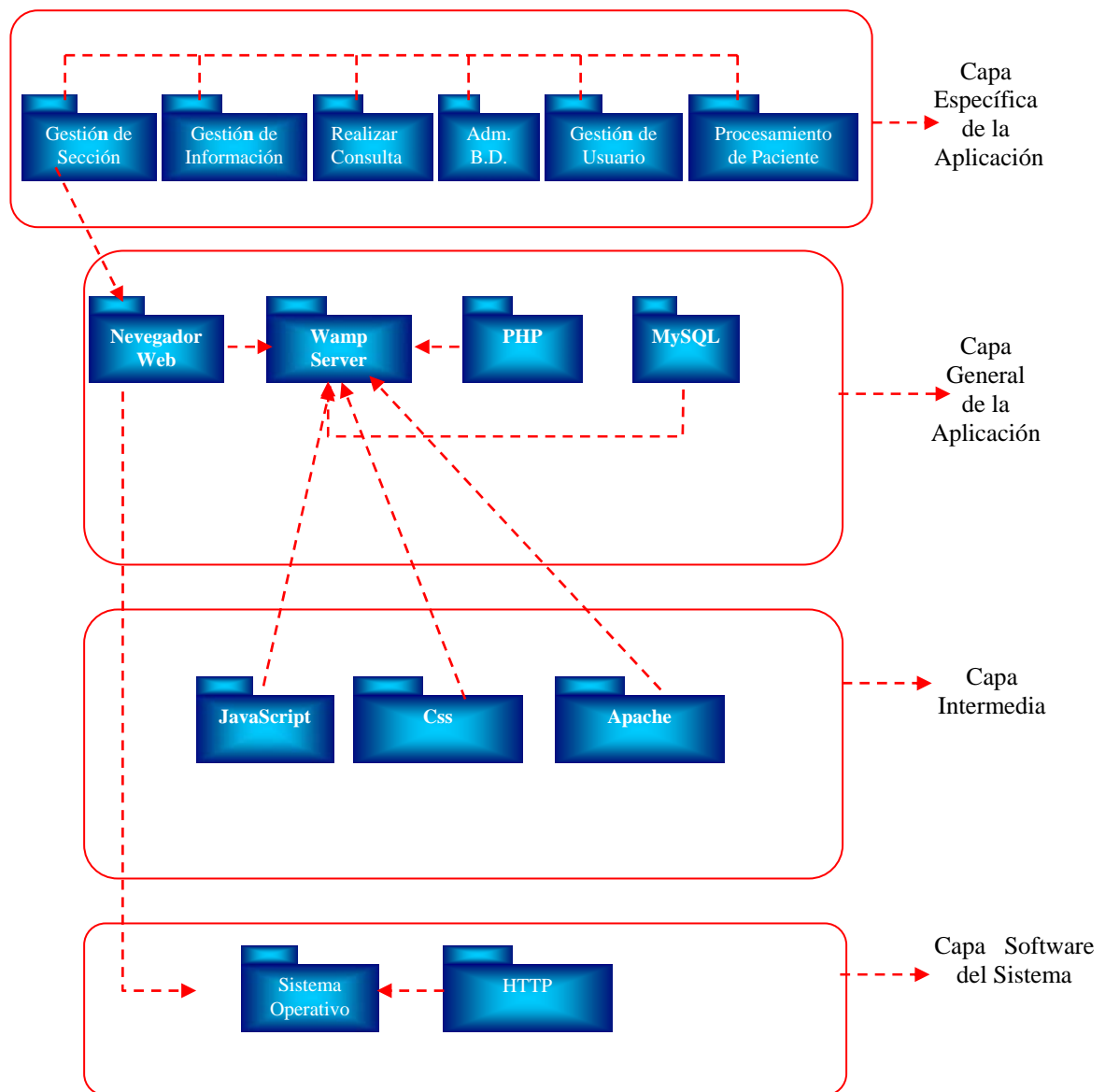
El diseño de la base de datos conforma una parte de suma importancia en el diseño del sistema, ya que por medio de esta es posible almacenar toda la información necesaria, de tal forma que los datos puedan ser manejados y almacenados óptimamente, ahorrando espacio en el disco y a su vez aumenta la velocidad de respuesta del software.



**Figura 4.24** Diagrama de secuencia de la realización del caso de uso Procesar Paciente del sistema SIREM

**Fuente:** Elaboración propia



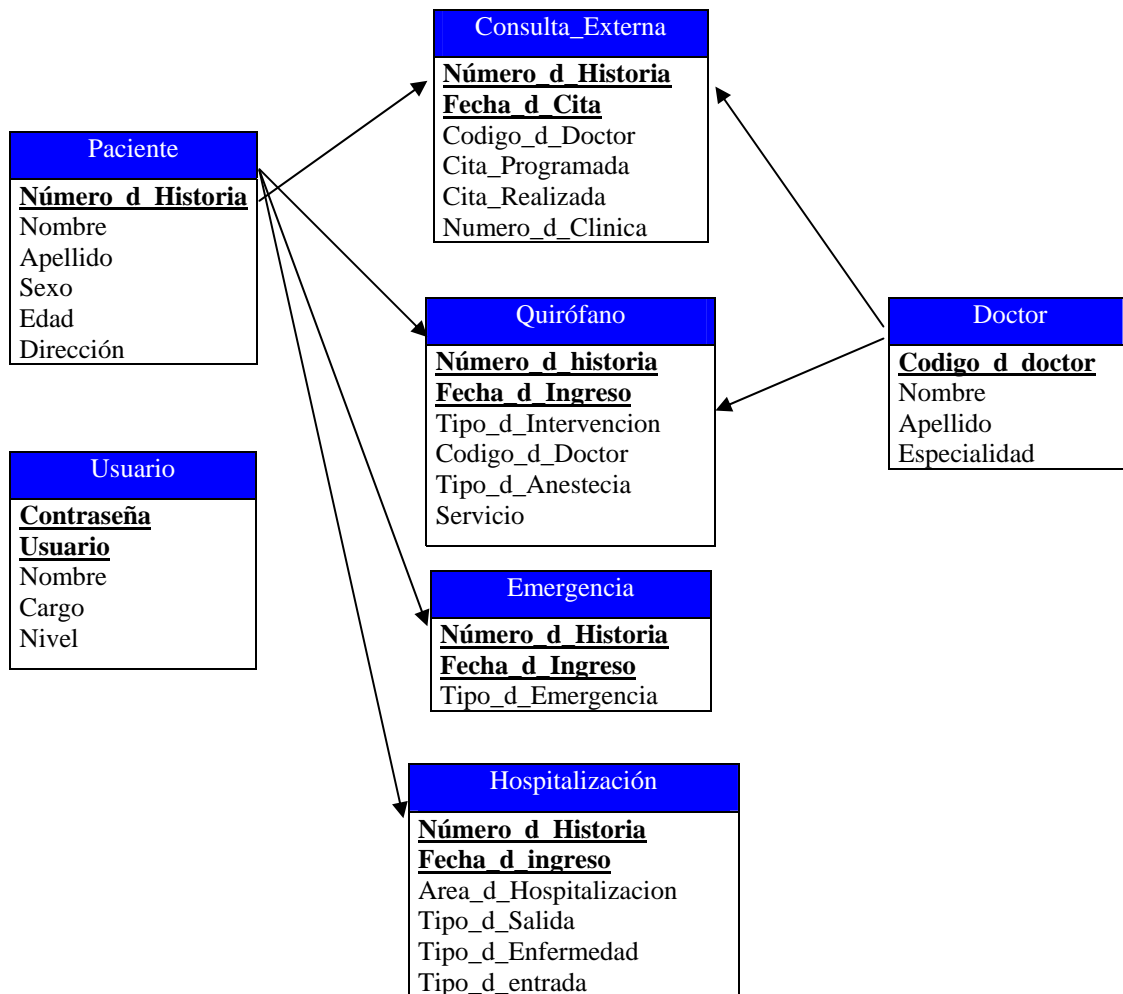


**Figura 25** Diseño de la arquitectura del sistema

**Fuente:** Elaboración Propia

Hemos utilizado un modelo de datos para el diseño de la base de datos del sistema SIREM que sigue las especificaciones del modelo relacional, como son la

representación de los datos y las relaciones entre estos datos mediante tablas, donde se garantiza la normalización de las mismas para evitar la redundancia de datos.



**Figura 4.26** Relación de Tablas de la Base de Datos

**Fuente:** Elaboración propia

#### 4.8.4.1 Tablas de la Base de Datos DB SIREM

Ahora presentaremos una serie de tablas que fueron diseñadas siguiendo los lineamientos del modelo relacional y las formas normales, las cuales constituyen la

base de datos BD SIREM, donde se almacenará la información más importante que posteriormente será utilizada para el funcionamiento de las diferentes áreas del hospital.

#### Tabla Acceso de Usuarios

Contiene el cargo y el tipo de acceso que tendrá el usuario al ingresar al sistema, estos datos se usan para lograr que el usuario entre solo a los módulos asignados. Se muestra en la figura 4.34.

**Tabla 4.4** Acceso de Usuarios

Nombre del Campo	Tipo	Longitud	Descripción
<u>Usuario</u>	carácter	10	Identificación del usuario
<u>Contraseña</u>	carácter	10	Clave secreta de acceso del usuario
Nombre	carácter	20	Nombre Completo del usuario
Cargo	Numérico	1	Indica el cargo del usuario
Nivel	Numérico	1	Indica el privilegio de acceso del usuario

**Fuente:** Elaboración propia

Tabla quirófano

Esta tabla es la que permite almacenar todas las informaciones referentes a quirófano.

**Tabla 4.5** Quirófano

Nombre del Campo	Tipo	Longitud	Descripción
<b><u>Numero d Historia</u></b>	Numérico	10	Identificación al paciente
<b><u>Fecha d Ingreso</u></b>	fecha	10	Indica la fecha de ingreso
Tipo_d_ Intervención	carácter	20	Indica el tipo de intervención que se realiza
Codigo_d_Doctor	Numérico	10	Identifica al doctor
Tipo_d_Anestesia	carácter	20	Indica el tipo de anestesia aplicada
Servicio	Numérico	1	Indica si la intervención se realizo o no

**Fuente:** Elaboración propia

Tabla emergencia

Esta tabla es la que permite almacenar todas las informaciones referentes a emergencia.

**Tabla 4.6** Emergencia

Nombre del Campo	Tipo	Longitud	Descripción
<b><u>Numero d Historia</u></b>	Numérico	10	Identificación al paciente
<b><u>Fecha d Ingreso</u></b>	fecha	10	Indica la fecha de ingreso
Tipo_d_Emergencia	carácter	20	Indica el tipo de emergencia

**Fuente:** Elaboración propia

Tabla consulta externa

Esta tabla es la que permite almacenar todas las informaciones referentes a consulta externa

**Tabla 4.7** Consulta externa

Nombre del Campo	Tipo	Longitud	Descripción
<b><u>Numero d Historia</u></b>	Numérico	10	Identificación al paciente
<b><u>Fecha d Cita</u></b>	fecha	10	Indica la fecha de ingreso
Codigo_d_Doctor	Numérico	10	Identifica al doctor
Cita_Programada	Numérico	1	Identifica la cantidad de cita programada durante el mes
Cita_Relizada	Numérico	1	Indica la cantidad de cita realizada
Numero de clinica	Numérico	1	

**Fuente:** Elaboración propia

Tabla hospitalización

Esta tabla es la que permite almacenar todas las información requiere el área de hospitalización

**Tabla 4.8** Hospitalización

Nombre del Campo	Tipo	Longitud	Descripción
<b><u>Numero d Historia</u></b>	Numérico	10	Identifica al Paciente
<b><u>Fecha d Ingreso</u></b>	carácter	10	Indica la fecha de ingreso
Area_d_Hospitalización	carácter	20	Identifica el área donde va hacer recluso
Tipo_d_Salida	carácter	20	Indica el tipo de salida
Tipo_d_Enfermedad	carácter	20	Indica el tipo de enfermedad
Tipo_d_Entrada	carácter	20	Indica la forma en que fue hospitalizado el paciente

**Fuente:** Elaboración propia

Tabla Paciente:

Esta tabla es la que permite almacenar toda la información del paciente.

Campos de la tabla

**Tabla 4.9** Paciente

Nombre del Campo	Tipo	Longitud	Descripción
<b><u>Numero d Historia</u></b>	Numérico	10	Identifica al Paciente
Nombre	carácter	10	Identifica el nombre del paciente
Apellido	carácter	20	Identifica el apellido del paciente
Sexo	carácter	1	Indica el tipo de sexo
Edad	Numérico	2	Indica la edad del paciente
Dirección	carácter	20	Identifica la dirección del paciente

**Fuente:** Elaboración propia

Tabla Doctor

Esta tabla es la que permite almacenar toda la información del doctor.

Campos de la tabla

**Tabla 4.10** Paciente

Nombre del Campo	Tipo	Longitud	Descripción
<b><u>Codigo d Doctor</u></b>	Numérico	10	Identifica al doctor
Nombre	carácter	20	Identifica el nombre del Doctor
Apellido	carácter	20	Identifica el apellido del doctor
Especialidad	carácter	20	Identifica la especialidad del doctor

**Fuente:** Elaboración propia

## 4.9 Implementación

En este flujo, se toma como base los resultados del diseño para implementar el sistema en términos de componentes, dónde se efectúa la primera versión del modelo de implementación, que describe la realización física de los elementos del modelo de diseño.

### 4.9.1 Identificación de Componentes de la Arquitectura

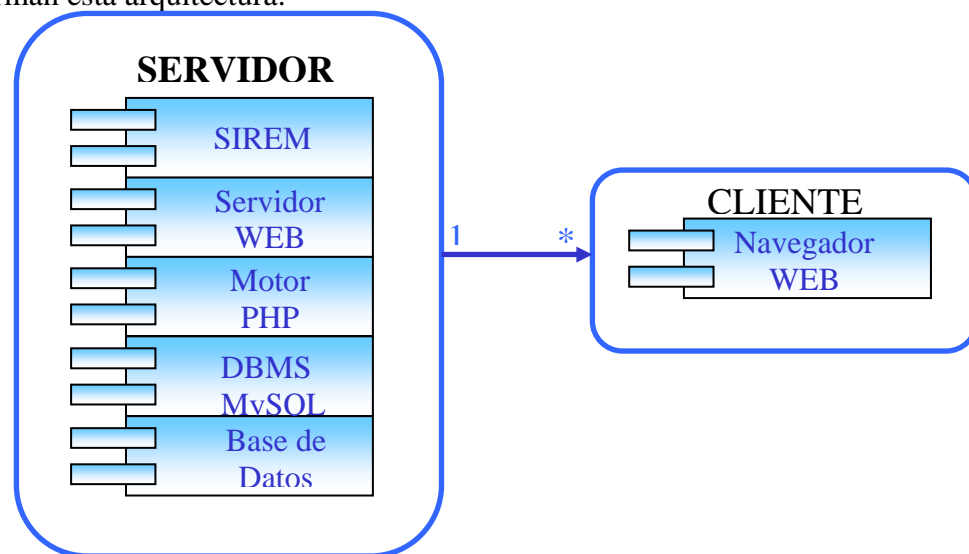
Un diagrama de despliegue muestra sobre cuál componente de hardware se instala (o despliega) cada componente de software. También muestra los enlaces de comunicación entre los componentes de hardware. Este diagrama expone las relaciones físicas de los distintos nodos que componen un sistema y el reparto de los componentes sobre estos nodos.

Para poder implementar la arquitectura de este sistema, se identificaron los componentes mostrados en la figura 4.27, en ella se muestra el servidor Web Apache

2.2.11 que permite la distribución de las páginas, el motor de PHP 5.3 para ejecutar los scripts que contienen la funcionalidad del sistema y el Sistema Manejador de Base de Datos (DBMS) MySQL5.1.36 que contiene la Base de Datos y provee la conexión entre ésta y la administración del manejo de los datos. Adicionalmente, en la misma figura, se observa el cliente provisto de navegador Web encargado de generar el documento HTML mediante el cual los usuarios podrán visualizar e interactuar con la aplicación. El Cliente y el Servidor se conectan a través de la Intranet mediante el protocolo de transmisión http.

#### 4.9.2 Implementación de la Arquitectura

En esta fase se implementó la arquitectura a través de la construcción de un prototipo de la página principal de Inicio de Sesión denominada Index, la cual conforma parte del Caso de Uso Gestionar Sesión, también se ejecutan los componentes que conforman esta arquitectura.



**Figura 4.27** Diagrama de Despliegue del Sistema.

**Fuente:** Elaboración Propia

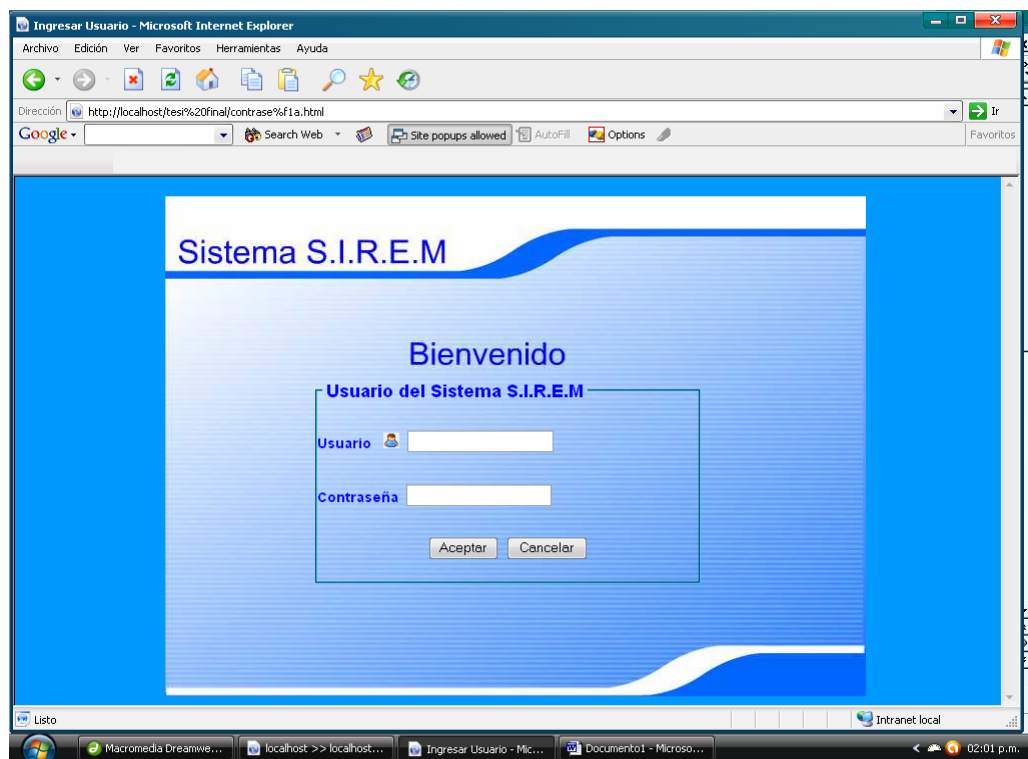
Esta página fue realizada utilizando lenguaje de scripts PHP, mostrando la interfaz de Inicio de Sesión y conteniendo las instrucciones necesarias para conectar y



consultar la Base de Datos. En la figura 4.28 se muestra la Pantalla de Inicio de Sesión.

### 4.9.3 Conclusión de la Fase de Elaboración

La Fase de Elaboración mostró la existencia de requisitos que no fueron abordados en la fase anterior y que fueron cubiertos con la creación de dos (2) casos de uso los cuales son: “Administrar Base de Datos” y “Gestionar Usuarios”, a los cuales se le aplicaron el flujo de trabajo de análisis igual que en la fase anterior.



**Figura 4.28** Pantalla de Inicio de Sesión

**Fuente:** Elaboración Propia.

Con la finalización de esta fase se obtienen todos los casos de usos que intervendrán en el sistema. Lo que permite describir de la arquitectura, que consistió:

en la identificación de los subsistemas, el diseño de clases, el diseño de las tablas pertenecientes a la base de datos y la realización del diagrama de despliegue del sistema, donde se muestra la distribución física del sistema en términos de funcionalidad, completando así, la línea base de la arquitectura.

#### **4.10 Fase de construcción**

El objetivo principal de esta fase es dejar listo el software SIREM en su versión operativa y con la calidad adecuada para su aplicación. En la fase anterior se establecieron los fundamentos de los elementos arquitectónicamente significativos de los modelos de diseño y despliegue. Estos fundamentos incluyen los subsistemas, las clases activas, así como la realización de los casos de uso significativos.

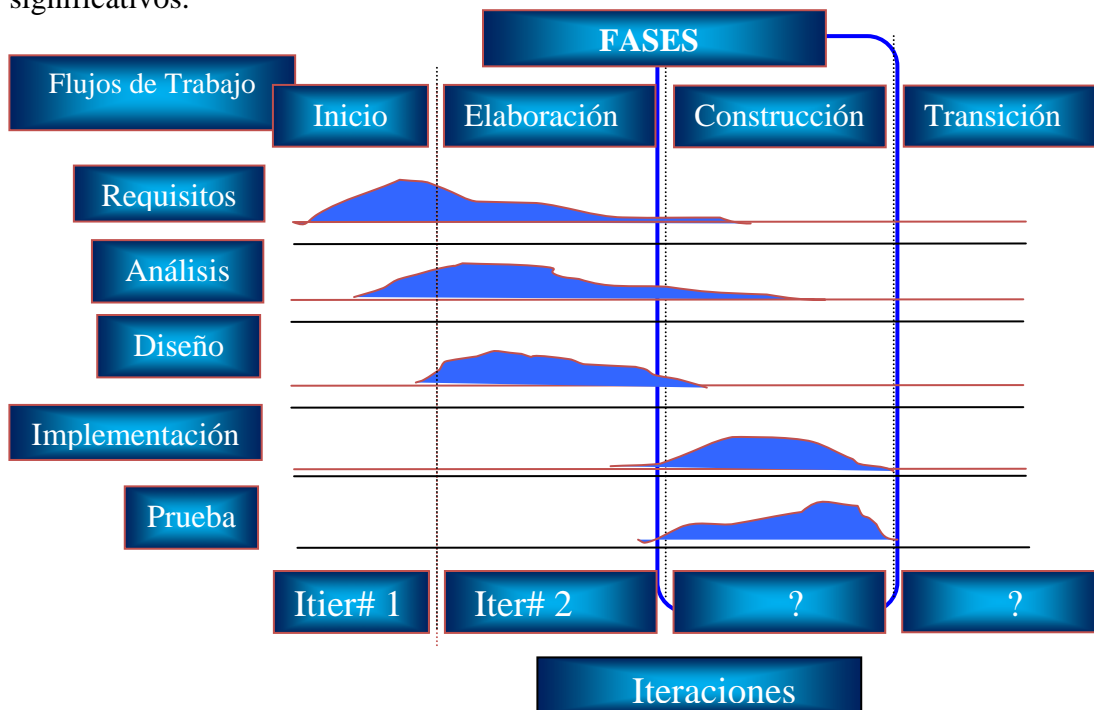
Para cumplir con los objetivos de la fase de Elaboración se recopilamos casi todos los requisitos pero no se detallaron completamente y esto es lo que vamos a realizar en la fase de Construcción, así como también se diseñaran las interfaces de usuarios del sistema SIREM. Además, se integraran los subsistemas y se probarán, de igual forma se integrara todo el sistema y se realizara la prueba de dicho subsistema.

En la fase de Construcción elaboraremos los materiales escritos de ayuda a los usuarios finales, como lo son los manuales de usuarios, diapositivas, etc.

La figura 4.29 ilustra una gráfica que muestra las cuatro fases y los cinco flujos de trabajos considerados en este sistema. En esta fase se abarca el 5% faltante de la captura de los requisitos, un 60% en el análisis del sistema y un 60% en el diseño del software, hasta el punto en el que el software está listo para ser utilizado por el usuario final.

##### **4.10.1 Requisitos**

Al culminar la fase anterior se habían identificados todos los requisitos y desarrollado el modelo de caso de uso definitivo del sistema SIREM (Figura 4.26) donde se muestran todos los diagramas de casos de uso y el detalle de los casos de uso más significativos.



**Figura 4.29** Flujo de trabajo de la fase de Elaboración del Sistema SIREM

**Fuente:** Elaboración Propia.

Para este flujo de trabajo el objetivo es diseñar las interfaces de usuarios, las cuales nos ayudan a comprender y especificar las interacciones entre actores y el sistema SIREM.

Para el diseño de las interfaces de usuario se comienza con los casos de uso y se identifica que se necesita de las interfaces de usuario para habilitar los casos de uso que los actores pueden acceder.

Después se desarrollan los prototipos que ilustran como los usuarios pueden utilizar el sistema para ejecutar los casos de uso.

El resultado final de este flujo de trabajo es un conjunto de esquemas de interfaces de usuarios que especifican la apariencia de dichas interfaces de cara a los actores más importantes y que satisfacen las necesidades de los usuarios.

#### **4.10.2 Diseño de la interfaz de Usuario**

El diseño de la interfaz de usuario permite al usuario llevar a cabo los casos de uso de manera eficiente. Para ello se construyen prototipos ejecutables de las configuraciones más importantes de elementos de interfaz de usuario.

La validación de las interfaces de usuarios a través de revisiones de prototipos y esquemas en los primeros momentos puede prevenir muchos errores que serán más caros de corregir después.

Uno de los aspectos más importantes de esta fase es el diseño de la interfaz de usuario, su importancia radica en los siguientes puntos.

- Permiten que el actor navegue de forma adecuada.
- Proporcionan una apariencia agradable y una forma consistente de trabajo con la interfaz de usuario, como por ejemplo, en cuanto a orden de tabulación y teclas rápidas.

La interfaz de usuario de SIREM está formada por un grupo de ventanas que permiten una opción específica del sistema, y de esta forma facilitar el uso del sistema.

Dichas interfaces se presentan como repuestas a eventos provocados por el usuario al pulsar el botón del Mouse o el teclado.

En las secciones siguientes se describen todas las interfaces de usuario adoptadas para el sistema SIREM, se hace una especificación detallada del diseño de las pantallas de la aplicación, mensajes de error, notificación de fallas del sistema, mensajes de diálogos, etc.

#### **4.11 Flujo de Trabajo: Implementación**

En esta etapa se describen las interfaces de Inicio de Sesión y Realizar Pedido, así como también el reporte en PDF del pedido. Para ello se seguirá el siguiente esquema.

- Breve explicación de la interfaz (en qué consiste).
- Identificación de los datos a manejar por el área. (datos que debe introducir el usuario durante su visita).
- Imagen referencial de la pantalla del sistema.
- Código Fuente.

#### **❖ Inicio de Sesión**

Esta Interfaz permitirá a los usuarios ingresar al sistema, al introducir el nombre y clave de usuario el sistema validará los datos y direccional a cada usuario a la interfaz que le corresponde .En la figura 4.30 se muestra la Pantalla de Inicio de Sesión.

#### **✓ CÓDIGO FUENTE**

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>Ingresar Usuario</title>
<LINK href="formato.css" type="text/css" rel="stylesheet">
<script language="javascript">
<!-- oculta el script para navegadores antiguos
function checkElement(el_obj)
{
    var result = true;
    if(el_obj.options==null) {
        if(el_obj.value=="") result = false;

    }
        else if(el_obj.selectedIndex==0)
            result = false;
        return result;
    }

<!--Muestra un mensaje por pantalla
function fixElement(element, message) {
alert(message);
element.focus();
}

<!-- función que verifica si los campos de texto no estén vacíos

function verifica(form) {
var passed = false;

    if (form.Usuario.value == "") {
        fixElement(form.Usuario, "Por favor, Introduzca el Nombre del
Usuario");
    }
    else if (form.Contraseña.value == "") {
        fixElement(form.Contraseña, "Por favor, Introduzca la Contraseña");
    }
    else
        return true;
return false;

```

```

}
</script>

</head>

<body class="b">

<p>&nbsp;</p><fieldset style="border-color:#33CCFF" >

    <h1>

<legend class="Estilo3"><span class="Estilo3">Usuario del Sistema
S.I.R.E.M</span>          </legend>

<form action="ingresar_usuario.php" method="post" onsubmit= "return
verifica(this);">

<label><span class="Estilo2" >Usuario&nbsp;&nbsp;&nbsp;&nbsp;</span>
    <input type="text" name="Usuario">
    </label>
<p> <label><span class="Estilo2" >Contraseña</span>

        <input type="password" name="Contraseña" />
    </label>
</p>
<center><INPUT type="submit" value="Aceptar" />
    <input name="" type="reset"

onclick="window.close()"value="Cancelar"/>
    </center>

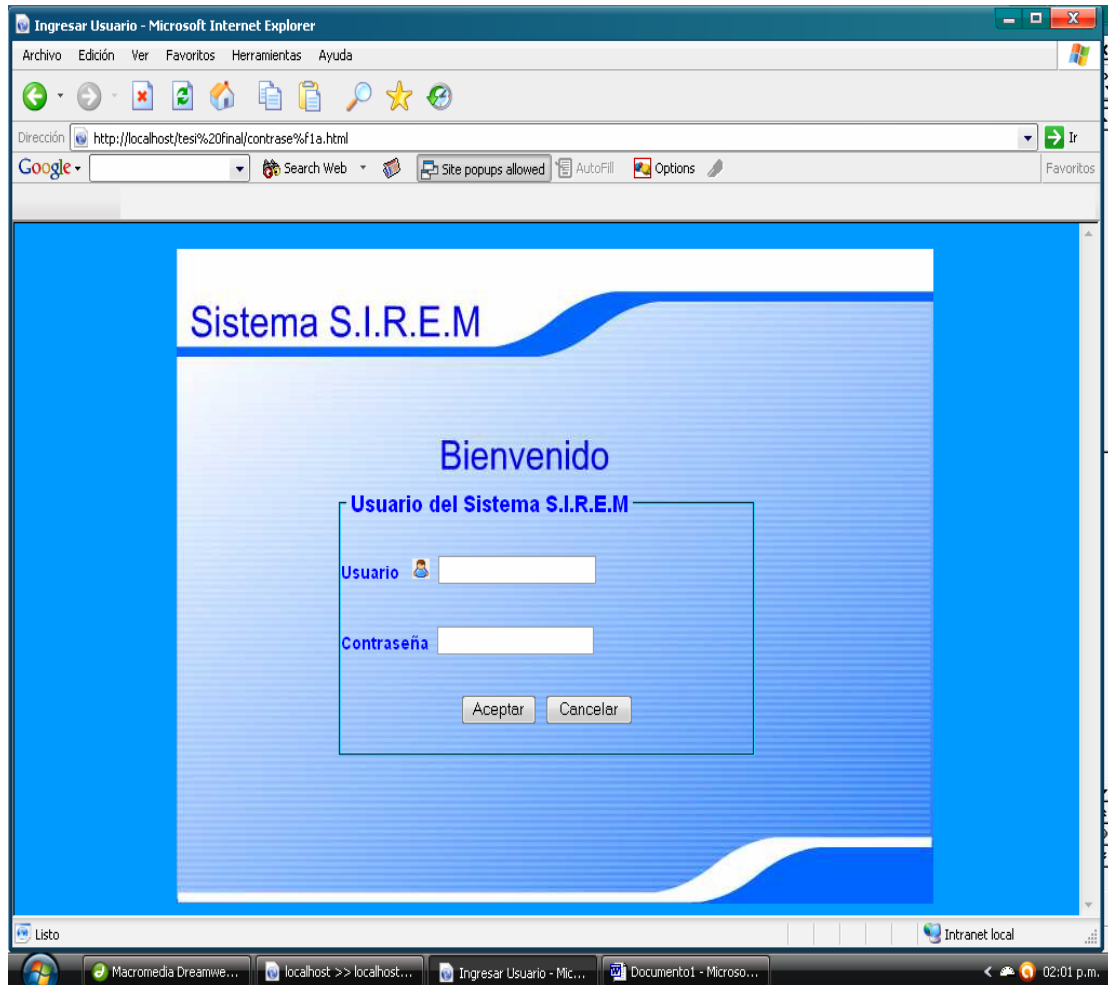
</form>

</fieldset>

</body>

</html>

```



**Figura 4.30** Pantalla de Inicio de Sesión

**Fuente:** Elaboración Propia.

#### ❖ Ingresar datos del paciente

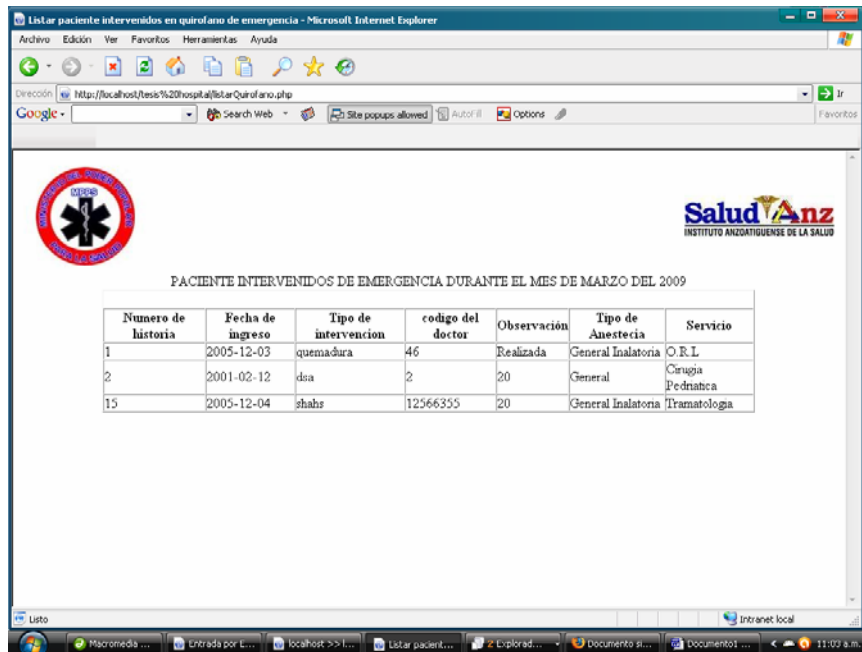
Esta interfaz le permitirá al actor usuario ingresar los datos del paciente diariamente de una manera más fácil y rápida, seleccionando: Nombre, Apellido, Edad, Sexo, Dirección, Fecha de Ingreso y número de Historia. . Esta interfaz se muestra en la figura 4.31







de intervención, Código del Doctor, Observaciones, Tipo de Anestesia, Servicio.  
Este reporte se muestra en la figura 4.32



Numero de historia	Fecha de ingreso	Tipo de intervencion	codigo del doctor	Observación	Tipo de Anestesia	Servicio
1	2005-12-03	quemadura	46	Realizada	General Inalatoria	O.R.L
2	2001-02-12	dsa	2	20	General	Cirugia Pediatrica
15	2005-12-04	shahs	12566355	20	General Inalatoria	Tramatologia

**Figura 4.32** Reporte de Quirófano

**Fuente:** Elaboración Propia.

✓ Código Fuente

```
<body>
<form name="form1" method="post" action="">
<table width="100%"><tr><td width="40%">
  
</td>
  <td width="60%">&nbsp;  </td>
<td width="40%">
  
</td>
</table>
<table width="80%" border="1" align="center" cellpadding="0" cellspacing="0">
  <caption align="top">
```

<p>PACIENTE INTERVENIDOS DE EMERGENCIA DURANTE EL MES  
DE MARZO DEL 2009</p>

```
<?php $mes = trim($_POST['mes']);
      $año = trim($_POST['año']);
      if ($mes==1)
          $f=Enero;
      if ($mes==2)
          $f=Febrero;
      if ($mes==3)
          $f=Marzo;
      if ($mes==4)
          $f=Abril;
      if ($mes==5)
          $f=Mayo;
      if ($mes==6)
          $f=Junio;
      if ($mes==7)
          $f=julio;
      if ($mes==8)
          $f=Agosto;
      if ($mes==9)
          $f=Septiembre;
      if ($mes==10)
          $f=Obtubre;
      if ($mes==11)
          $f=Noviembre;
      if ($mes==12)
          $f=Diciembre;
      echo $f;
      echo " ";
      echo $año;
```

?>

</caption>

<tr bordercolor="#FFFFFF" ><td bgcolor="#FFFFFF">&nbsp;  </td>

</tr>

<tr>

<th scope="col">Numero de historia</th>

<th scope="col">Fecha de ingreso</th>

<th scope="col">Tipo de intervencion </th>

<th scope="col">codigo del doctor </th>

<th scope="col">Observaci&oacute;n</th>

<th scope="col">Tipo de Anestecia </th>

```

        <th scope="col">Servicio</th>
    </tr>
    <?php
// base de datos de quirofano
    $link = mysql_connect("", "pedro", "8294")
or die ("no se ha podido conectar");
mysql_select_db("hospital")
or die("Error al tratar de seleccionar esta base");
$result = mysql_query("SELECT * from quirofano");
if (!$result) {
    echo 'Error ' . mysql_error();
    exit;
}
//base de datos de paciente
$link = mysql_connect("", "pedro", "8294")
or die ("no se ha podido conectar");
mysql_select_db("hospital")
or die("Error al tratar de seleccionar esta base");
$resulta = mysql_query("SELECT * from paciente");
if (!$resulta) {
    echo 'Error ' . mysql_error();
    exit;
}
//***** base de datos doctor
$link = mysql_connect("", "pedro", "8294")
or die ("no se ha podido conectar");
mysql_select_db("hospital")
or die("Error al tratar de seleccionar esta base");
$r = mysql_query("SELECT * from doctor");
if (!$result) {
    echo 'Error ' . mysql_error();
    exit;
}
while ($row = mysql_fetch_row($result))
{
    $numero = trim($row[0]);
    $fecha = trim($row[1]);
    $tipo = trim($row[2]);
    $codigo = trim($row[3]);
    $observacion = trim($row[4]);
    $tipo1=trim($row[5]);
    $servicio=trim($row[6]);
    echo "<tr>";

```

```

        echo "<td w>";
        echo $numero;
        echo "</td>";
    echo "<td>";
        echo $fecha;
        echo "</td>";
    echo "<td>";
        echo $tipo;
        echo "</td>";
    echo "<td>";
        echo $codigo;
        echo "</td>";

    echo "<td>";
        echo $observacion;
        echo "</td>";

        echo "<td>";
        echo $tipo1;
        echo "</td>";

        echo "<td>";
        echo $servicio;
        echo "</td>";

    echo "</tr>";
    }
    ?>
</table>
</form>
</body>
</html>

```

#### 4.11.1 Flujo de Trabajo: Pruebas

El Flujo de Trabajo Pruebas, es una de las fases más importantes del ciclo de vida de Desarrollo de Software. Un producto de software que se desarrolla se debe entregar al Cliente libre de defectos o de errores. La prueba es el proceso de ejecutar la aplicación con la intención de descubrir defectos en el programa.

En el ciclo de vida de desarrollo de la aplicación, la Fase de Prueba ocurre después de la Fase de Programación. En tal sentido se plantea la revisión del funcionamiento de los distintos módulos de la aplicación, sometiendo cada uno a una entrada, proceso y salida esperada, a fin de verificar la confiabilidad, seguridad e integridad del sistema. Las pruebas garantizan que el producto es de calidad.

#### **4.11.2 Pruebas de Integración**

Para la integración de los módulos del sistema, las pruebas se planifican utilizando la integración incremental (antítesis del enfoque del “bigbang”). El programa se construye y prueba en pequeños incrementos, en los cuales resulta fácil aislar y corregir errores. Se planea utilizar, como estrategia de integración, la integración incremental ascendente.

La prueba de integración ascendente, como su nombre lo indica, empieza la construcción y la prueba con módulos atómicos (es decir, componentes de los más bajos de la estructura del programa). Debido a que los componentes se integran de abajo hacia arriba, siempre está disponible el procesamiento requerido para los componentes subordinados a un determinado nivel y se elimina la necesidad de resguardos.

#### **4.11.3 Pruebas de Caja Negra**

Las Pruebas de Caja Negra se concentran en los requisitos funcionales del software; es decir, permiten derivar un conjunto de condiciones de entrada que ejercitarán por completo todos los requisitos funcionales de un programa. La Prueba de Caja Negra es un enfoque complementario al Enfoque de Caja Blanca, debido a que provee la posibilidad de descubrir una clase diferente de errores.

Las Pruebas de Caja Negra tratan de encontrar errores derivados por funciones incorrectas o faltantes, errores de interfaz, errores en estructura de datos, errores de comportamiento, errores de inicialización y término.

Para realizar las Pruebas de Caja Negra existe una técnica algebraica llamada "clases de equivalencia", que consiste en tratar a todos las posibles entradas y parámetros como un modelo algebraico y utilizar las clases de este modelo para probar un amplio rango de posibilidades.

A continuación se muestran las clases de equivalencias para las operaciones más importantes de “*SIREM*”

En la tabla 4.11 se muestra la Clase de Equivalencia para la Operación Crear Cliente, cabe destacar que los campos de la Operación Crear Proveedor y son los mismos que posee la Operación Crear Cliente, por lo que no se describirán en esta prueba.

**Tabla 4.11** Clase de Equivalencia para la Operación: Crear paciente

NÚMERO	CAMPOS	CASO DE PRUEBA	SALIDA
1	Numero de historia	Nulo	Ingresar número de Historia
2	Nombre	Nulo	Ingresas nombre del paciente
3	Apellido	Nulo	Ingresar Apellido del Paciente
4	Sexo	Nulo	Ingresar Sexo del paciente

**Fuente:** Elaboración Propia.



**Tabla 4.12** Clase de Equivalencia para la Operación: Crear Doctor

NÚMERO	CAMPOS	CASO DE PRUEBA	SALIDA
1	Código del Doctor	Nulo	Ingresar el código del doctor
2	Nombre	Nulo	Ingresar el Nombre del Doctor
3	Apellido	Nulo	Ingresar el Apellido del Doctor
7	Especialidad	Nulo	Ingresar La especialidad del Doctor

**Fuente:** Elaboración Propia.

**Tabla 4.13** Clase de Equivalencia para la Operación: Modificar Paciente

NÚMERO	CAMPOS	CASO DE PRUEBA	SALIDA
1	Edad	Nulo	Ingresar la edad del paciente
2	Dirección	Nulo	Ingresar dirección del paciente
3	Fecha	Nulo	Ingresar Fecha

**Fuente:** Elaboración Propia

#### 4.11.4 Conclusión de la Fase de Construcción

La implementación de los componentes del sistema, se llevó a cabo por medio de la construcción de un código que tuvo como base fundamental los diagramas logrados en los dos capítulos anteriores, lográndose así, la versión beta del sistema “*SIREM*”. Para obtener un producto con la calidad y confiabilidad necesaria, le fueron aplicadas distintas pruebas al sistema demostrando así que el producto es estable para ser usado.

#### 4.12 Fase de Transición

El objetivo de esta fase es asegurar que los requisitos se han cumplido y que el software está disponible para los usuarios finales. Por eso, ésta fase está dirigida por

la retro alimentación de los usuarios a partir de la información que se deduzca de la versión beta del sistema en funcionamiento. Para lograrlo, se siguieron los siguientes pasos:

- ✓ Se corrigen los fallos que hubiera, para lo cual se realizan las pruebas.
- ✓ Se determinan los elementos que deban ajustarse como: problemas no detectados, refinamiento y mejora de algunas características con un desarrollo adicional.
- ✓ Se actualiza la documentación correspondiente.
- ✓ Se deben descubrir riesgos no detectados anteriormente.

Los ajustes que se hagan serán específicos y de corto alcance. Ajustes estructurales mayores debieron haberse resuelto anteriormente en el ciclo de vida y deberán documentarse para futuras ampliaciones.

Estando en marcha la versión beta del sistema, se reemplaza por el sistema definitivo que se despliega entre los usuarios.

La carga de trabajo de esta fase la soportan los programadores (que hacen los cambios necesarios) y el equipo de control de calidad (que prueba los cambios).

Al final de la fase, se decide si se han cumplido los objetivos previstos y se puede determinar si es necesario empezar otro ciclo de desarrollo. Por otra parte, se puede llevar registro de los avances, para aplicarlas en próximas actualizaciones del sistema.

#### **4.12.1 Evaluación de la Fase de Transición**

El software cumple con los requisitos obtenidos en las fases de Inicio y Elaboración, obteniéndose así el funcionamiento adecuado con respecto a lo requerido por los usuarios que realizan las operaciones en el sistema.

#### **4.12.2 Flujo de Trabajo Pruebas**

Este flujo de trabajo tiene la finalidad de comprobar el correcto funcionamiento de cada uno de los módulos que se codificaron e integraron para obtener una aplicación robusta y confiable.

Se aplicaron pruebas a las operaciones más relevantes para determinar si la aplicación está trabajando correctamente y cumpliendo a cabalidad los requisitos capturados en las fases previas.

##### **4.12.2.1 Aplicación de las Pruebas**

Se hicieron un conjunto de pruebas con el personal seleccionado del Hospital Niños con la versión instalada en los servidores de desarrollo, estas pruebas se llevaron a cabo tomando como base los mismos casos de pruebas realizados en la fase de construcción.

#### **4.12.3 Conclusión de la Fase de Transición**

En la Fase de Transición no fue necesario realizar modificaciones o correcciones importantes sobre los componentes del software, demostrando así, la robustez y estabilidad de la línea base de la arquitectura, alcanzadas en fases anteriores.

Finalmente, se demostró que el sistema ha cubierto las expectativas y los requerimientos del Hospital de Niños, lo que quiere decir, que se han alcanzado los objetivos fijados en la Fase de Inicio y que los usuarios principales están satisfechos.

## CONCLUSIONES

Una vez finalizado con el desarrollo de la automatización del sistema del Departamento de Registro y Estadísticos Médicos del Hospital de Niños “Dr. Rafael Tobías Guevara”; ubicado en la ciudad de Barcelona Estado Anzoátegui, al cual nombramos: SIREM (Sistema de Información de Registros y Estadístico Médicos).

Como punto de partida, fue necesaria la comprensión del estado en que se encontró el manejo de la información del Departamento de Registro y Estadísticos Médicos para así lograr determinar, todos los procesos necesarios y sus actividades relacionadas, para la obtención de los requerimientos del nuevo sistema y de los distintos módulos que posee.

El proceso Unificado de Desarrollo de software el cual permitió visualizar de una forma clara y sencilla; el contexto del sistema y crear una arquitectura robusta, sencilla, para que en un futuro si se desea se pueda mejorar. Esto se logró mediante flujos de requisitos, análisis, diseño, implementación y prueba.

En la fase de inicio se capturaron los requisitos necesarios para dar origen al primer modelo general del sistema, se identificaron los actores y casos de uso más importantes del sistema SIREM, que permitió una visión general del funcionamiento del sistema.

En la fase de elaboración se recopilaron los requisitos pendientes y se estableció una línea base de la arquitectura sólida del sistema SIREM, además se obtuvo información suficiente para realizar un análisis factible del sistema.

En la fase de construcción se detallaron y analizaron los requisitos restantes, se diseñaron las interfaces del sistema, se realizó la implementación y prueba del sistema, hasta el punto de estar listo para ser utilizado por el usuario final.

La interfaz de usuario del sistema SIREM; fue lo más clara y entendible para ser manejada por el usuario, permitiendo no solo la ejecución de un comando mediante un solo camino mediante el uso del Mouse, si no también mediante el uso de combinación de teclas (Hot Keys).

Los reportes generados son de gran ayuda a las personas; que llevan las estadísticas del hospital de niños.

El diseño de la base de datos es robusto, se utilizaron las técnicas del modelo relacional para garantizar la fiabilidad de los datos, ya que el sistema maneja un gran flujo de información.

Este proyecto de investigación cumplió con los objetivos planteados; como fue la automatización de la información del Hospital de Niños “Tobías Guevara”.

## RECOMENDACIONES

1. Es recomendable darle utilidad al sistema SIREM, ya que este es un proyecto de investigación; que puede ayudar al hospital a mejorar su funcionamiento.
2. Es aconsejable periódicamente implementar rutinas que permitan el mantenimiento de las bases de datos del sistema SIREM, como sería efectuar el respaldo de la información, chequear su tamaño en el disco, restaurarla en caso de daños, así se facilitarían las tareas de soporte técnico.
3. Para la instalación del sistema SIREM se recomienda tener instalado en el computador Apache versión: Apache/1.3.33 (Win32), PHP versión: 5.0.4, MySQL versión : 4.1.10a-nt - extensión : mysql y Cualquier navegador de Internet y así mismo poseer un hardware con las características mínimas siguientes: Procesador de Pentium en adelante y 128 MB de Ram.

## BIBLIOGRAFÍA

ALVAREZ, M. WYSMARI y BRITO, M. (2001). **“Estudio Sistémico del Servicio Médico de Salud ocupacional de una empresa fabricante de tubos de acero, para el Diseño de un sistema de control de gestión de información”**, TRABAJO DE GRADO DE SISTEMAS, Universidad de Oriente, Puerto La Cruz, Venezuela.

ARMIRA, R. Y LOPEZ, C. (2001). **“Estudio del clima organizacional para mejorar el rendimiento en el área operativa y administrativa de los Departamentos Críticos del Hospital de Veteranos “Dr. Julio Rodríguez””** TRABAJO DE GRADO DE SISTEMAS, Universidad de Oriente, Puerto La Cruz, Venezuela.

ELMASRI, R y NAVATHE, S. (2000). **“Sistemas de Bases de Datos. Conceptos Fundamentales”**. 2da Edición, Editorial Addison Wesley Longman, México.

FAIRLEY, R. (1.987). **“Ingeniería de Software”**. 2da Edición, Editorial McGraw-Hill, México.

GOTILLA, M. y GUACARAN, G. (2001). **“Diseño de un Plan Estratégico para el Departamento de Gineco-obstetrica del Hospital Universitario “Dr. Luis Razetti”** del Estado Anzoátegui, TRABAJO DE GRADO DE SISTEMAS, Universidad de Oriente, Puerto La Cruz, Venezuela.

HALVORSON, M. (1999). **“Aprenda Microsoft Visual Basic 6.0”**. 1ra Edición, Editorial McGraw-Hill España.



KENDALL y KENDALL, (1997). **“Análisis y Diseño de Sistemas”**, Tercera Edición, Prentice Hall, México.

MONTILVA, J. (1.995). **“Desarrollo de Sistemas de Información”**. Universidad de los Andes Consejo de Publicaciones, 2da Edición, Venezuela.

RUMBAUGH, J. JACOBSON, I. BOOCH, G. (2000). **“El Proceso Unificado del Desarrollo de Software”**. 1ra Edición, Editorial Pearson Educación, España.

SENN, J. (1.993). **“Análisis y Diseño de Sistemas de Información”**. 2da Edición, Editorial Mc. Graw-Hill, México.

URBIN, B. y CEDEÑO, E. (2001). **“Diseño de un plan estratégico para el departamento de Registro Médicos y Estadística del hospital Universitario “Dr. Luis Razetti” del estado Anzoátegui”**, TRABAJO DE GRADO DE SISTEMAS, Universidad de Oriente, Puerto La Cruz, Venezuela.



# ANEXOS

## MANUAL DE USUARIO

# MANUAL DE USUARIO

## Introducción

Bienvenidos a SIREM, sistema de información para el control de registros médicos y estadística del Hospital de niños “Dr. Rafael Tobías Guevara”, es un Software diseñado para la gestión de todas las actividades que se realizan en las diferentes áreas de trabajo que como son: Consulta externa, emergencia , hospitalización y quirófano. El cuales está diseñado para realizar la entrada de los pacientes que entran ha dicho hospital y la información de los doctores que allí elaboran, para luego así luego obtener la información de los reportes estadísticos que son necesarios para poder llevar el control.

## Requisitos del sistema

Para el buen funcionamiento del sistema se requieren los siguientes requisitos:

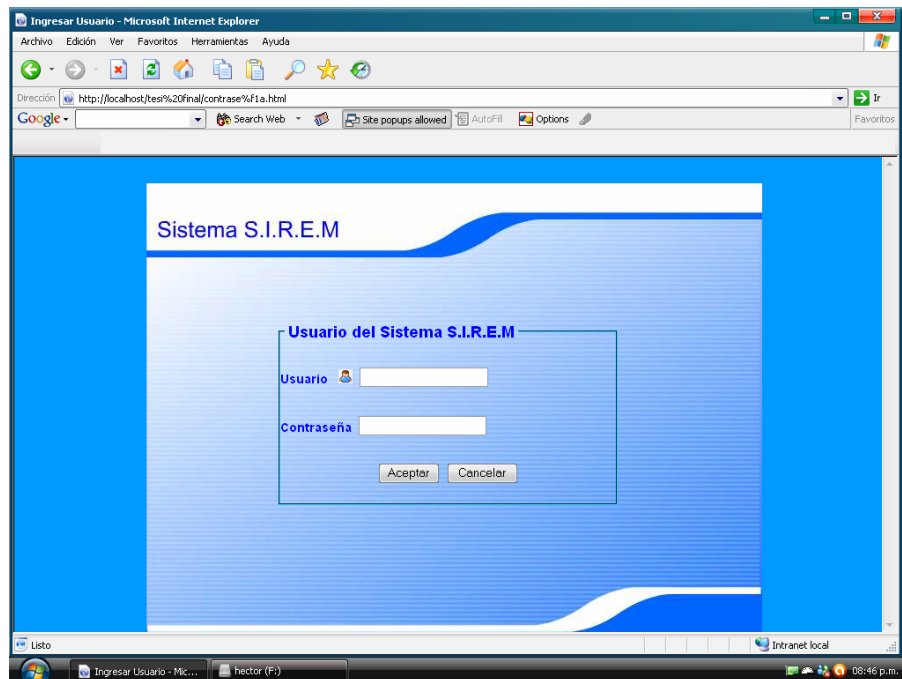
### Requisitos Mínimos

- Procesador INTEL Pentium en delante de 256Mb RAM o similar.
- Monitor SVGA Resolución 800x600 píxeles o superior.
- Impresora.
- Cualquier navegador de Internet.
- Se recomienda tener instalado: Apache versión: Apache/1.3.33 (Win32), PHP versión: 5.0.4, MySQL versión: 4.1.10a-nt - extensión: MySQL.

## Inicio

Al iniciar el sistema se mostrará la pantalla de la figura. Es este caso el usuario puede ser nombre de la persona que va a manejar el Software o cualquier otro nombre y la

cedula o cualquier otro numero va hacer la contraseña, el usuario tiene que ser letra no se permite números y la contraseña números no se permiten letras. Una vez introducido los datos correctos se inician la secciones de trabajos de lo contrario no puede ingresar al sistema. Ver figura 1.



**Figura 1.** Interfaz de Inicio.

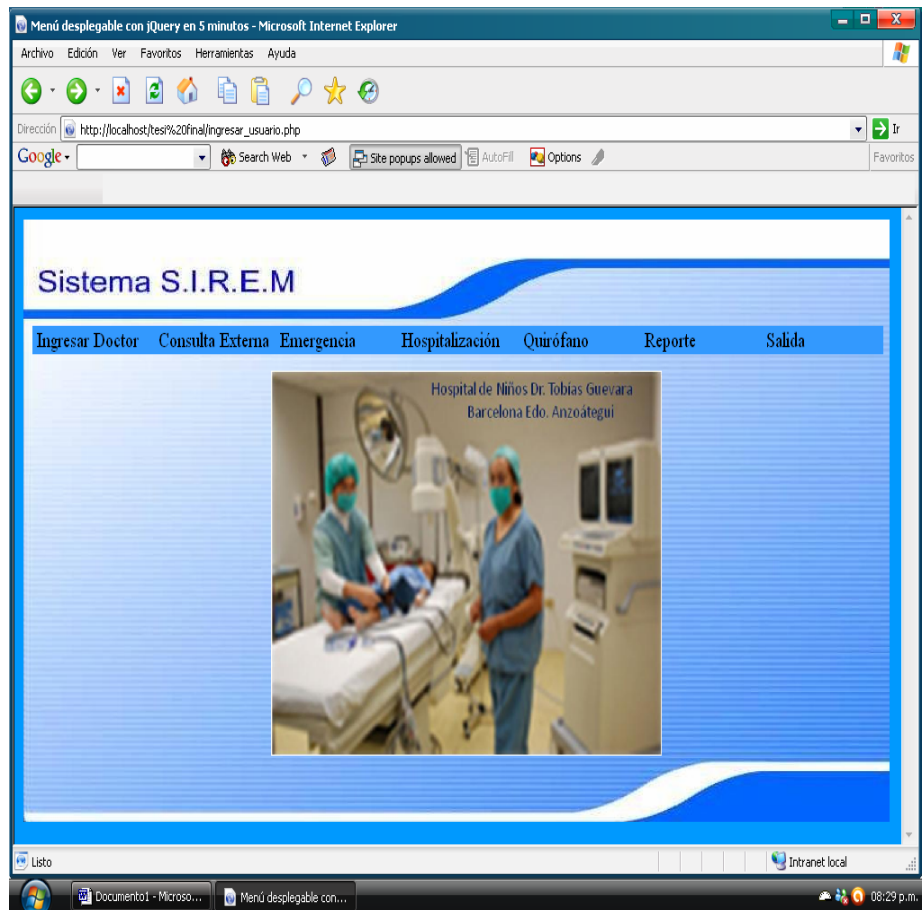
**Fuente:** Elaboración propia.

### **Inicio de secciones**

Al iniciar las sección principal del sistema se encuentran con variar actividades que realiza el sistema, las cuales están ubicada en la parte de arriba de la pantalla dependiendo de la actividad ha realizar tiene que hacer clic sobre ella las cuales se inicia una nueva sección de trabajo. Ver figura 2.

## Sección de Ingresar Doctor

Se encarga de desplegar uno submenú que esta compuesto por: ingresar doctor, modificar doctor y eliminar doctor.



**Figura 2.** Interfaz de Principal

**Fuente:** Elaboración propia.

### ➤ **Ingresar Doctor**

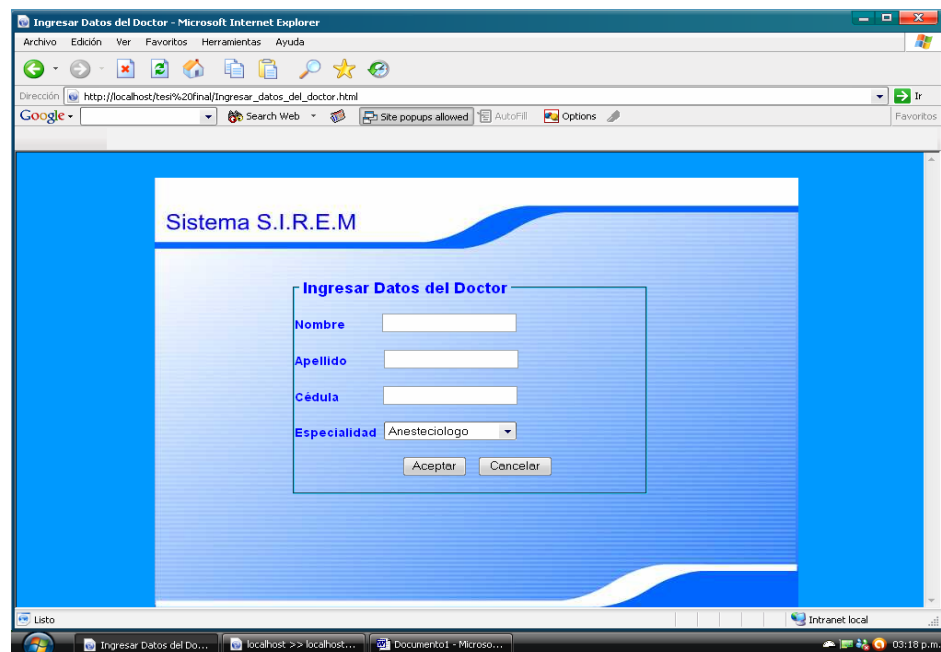
Se trata de un formulario de entrada de datos que permite al usuario del sistema ingresar los datos del doctor para así poderlo almacenar en la base de datos: nombre, apellido, cedula y la especialidad. Si no se ingresan los datos completos el sistema mostrara un mensaje de error. Ver figura 3.

➤ **Modificar Doctor**

Este formulario permite modificar solamente la especialidad del doctor. Ver figura 4.

➤ **Eliminar Doctor**

Permite eliminar todos los datos del doctor de la base de datos. Ver figura 5.



**Figura 3** Interfaz de Ingresar Doctor.

**Fuente:** Elaboración propia.

### **Sección de Consulta Externa**

Este se encarga de desplegar un submenú que va a estar compuesto por: paciente nuevo, paciente existente y ingresar consulta.

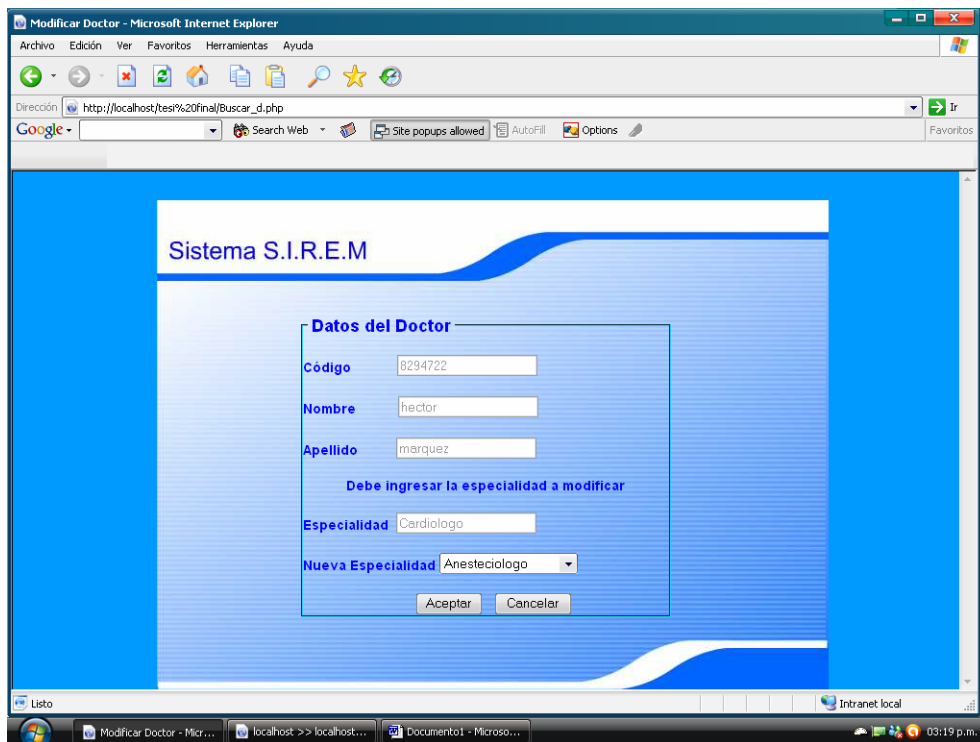
➤ **Paciente nuevo**

Permite el ingreso de los datos del paciente requerido como son: nombre, apellido, edad (0-12) años, sexo, dirección, fecha de ingreso y el numero

de historia que es asignado por el usuario del sistema. Si no se ingresan los datos completos el sistema mostrara un mensaje de error. Ver figura 6.

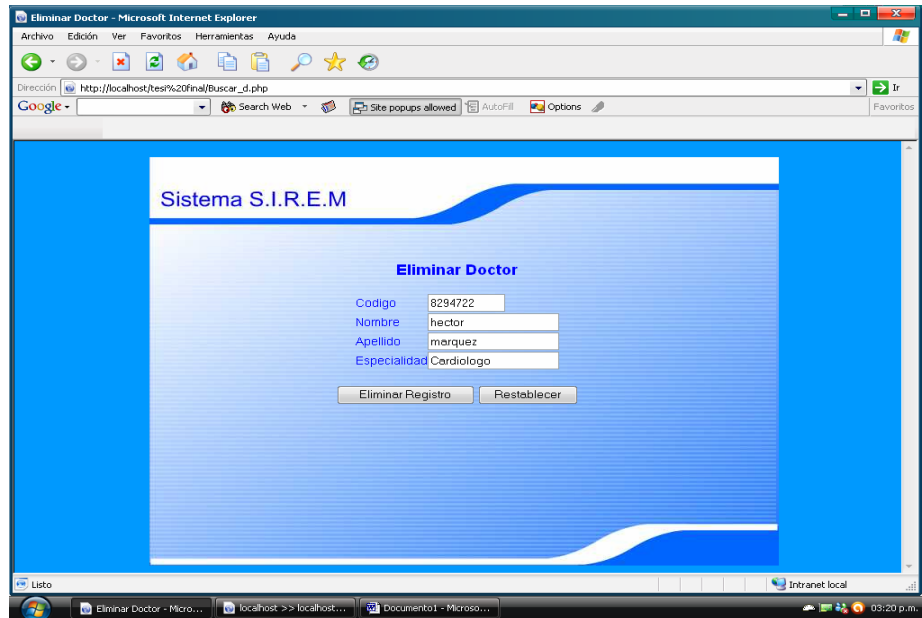
➤ **Modificar Paciente**

Permite modificar datos del paciente, esto se hace a través del número de historia que es el que se ingresa primero, una vez introducido este arrojará un formulario con todos los datos del paciente de lo contrario mostrara un mensaje de error. Los datos que se pueden modificar son: la dirección, la edad y la fecha. Ver figura 7.



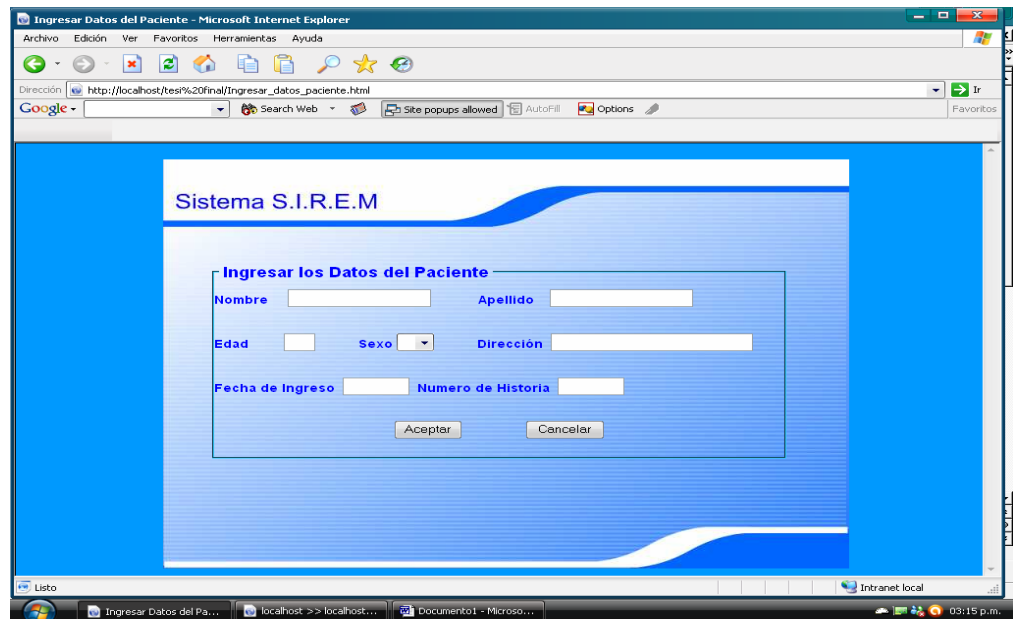
**Figura 4** Interfaz de Modifica Doctor.

**Fuente:** Elaboración propia.



**Figura 5** Interfaz de Eliminar Doctor.

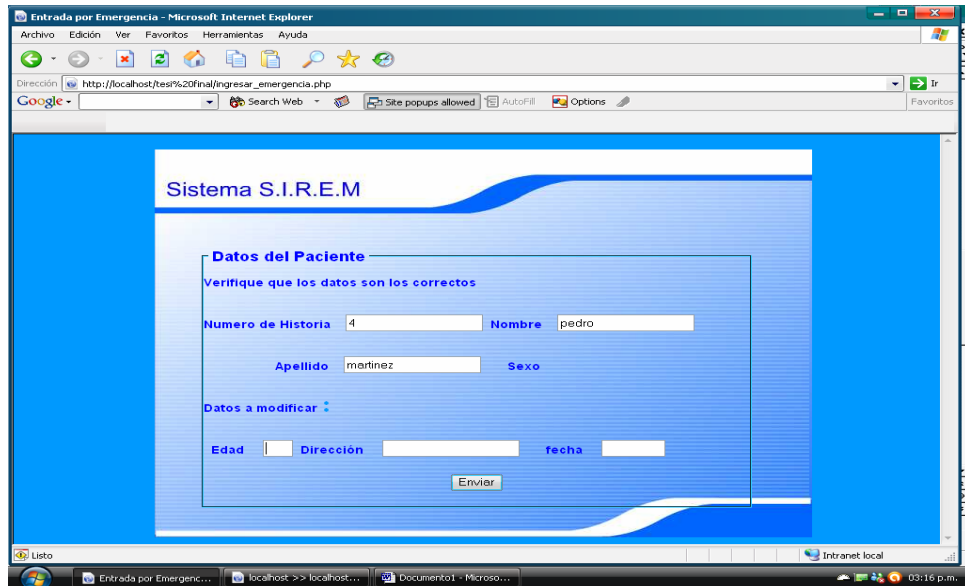
**Fuente:** Elaboración propia.



**Figura 6.** Interfaz de Paciente Nuevo

**Fuente:** Elaboración propia.





**Figura 7.** Interfaz de Modificar Paciente

**Fuente:** Elaboración propia.

➤ **Ingresar Consulta**

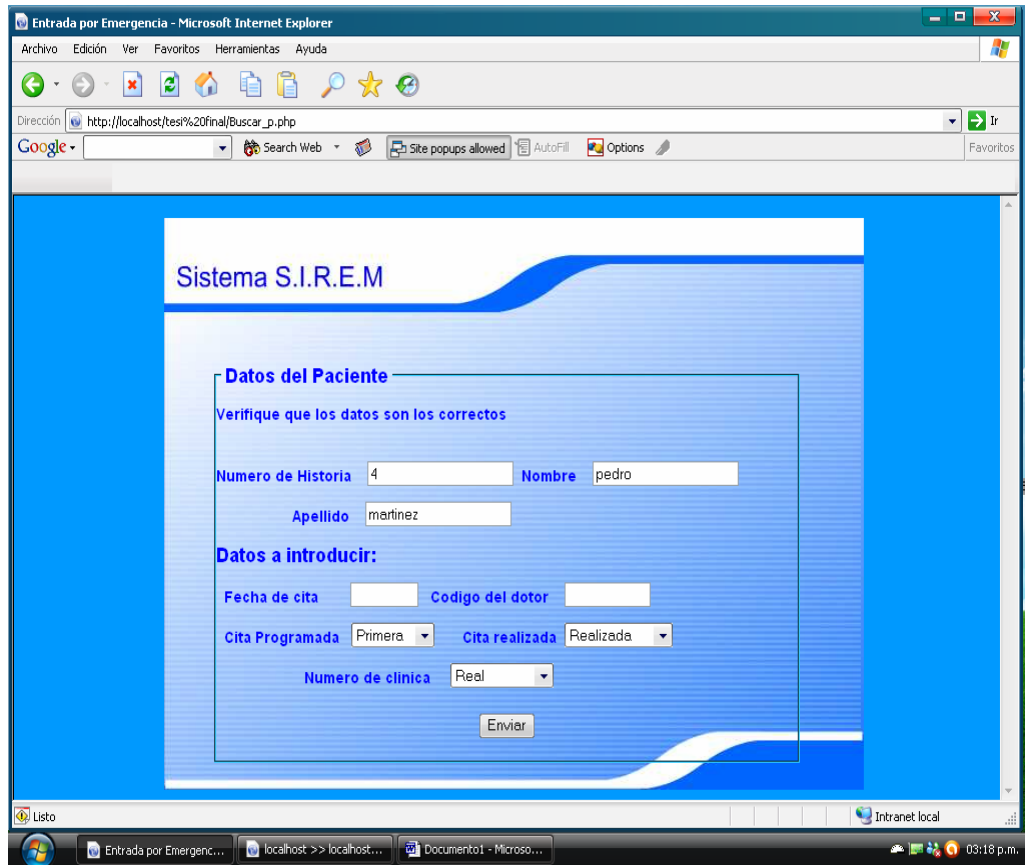
Permite la entrada de datos de la respectiva consulta que se le realiza al paciente. Esto se hace ingresando el número de historia para luego obtener un formulario con algunos datos del paciente y los nuevos datos que se requieren como son: fecha de cita, código del doctor, cita programada (primera o sucesiva), cita realizada (realizada o no) y número de clínica (real o programada). Ver figura 8.

### **Sección de Emergencia**

En esta sección se despliega lo submenú: paciente nuevo y paciente existente

➤ **Paciente nuevo**

Permite el ingreso de los datos del paciente requerido como son: nombre, apellido, edad (0-12) años, sexo, dirección, fecha de ingreso y el número de historia que es asignado por el usuario del sistema. Si no se ingresan los datos completos el sistema mostrará un mensaje de error. Ver figura 6.

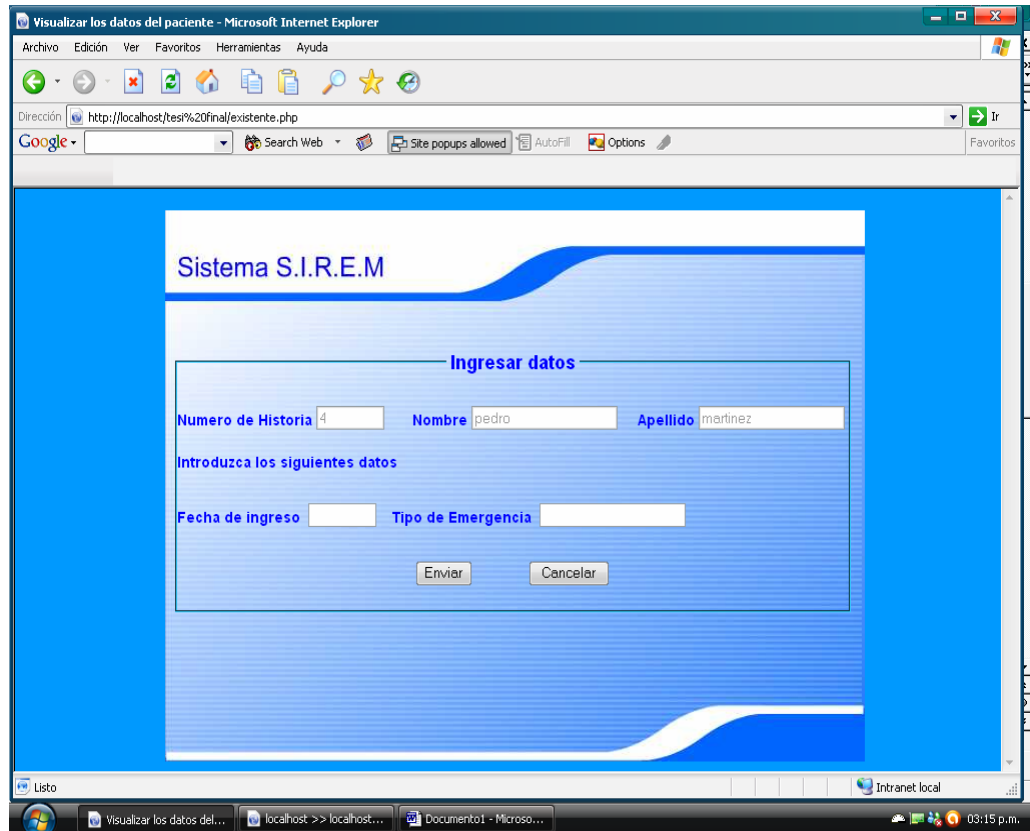


**Figura 7.** Interfaz de Ingresar Consulta.

**Fuente:** Elaboración propia.

➤ **Paciente existente**

Permite ingresar datos requeridos, esto se hace a través del número de historia, que es el que se ingresa primero, una vez introducido este arrojará un formulario con algunos datos del paciente de lo contrario mostrara un mensaje de error. Los datos requeridos son: fecha de ingreso y el tipo de emergencia que presenta el paciente, se tienen que introducir todos los datos de lo contrario no se podrá almacenar la información. Ver figura 8.



**Figura 8.** Interfaz de Paciente existente.

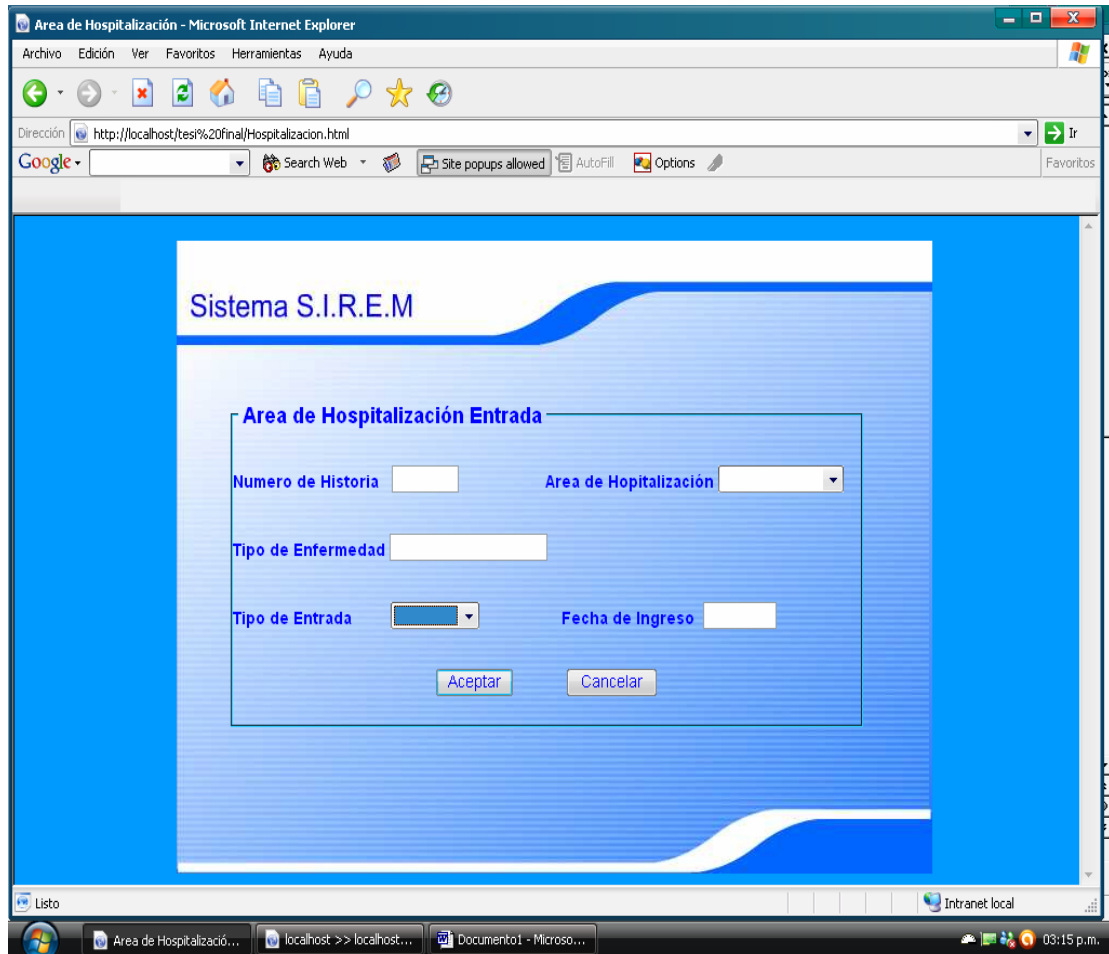
**Fuente:** Elaboración propia.

## **Sección de Hospitalización**

En esta sección se despliega lo submenú: entrada y salida de paciente.

### ➤ **Entrada Paciente**

Permite ingresar datos requeridos en el sistema los cuales se especifican en el formulario como son: número de historia, el área de hospitalización que está compuesta por varias áreas que dependiendo del tipo de enfermedad se tendrá que seleccionar una, tipo de enfermedades, tipo de entrada y fecha de ingreso. Todos los campos son requeridos de lo contrario dará un error de mensaje. Ver figura 9.

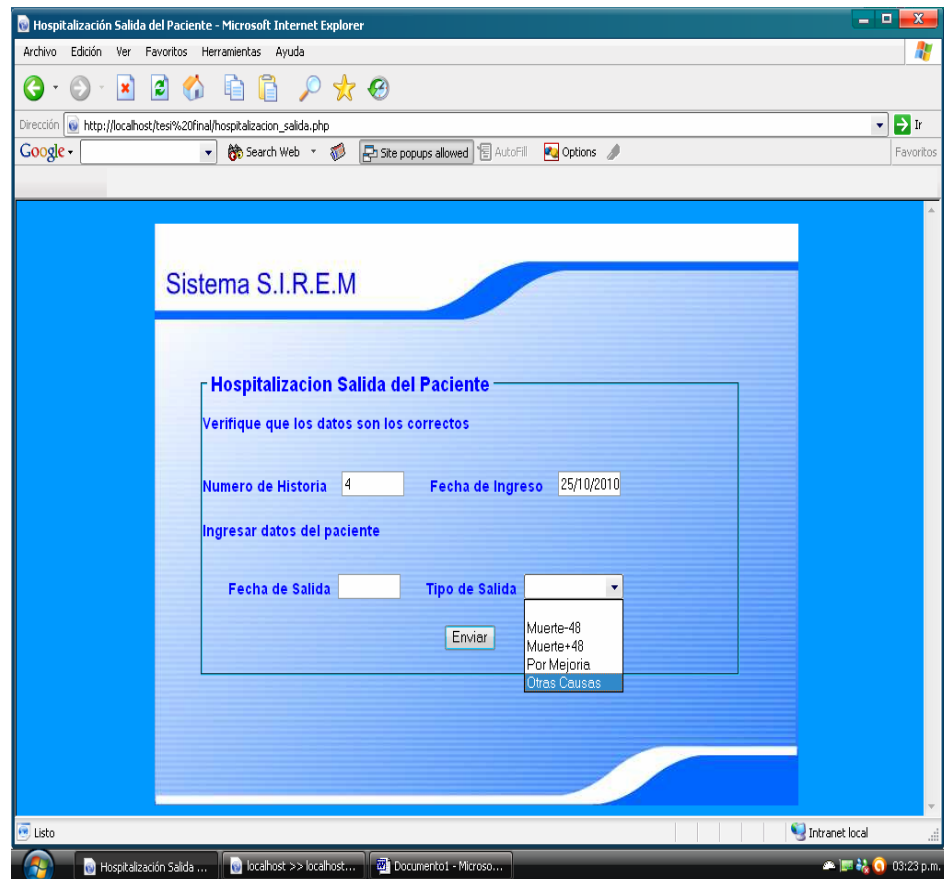


**Figura 9.** Interfaz de Entrada Paciente

**Fuente:** Elaboración propia.

➤ **Salida Paciente**

Permite ingresar los datos del tipo de salida del paciente. Esto se hace ingresando el número de historia para ver la fecha que ingreso y luego se coloca la fecha de salida y el tipo de salida que tuvo el paciente. Todos estos datos son requerido de lo contrario dará un mensaje de error. Ver figura 10.



**Figura 10.** Interfaz de Salida Paciente.

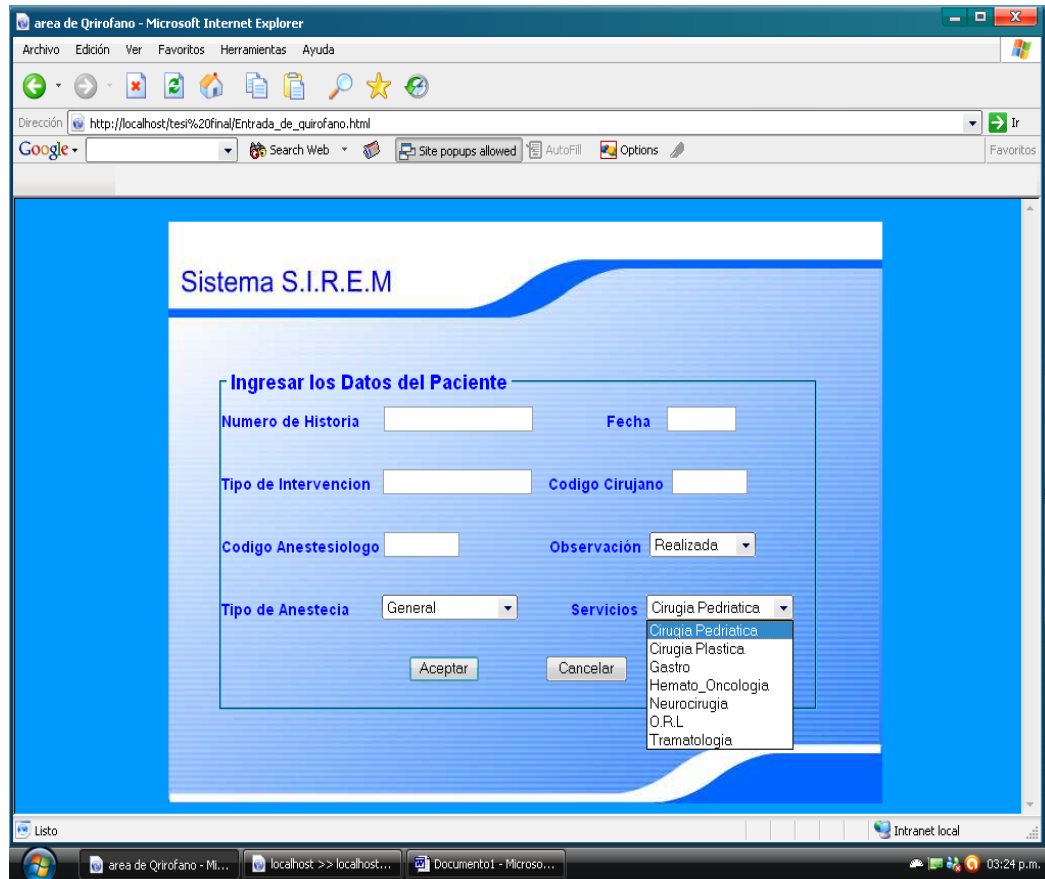
**Fuente:** Elaboración propia.

## **Sección de Quirófano**

En esta sección se despliega el submenú: entrada paciente.

### ➤ **Entrada Paciente**

Permite la entrada de datos requerido como son: numero de historia, fecha de ingreso, tipo de intervención, código del cirujano, código del anestesiólogo, observación (si la intervención se realizo o no), tipo de anestesia (general o general inalatoria) y servicio (tipo de servicio prestado). Ver figura 11.



**Figura 11.** Interfaz de Entrada Paciente.

**Fuente:** Elaboración propia.

## Sección Reportes

Permite desplegar los siguientes submenú: reporte consulta externa, reporte del tipo de enfermedad emergencia, reporte total emergencia, reporte por tipo de enfermedad hospitalización, reporte total hospitalización y reporte quirófano.

Cada uno de esto se hace introduciendo el mes y el año, para obtener los respectivos reportes. Ver figura que se muestran a continuación.

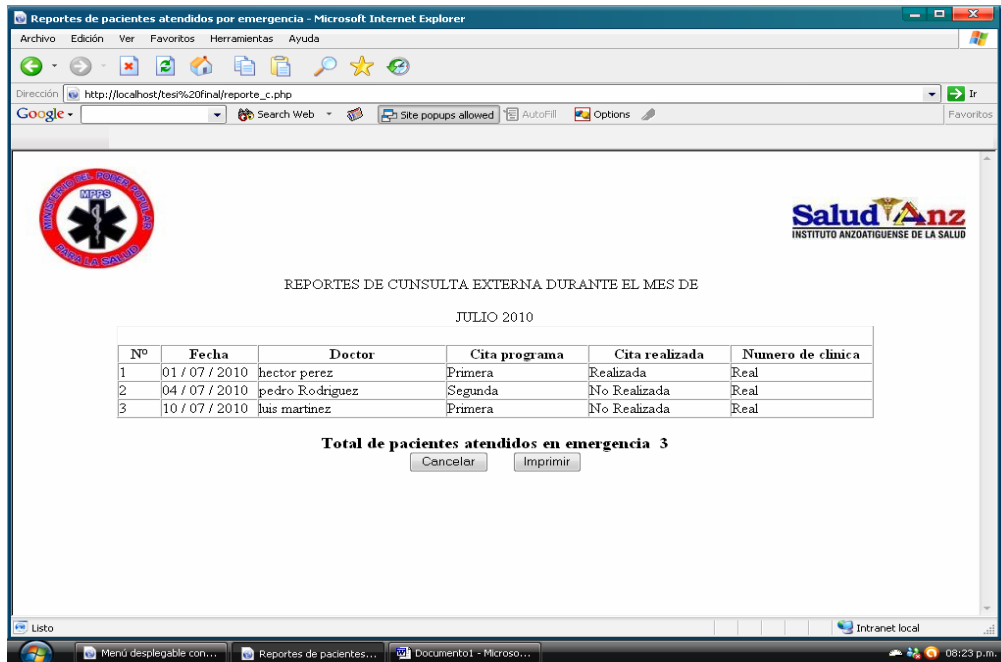


Figura 12. Interfaz de Reporte de Consulta Externa.

Fuente: Elaboración propia.

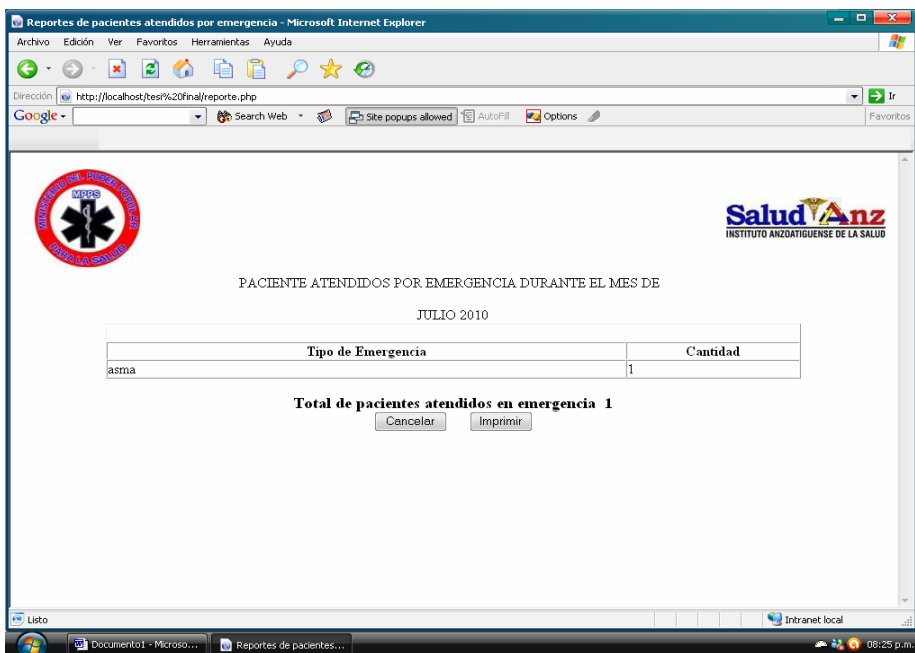


Figura 13. Interfaz de Reporte por tipo de Emergencia.

Fuente: Elaboración propia.

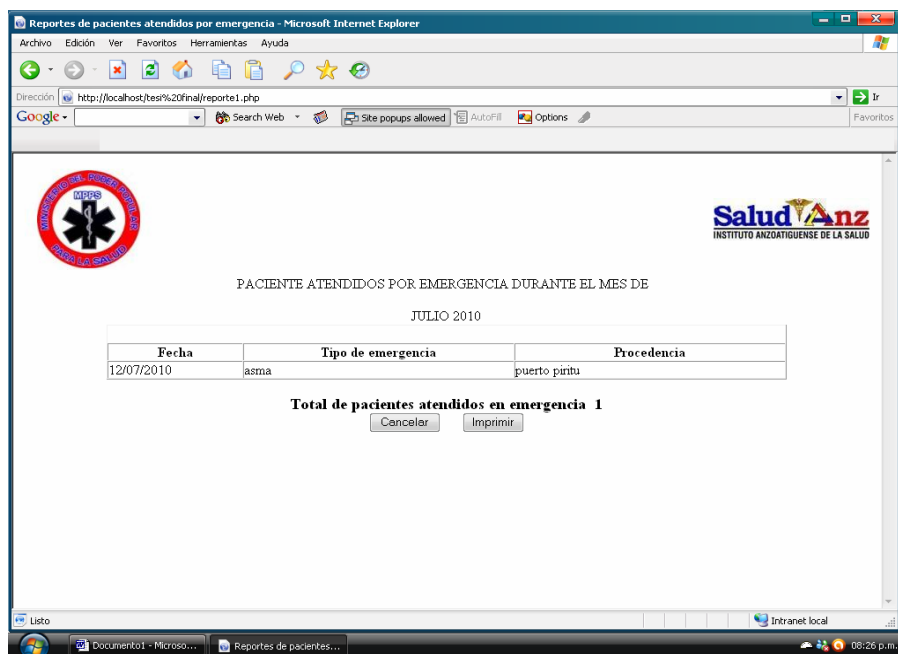


Figura 14. Interfaz de Reporte por total de Emergencia.

Fuente: Elaboración propia.

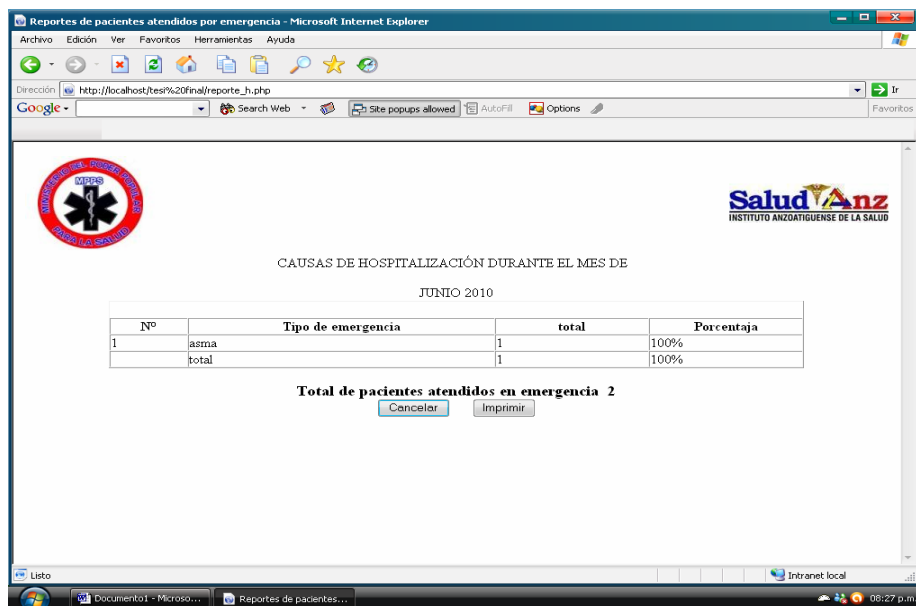
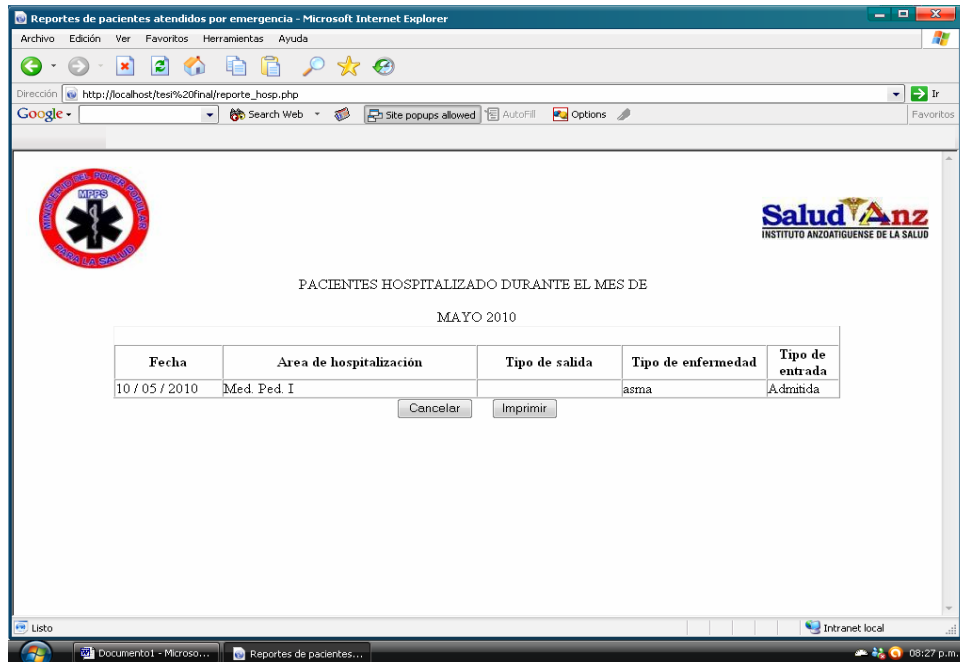


Figura 15. Interfaz de Reporte por tipo de Emergencia de Hospitalización.

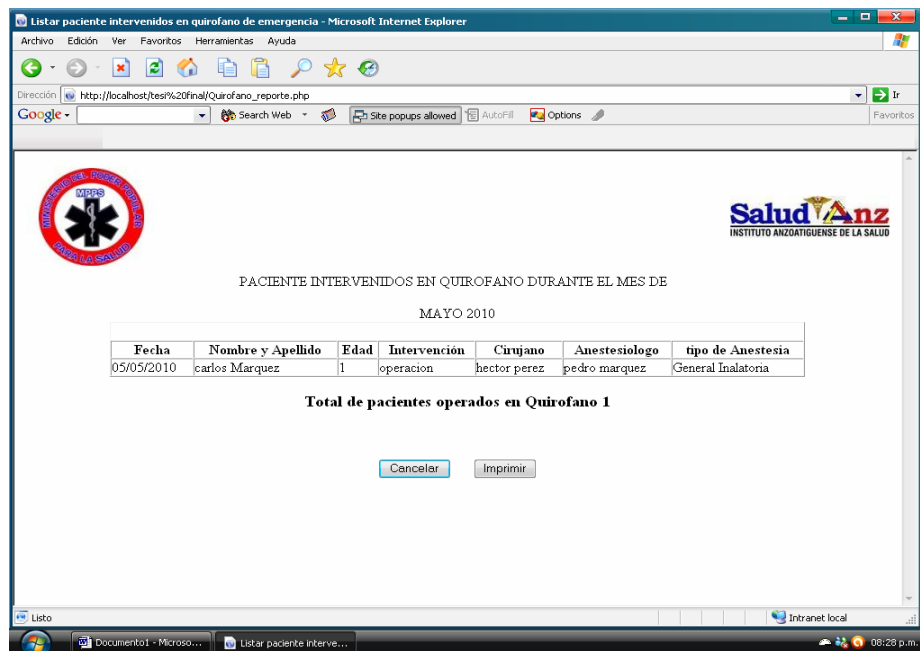
Fuente: Elaboración propia.





**Figura 16.** Interfaz de Reporte total de Hospitalización.

**Fuente:** Elaboración propia.



**Figura 17.** Interfaz de Reporte de Quirófano.

**Fuente:** Elaboración propia.





**METADATOS PARA TRABAJOS DE GRADO, TESIS Y ASCENSO:**

**CONTRIBUIDORES:**

<b>APELLIDOS Y NOMBRES</b>	<b>ROL / CÓDIGO CVLAC / E_MAIL</b>				
<b>Carrasquero, Manuel</b>	<b>ROL</b>	<b>CA</b>	<b>AS X</b>	<b>TU</b>	<b>JU</b>
	<b>CVLAC:</b>	<b>7.374.987</b>			
	<b>E_MAIL</b>	<b>manuelscm@hotmail.com</b>			
	<b>E_MAIL</b>				
<b>Rodríguez, Rhonald</b>	<b>ROL</b>	<b>CA</b>	<b>AS</b>	<b>TU</b>	<b>JU X</b>
	<b>CVLAC:</b>	<b>14.077.185</b>			
	<b>E_MAIL</b>	<b>Rhoen2003@hotmail.com</b>			
	<b>E_MAIL</b>				
<b>Caraballo, Aida</b>	<b>ROL</b>	<b>CA</b>	<b>AS</b>	<b>TU</b>	<b>JU X</b>
	<b>CVLAC:</b>	<b>15.423.826</b>			
	<b>E_MAIL</b>	<b>aidacape@gmail.com</b>			
	<b>E_MAIL</b>				
	<b>ROL</b>	<b>CA</b>	<b>AS</b>	<b>TU</b>	<b>JU</b>
	<b>CVLAC:</b>				
	<b>E_MAIL</b>				
	<b>E_MAIL</b>				

**FECHA DE DISCUSIÓN Y APROBACIÓN:**

<b>AÑO</b> <b>2010</b>	<b>MES</b> <b>08</b>	<b>DÍA</b> <b>04</b>
---------------------------	-------------------------	-------------------------

**LENGUAJE. SPA**

**METADATOS PARA TRABAJOS DE GRADO, TESIS Y ASCENSO:**

**ARCHIVO (S):**

<b>NOMBRE DE ARCHIVO</b>	<b>TIPO MIME</b>
<b>TESIS. Diseño de registro medico.doc</b>	<b>APPLICATION/MSWORD</b>

**CARACTERES EN LOS NOMBRES DE LOS ARCHIVOS:** A B C D E F G H I J K L M N O P Q  
R S T U V W X Y Z . a b c d e f g h i j k l m n o p q r s t u v w x y z . 0 1 2 3 4 5 6 7 8 9.

**ALCANCE**

**ESPACIAL:** \_\_\_\_\_ (OPCIONAL)

**TEMPORAL:** \_\_\_\_\_ (OPCIONAL)

**TÍTULO O GRADO ASOCIADO CON EL TRABAJO:**

\_\_\_\_\_ INGENIERO DE SISTEMAS Y COMPUTACIÓN \_\_\_\_\_

**NIVEL ASOCIADO CON EL TRABAJO:**

\_\_\_\_\_ PRE-GRADO \_\_\_\_\_

**ÁREA DE ESTUDIO:**

\_\_\_\_\_ DEPARTAMENTO DE COMPUTACIÓN Y SISTEMAS \_\_\_\_\_

**INSTITUCIÓN:**

\_\_\_\_\_ UNIVERSIDAD DE ORIENTE, NÚCLEO DE ANZOÁTEGUI \_\_\_\_\_

**METADATOS PARA TRABAJOS DE GRADO, TESIS Y ASCENSO:**

**DERECHOS**

DE ACUERDO CON EL ARTÍCULO 41 DEL REGLAMENTO DE TRABAJO DE GRADO. “LOS TRABAJOS DE GRADO SON DE EXCLUSIVA PROPIEDAD DE LA UNIVERSIDAD DE ORIENTE Y SÓLO PODRÁN SER UTILIZADOS PARA OTROS FINES CON EL CONSENTIMIENTO DEL CONSEJO DE NÚCLEO RESPECTIVO, QUIÉN LO PARTICIPARÁ AL CONSEJO UNIVERSITARIO, PARA SU AUTORIZACIÓN.”

---

**Martínez, Luis**  
**AUTOR**

---

**Márquez, Héctor**  
**AUTOR**

---

**Ing. Carrasquero, M.**  
**ASESOR**

---

**Ing. Caraballo, A.**  
**JURADO**

---

**Ing. Rodríguez, R.**  
**JURADO**

---

**Ing. Rojas, L.**  
**POR LA SUBCOMISION DE TESIS**