

**UNIVERSIDAD MAYOR DE SAN ANDRÉS
FACULTAD DE CIENCIAS PURAS Y NATURALES
CARRERA DE INFORMÁTICA**



PROYECTO DE GRADO

**“SISTEMA DE CONTROL Y SEGUIMIENTO DE
INVENTARIO DE FARMACOS”
CASO: CLINICA SAN DAMIAN**

**PARA OPTAR AL TÍTULO DE LICENCIATURA EN INFORMÁTICA
MENCIÓN: INGENIERÍA DE SISTEMAS INFORMÁTICOS**

**POSTULANTE : JESUS HERMOGENES VALLE QUISPE
TUTOR : LIC. EFRAÍN SILVA SÁNCHEZ
REVISOR : LIC. JAVIER HUGO REYES PACHECO**

**La Paz – Bolivia
2009**

DEDICATORIA

A mi padre Hermogenes, a quien amo, quien es mi ejemplo de vida.

A mi madre Máxima, la mujer que más amo, quien me apoya y me da fuerzas para seguir adelante, y vela para que nunca me falte nada.

A mi hermana Guissermina a quien quiero mucho, por su cariño, por aguantarme y apoyarme en esos momentos.

AGRADECIMIENTOS

A DIOS porque gracias a él tengo familia, amigos, amigas y mis estudios, que ellos forman mi pilar de vida.

A mi familia que sin su apoyo en momentos críticos quizás no hubiese salido adelante, y que gracias a los consejos de mi padre y de mi madre a ellos principalmente les doy las gracias.

Quiero de gran manera agradecer al Lic. Efraín Silva Sánchez por su paciencia y su colaboración en el transcurso de la elaboración del Proyecto de Grado, de igual manera agradezco a mi revisor Lic. Javier Hugo Reyes Pacheco por su apoyo, sugerencias y observaciones que me ayudaron a superar y alcanzar los objetivos trazados.

Gracias a mis amigos que me apoyaron a lo largo de mi vida universitaria, y que de mi parte siempre tendrán a alguien incondicional.

ÍNDICE

| | |
|---------------------------|----|
| ÍNDICE DE CONTENIDO | I |
| ÍNDICE DE FIGURAS | VI |
| ÍNDICE DE TABLAS | IX |
| RESUMEN | XI |

INDICE DE CONTENIDO

CAPITULO 1

MARCO REFERENCIAL

| | |
|--|---|
| 1.1 INTRODUCCIÓN | 1 |
| 1.2 ANTECEDENTES | 2 |
| 1.2.1 ANTECEDENTES DE LA CLÍNICA | 2 |
| 1.2.2 ANTECEDENTES DE PROYECTOS ANTERIORES | 3 |
| 1.3 OBJETO DE ESTUDIOS | 5 |
| 1.4 PRESENTACIÓN DEL PROBLEMA | 5 |
| 1.4.1 IDENTIFICACIÓN DEL PROBLEMA | 5 |
| 1.4.2 PLANTEAMIENTO DEL PROBLEMA | 6 |
| 1.4.3 FORMULACIÓN DEL PROBLEMA | 6 |
| 1.5 OBJETIVOS | 6 |
| 1.5.1 OBJETIVO GENERAL | 6 |
| 1.5.2 OBJETIVOS ESPECÍFICOS | 6 |
| 1.6 JUSTIFICACIÓN | 7 |
| 1.6.1 JUSTIFICACIÓN ECONÓMICA | 8 |
| 1.6.2 JUSTIFICACIÓN SOCIAL | 8 |
| 1.6.3 JUSTIFICACIÓN TÉCNICA | 8 |

| | |
|--|----|
| 1.7 ALCANCES Y LIMITES | 9 |
| 1.7.1 APORTES | 9 |
| 1.8 METODOLOGÍA DE INVESTIGACIÓN | 10 |
| 1.8.1 MARCO LÓGICO | 11 |
| 1.8.2 HERRAMIENTAS Y TÉCNICAS | 11 |

CAPITULO 2

MARCO TEÓRICO

| | |
|---|----|
| 2.1 INTRODUCCIÓN | 13 |
| 2.2 PROCESO UNIFICADO DE RATIONAL (RUP) | 13 |
| 2.2.1 DIRIGIDO POR CASOS DE USO | 15 |
| 2.2.2 CENTRADO EN LA ARQUITECTURA | 16 |
| 2.2.3 ITERATIVO E INCREMENTAL | 17 |
| 2.3 LENGUAJE DE MODELADO UNIFICADO(UML) | 19 |
| 2.3.1 FUNCIONES | 19 |
| 2.3.2 DIAGRAMAS DE CASOS DE USO | 19 |
| 2.3.2.1 DIAGRAMAS DE CASO DE USO DE ALTO NIVEL .. | 21 |
| 2.3.2.2 DIAGRAMAS DE CASO DE USO EXPANDIDO | 22 |
| 2.3.3 DIAGRAMA DE SECUENCIAS | 23 |
| 2.3.4 DESCRIPCIÓN DE CASOS DE USO REAL | 23 |
| 2.3.5 DIAGRAMA DE COLABORACIÓN | 24 |
| 2.3.6 DIAGRAMA DE CLASES | 24 |
| 2.4 DEFINICIÓN DEL ESQUEMA DE BASE DE DATOS | 27 |
| 2.5 INVENTARIOS | 29 |
| 2.5.1 INTRODUCCIÓN | 29 |

| | | |
|---------|---|----|
| 2.5.2 | CONCEPTOS BÁSICOS DE INVENTARIO | 29 |
| 2.5.3 | MODELO DE INVENTARIOS SIN DÉFICIT | 30 |
| 2.5.3.1 | ANÁLISIS DE ECUACIONES | 32 |
| 2.6 | MÉTRICAS DE CALIDAD | 34 |
| 2.6.1 | FUNCIONALIDAD | 35 |
| 2.6.2 | CONFIABILIDAD | 38 |
| 2.6.3 | PORTABILIDAD | 39 |
| 2.6.4 | MANTENIBILIDAD | 39 |
| 2.7 | TECNOLOGÍAS | 40 |
| 2.7.1 | SEGURIDAD DEL SISTEMA | 40 |
| 2.7.1.1 | ¿QUE ES PHP? | 41 |
| 2.7.1.2 | CARACTERÍSTICAS DE PHP | 42 |
| 2.7.2 | CONCEPTOS BÁSICOS ASOCIADOS A REDES | 43 |
| 2.7.2.1 | CLASIFICACIÓN DE LAS REDES | 43 |
| 2.7.2.2 | TOPOLOGÍAS DE REDES | 44 |
| 2.7.2.3 | COMPONENTES DE HARDWARE DE UNA RED | 45 |
| 2.8 | PRESUPUESTO | 46 |
| 2.8.1 | ANÁLISIS DE COSTOS | 46 |

CAPITULO 3

MARCO APLICATIVO

| | | |
|-------|---|----|
| 3.1 | INTRODUCCIÓN | 47 |
| 3.2 | ANÁLISIS DEL SISTEMA ACTUAL | 47 |
| 3.2 | FASE DE PLANEACIÓN | 49 |
| 3.2.1 | DESCRIPCIÓN DE LOS ACTORES z..... | 49 |

| | | |
|---------|---|----|
| 3.2.2 | IDENTIFICACIÓN DE LOS CASOS DE USO | 50 |
| 3.2.3 | CATÁLOGO DE REQUERIMIENTOS DEL SISTEMA | 51 |
| 3.2.4 | FUNCIONES BASICAS | 52 |
| 3.3 | ANÁLISIS | 54 |
| 3.3.1 | DISEÑO DE LOS CASOS DE USO | 54 |
| 3.3.1.1 | DIAGRAMA CASOS DE USO DE ALTO NIVEL | 55 |
| 3.3.1.2 | DESCRIPCIÓN CASOS DE USO DE ALTO NIVEL .. | 56 |
| 3.3.1.3 | CASOS DE USO ESENCIAL O EXPANDIDO | 57 |
| 3.3.2 | DIAGRAMA DE SECUENCIA | 68 |
| 3.3.3 | DIAGRAMA DE ESTADOS | 71 |
| 3.4 | DISEÑO | 74 |
| 3.4.1 | DIAGRAMA DE CLASES | 74 |
| 3.4.2 | DEFINICIÓN DEL ESQUEMA DE BASE DE DATOS | 75 |
| 3.4.3 | DIAGRAMA DE COLABORACION | 76 |
| 3.7 | ARQUITECTURA DEL SISTEMA | 79 |
| 3.8 | SEGURIDAD DEL SISTEMA | 80 |
| 3.9 | DISEÑO DE INTERFAZ | 81 |

CAPITULO 4

CALIDAD DE SOFTWARE

| | | |
|-----|----------------------|----|
| 4.1 | INTRODUCCIÓN | 85 |
| 4.2 | FUNCIONALIDAD | 85 |
| 4.3 | CONFIABILIDAD | 90 |
| 4.4 | PORTABILIDAD | 92 |
| 4.5 | MANTENIBILIDAD | 93 |

| | |
|--------------------------------------|----|
| 4.5.1 MANTENIMIENTO ADAPTIVO | 93 |
| 4.5.2 MANTENIMIENTO PERFECTIVO | 93 |
| 4.6 FACILIDAD DE USO | 93 |
| 4.7 CONCLUSIONES | 94 |

CAPITULO 5

MARCO CONCLUSIVO

| | |
|---------------------------|----|
| 5.1 CONCLUSIONES | 95 |
| 5.2 RECOMENDACIONES | 96 |

BIBLIOGRAFÍA

ANEXOS

- ANEXO 1. ÁRBOL DE PROBLEMAS
- ANEXO 2. ÁRBOL DE OBJETIVOS
- ANEXO 3. MARCO LÓGICO

DOCUMENTACIÓN

ÍNDICE DE FIGURAS

| | | |
|-------------|---|----|
| Figura 2.1 | Proceso de Desarrollo de software | 13 |
| Figura 2.2 | Historial de Procesos unificados | 14 |
| Figura 2.3 | Casos de uso que en laza los flujos trabajo .. | 16 |
| Figura 2.4 | Proceso Iterativo e incremental | 18 |
| Figura 2.5 | Actor | 20 |
| Figura 2.6 | Caso de uso | 21 |
| Figura 2.7 | Relación de usos | 21 |
| Figura 2.8 | Diagrama de secuencia | 23 |
| Figura 2.9 | Diagrama de clases | 26 |
| Figura 2.10 | Modelo de inventario sin déficit | 31 |
| Figura 2.11 | Componentes de costo total | 34 |
| Figura 2.12 | Clasificación de tipos de mantenimiento | 39 |
| Figura 2.13 | Funcionalidad del PHP | 41 |
| Figura 2.14 | Topologías físicas | 45 |
| Figura 3.1 | diagrama de flujo de datos | 48 |
| Figura 3.2 | Diagrama de caso de uso principal | 55 |
| Figura 3.3 | Diagrama de caso de uso de Solicitud de medicamentos | 58 |
| Figura 3.4 | Diagrama de caso de uso de Registro de Ingreso de medicamentos | 60 |
| Figura 3.5 | Diagrama de casos de uso de Registro de salida de medicamentos | 62 |

| | |
|---|----|
| Figura 3.6 Diagrama de casos de uso | |
| Registro de devolución de medicamentos | 64 |
| Figura 3.7 Diagrama de caso de uso | |
| de Control de Inventario | 66 |
| Figura 3.8 Diagrama de caso de uso | |
| de Emisión de reportes y consultas | 67 |
| Figura 3.9 Diagrama de secuencia | |
| de registro de solicitud de medicamentos | 68 |
| Figura 3.10 Diagrama de secuencia | |
| de registro de ingreso de medicamentos | 69 |
| Figura 3.11 Diagrama de secuencia | |
| de registro de salida de medicamentos | 69 |
| Figura 3.12 Diagrama de secuencia | |
| de control de inventario | 70 |
| Figura 3.13 Diagrama de secuencia | |
| de registro de devolución de medicamentos | 70 |
| Figura 3.14 Diagrama de secuencia | |
| de consulta y reportes | 71 |
| Figura 3.15 Diagrama de estado | |
| de registro de solicitud de medicamentos | 72 |
| Figura 3.16 Diagrama de estado | |
| de registro de ingreso de medicamentos | 72 |
| Figura 3.17 Diagrama de estado | |
| de registro de salida de medicamentos | 73 |

| | |
|--|----|
| Figura 3.18 Diagrama de estado de control de inventario | 73 |
| Figura 3.19 Diagrama de estado de consultas y reportes | 74 |
| Figura 3.20 Diagrama de clases | 75 |
| Figura 3.21 Modelo conceptual | 76 |
| Figura 3.22 Diagrama de colaboración de solicitud de medicamentos | 77 |
| Figura 3.23 Diagrama de colaboración ingreso de medicamentos | 77 |
| Figura 3.22 Diagrama de colaboración de solicitud de medicamentos | 78 |
| Figura 3.23 Diagrama de colaboración ingreso de medicamentos | 78 |
| Figura 3.26 Arquitectura del sistema | 79 |
| Figura 3.27 Conformación del reparto de equipos | 80 |
| Figura 3.28 Pantalla de ingreso | 81 |
| Figura 3.29 Ingreso de medicamentos | 82 |
| Figura 3.30 Registro de devolución de medicamentos | 82 |
| Figura 3.31 Registro de la unidad solicitante | 83 |
| Figura 3.32 Consultas | 83 |

ÍNDICE DE TABLAS

| | | |
|------------|--|----|
| Tabla 1.1 | Proyectos Relacionados | 4 |
| Tabla 1.2 | Especificación del Software | 12 |
| Tabla 1.3 | Especificación de Hardware | 12 |
| Tabla 2.1 | Categoría de las Funciones | 19 |
| Tabla 2.2 | Formato de Casos de Uso de Alto nivel | 22 |
| Tabla 2.3 | Formato de Caso de Uso Expandido | 22 |
| Tabla 2.4 | Caso de Uso Real | 24 |
| Tabla 2.5 | Dominios de información de Punto Función | 26 |
| Tabla 2.6 | Cálculos de los Punto Función | 26 |
| Tabla 2.7 | Costos de la clínica con respecto al sistema .. | 46 |
| Tabla 3.1 | Identificación de casos de uso | 51 |
| Tabla 3.2 | Registro de Solicitud de Medicamentos | 53 |
| Tabla 3.3 | Registro de salida de Medicamentos | 53 |
| Tabla 3.4 | Registro de salida de Medicamentos | 54 |
| Tabla 3.5 | Control de Inventario | 54 |
| Tabla 3.6 | Registro de solicitud de Medicamentos | 56 |
| Tabla 3.7 | Ingreso de Medicamentos | 56 |
| Tabla 3.8 | Registro de salida de Medicamentos | 57 |
| Tabla 3.9 | Registro de Control de Inventario | 57 |
| Tabla 3.10 | Devolución de Medicamentos | 57 |
| Tabla 3.11 | Caso de uso Esencial de registro de solicitud de medicamentos | 59 |
| Tabla 3.12 | Caso de uso de registro de ingreso | |

| | |
|---|----|
| de medicamentos | 60 |
| Tabla 3.13 Caso de uso de Registro | |
| de Salida de medicamentos | 62 |
| Tabla 3. 14 Caso de uso de Devolución | |
| de medicamentos | 64 |
| Tabla 3.15 Caso de de Control de inventario | 66 |
| Tabla 4.1 Entradas para el cálculo de funcionalidad | 85 |
| Tabla 4.2 Calculo de puntos de función sin ajustar | 86 |
| Tabla 4.3 Ajuste de complejidad del punto función | 87 |
| Tabla 4.4 Valores de ajuste de complejidad | 88 |
| Tabla 4.5 Cálculo de la confiabilidad | |
| de cada instancia de los módulos | 89 |
| Tabla 4.6 Resultados para el cálculo | |
| de facilidad de uso | 92 |

RESUMEN

La clínica “San Damián” es una clínica que satisface demandas de medicamentos a médicos, odontólogos, laboratorios, clínicas, farmacias, pacientes y otros, lo cual a influido en el crecimiento de su clientela provocando como es lógico un crecimiento considerable de transacciones con información relevante, este fenómeno provoco procesos manuales lentos, también afecto directamente a los directivos de la Clínica en la toma de decisiones, ya que el control de los inventarios se torno en un proceso muy complejo.

El presente proyecto de grado, se presenta como alternativa de solución a estos problemas de crecimiento a través del desarrollo de un sistema de control y seguimiento de inventario de fármacos esta herramienta permitirá un adecuado control del inventario que cumpla con todos los requerimientos cambiantes de la clínica y de los usuarios a través de registros de transacciones, informes que apoyen a los inventarios físicos, elaboración de reportes de clientes, proveedores, ítems y otros que van de acuerdo a las exigencias de las áreas involucradas.

El presente proyecto se divide en 5 capítulos que a continuación se los describe:

Cáp. I *Comprende las generalidades del proyecto de grado.*

Cáp. II *Se explica todos los conceptos que serán utilizados en el desarrollo del proyecto.*

Cáp. III *Comprende la Planeación, análisis, diseño e implantación del proyecto.*

Cáp. IV *Comprende la calidad del software, que permite evaluar el sistema .*

Cáp. V *En este capítulo se encuentra las conclusiones del proyecto y las recomendaciones.*

ABSTRACT

The clinic "San Damian" is a clinic that meets demands for drugs to doctors, dentists, laboratories, clinics, pharmacies, patients and others, which to influence the growth of its customer naturally causing a considerable growth of information transactions relevant, this phenomenon causes slow manual processes, also directly affect the clinic managers in making decisions, as the inventory control becomes an extremely complex process. This draft grade, provides an alternative solution to these problems of growth through the development of a system of inventory control and tracking of drugs this tool will allow an adequate control of inventory that meets all the changing requirements of the clinic and users through the transaction logs, reports that support physical inventories, compilation of reports from customers, suppliers, and other items that are in line with the requirements of the areas involved.

This project is divided into 5 chapters which are described below:

Chap. I project includes the general degree.

Chap. II It explains all the concepts that will be used in developing the project.

Chap. III includes the planning, analysis, design and implementation of the project.

Chap. IV includes the quality of software that allows for evaluating the system.

Chap. This chapter V is the project's findings and recommendations.

CAPÍTULO I

MARCO REFERENCIAL

MARCO REFERENCIAL

1.1 INTRODUCCIÓN

Hoy en día, la informática en red se ha convertido en un factor importante en la vida de una empresa la razón principal implica la cantidad de información que actualmente se maneja, hace que el tratamiento automático de la información sea realmente útil y necesario.

En la actualidad los sistemas de información están basados en computadoras que son objetos de gran consideración en la toma de decisiones oportunas, confiables y efectivas en cuanto a técnicas de planificación, programación y administración con el fin de garantizar su éxito, limitar el riesgo y reducir costos y aumentar las ganancias.

Debido a esta razón, nace la idea de automatizar las actividades cotidianas en las organizaciones; cabe mencionar el vertiginoso avance de las tele comunicaciones y el progreso que han experimentado las ciencias informáticas que obliga a estar a tono y entrar al moderno mundo de la tecnología, ser competitivos y no que darse relegados en las tareas que proporcionan beneficios para proyectarse al futuro.

El análisis hecho en el Almacén, han identificado claramente falencias que afectan el funcionamiento de la Clínica “San Damián”, situación que se ha hecho evidente en la forma como se ejecutan los procesos y funciones propias de estas áreas.

Por lo tanto, estos aspectos son importantes para la elaboración y diseño de

sistemas de información, así satisfacer los requerimientos de los usuarios y mejorar las tareas de inventarios de la clínica que se especializa en el área de salud.

Actualmente la clínica “San Damián” no cuenta con un sistema informático que coadyuvé en estas tareas con una herramienta de control que ayude a superar estas falencias, optimizar la toma de decisiones que está estrechamente relacionado con los procesos de planificación y control, logrando así sus objetivos.

De acuerdo a esto, propongo la elaboración de un “Sistema de control y seguimiento de inventario de fármacos”, que servirá de apoyo a la dirección general la cual carece de información precisa y oportuna.

El área de inventarios de la clínica se encarga de registrar, clasificar todos los movimientos de almacén. Además de controlar las adquisiciones y venta de los mismos. Estos procesos se realizan manualmente, lo que representa un esfuerzo por parte de los encargados de esta área por que cada vez se tarda más en la ubicación de un medicamento además de sus características básicas, si contaran con un sistema de información, se realizaría sus tareas en menor tiempo

1.2 ANTECEDENTES

1.2.1 ANTECEDENTES DE LA CLÍNICA

La clínica fue creada en el año 1980, sin fines de lucro apoyando a la salud de los pacientes, durante los primeros año fue una dura batalla para la clínica ya que no se contaba con el personal idóneo y suficiente para cubrir los requerimientos de los pacientes. Con el transcurrir del tiempo la universidad

mayor de san Andrés fue introduciendo al mercado nuevos profesionales en salud por lo cual se abasteció las necesidades de los pacientes en las diferentes especialidades como ser ginecología, traumatología, cirugías, et c.

Llegando a abrir nuevas sucursales en los diferentes lugares en el área urbana de la ciudad de La Paz como ser Diagnoslab ubicado en la zona sur, que permite hacer consultas especializadas en laboratorio. Clínica san Damián ubicada en villa Copacabana que atiende los mismos servicios de ginecología, traumatología, cirugías, etc.

El fin sigue siendo el mismo, llegar a más lugares para ofrecer los servicios de salud anteriormente mencionados. Las metas nuevas son coadyuvar en la salud de los pacientes, y promover el control de la salud, que los controles se los realicen al menos una vez al año y no solo se preocupen cuando ya es demasiado tarde.

1.2.2 ANTECEDENTES DE PROYECTOS ANTERIORES

El pedido de los medicamentos se realiza (antes que se acabe en stock) a la dirección de la clínica, la comercialización o venta de estos lo realiza el encargado de almacén de forma manual y lenta, lo cual dificulta hacer más contratos y encubrir la clínica a otros estatus.

Es por este motivo, la clínica requiere un sistema automatizado y apropiado para satisfacer de manera más adecuada y eficiente los requerimientos del personal encargado del área más importante de la clínica.

Por lo general un buen control de inventarios constituye un pilar para la planificación de operaciones y en la generación de información que ayude a

la toma de decisiones y de esta manera anular el gran problema que no solo acomete a esta clínica sino a muchas que colapsan por la falta de este instrumento.

A continuación se cita proyectos de grado con sultados de la biblioteca de Informática que trata temas relacionados con este documento pero se enfoca áreas específicas, se tomo en cuenta algunas que reflejan el contexto a estudiar.

Tabla 1.1 Proyectos Relacionados

| TITULO | AUTOR | SÍNTESIS | ENTIDAD |
|--|--|--|--------------------------------|
| <i>Sistema de control de inventarios para almacenes CRESPAL S.A.</i> | <i>Juan Lucio Ramos Palle, 2006</i> | <i>Se trata de un sistema que mejora el control de entradas y salidas de medicamentos desde y hasta almacenes.</i> | <i>Almacén Crespal S.A.</i> |
| <i>Sistema de almacenes y producción SOCOVIAL.</i> | <i>Alex Alberto Machicado Boetano.</i> | <i>Es un sistema de información para el control de inventarios en almacenes y mejora el control productivo.</i> | <i>Empresa "SOCOVIAL"</i> |
| <i>Sistema integrado de administración y gestión CREATRONIC SRL</i> | <i>Carla Hortensia Nadeau Inchausti.</i> | <i>Es un sistema que controla la información que se genera en cada proceso que se desarrolla en almacén.</i> | <i>Empresa CREATRONICA SRL</i> |

Fuente: Creación Propia

De todos los trabajos que se ha podido consultar en ninguno enfoca problemas relacionados con procesos de una Clínica por lo que el presente proyecto es un aporte a esta área.

1.3 OBJETO DE ESTUDIOS O INVESTIGACIÓN

Se realizara el estudio de métodos y sistemas de inventarios que existen, y ver como serán utilizados dentro de la Clínica y tratar de implementa r un modelo, sistema o método. Que pueda ser utilizado. Un estudio de modelos de inventarios sin déficit.

1.4 PRESENTACIÓN DEL PROBLEMA

1.4.1 IDENTIFICACIÓN DEL PROBLEMA

En base a estudios realizados de la información que se maneja y genera en almacén de la Clínica “San Damián”, se identifican los siguientes problemas.

- *El registro y búsqueda de los medicamentos es manual, se emplea demasiado tiempo.*
- *La actualización periódica de los registros de medicamentos es manual y deficiente.*
- *El registro de compra de medicamentos es de forma manual.*
- *El abastecimiento de medicamentos no se lo pronostica de manera óptima.*
- *Se emplea mucho tiempo en la consulta de manuales para conocer las características y funciones de los medicamentos.*
- *Se demora en la ubicación de los proveedores de medicamentos.*

- *La elaboración de informes por sucursales de existencia y abastecimiento de medicamentos no se proporciona a tiempo.*
- *La dirección no cuenta con información oportuna para una buena toma de decisiones.*

Para identificar mejor estos problemas que se encuentran centralizados en el almacén se puede observar el árbol de problemas. (ver anexos)

1.4.2 PLANTEAMIENTO DEL PROBLEMA

Debido al proceso manual de información de los fármacos, recolectados de cada uno de las sucursales, existe mucha retardación en la elaboración de informes y reportes, ocasionando que la asignación, control y supervisión de los fármacos por parte de Almacenes no sea adecuada.

1.4.3 FORMULACIÓN DEL PROBLEMA

El sistema de control y seguimiento de inventario de fármacos hará posible optimizar los procesos de inventariación y consultas de existencia de los fármacos de Almacén, de la clínica “San Damián”.

1.5 OBJETIVOS

1.5.1 OBJETIVO GENERAL

Diseñar y desarrollar un sistema de control y seguimiento de inventarios para la Clínica “San Damián”, que le ayude a mejorar su proceso de negocios.

1.5.2 OBJETIVOS ESPECÍFICOS

- *Diseñar una base de datos segura y confiable.*
- *Llevar el control automatizado de los fármacos de la Clínica.*
- *Controlar la adquisición y venta de los fármacos.*
- *Diseñar un modelo de inventarios para el control de entradas y salidas de los fármacos.*
- *Diseñar formularios de control de pedidos, entradas y salidas de fármacos.*
- *Controlar, buscar y registrar todos los movimientos de inventarios.*
- *Realizar un listado detallado que contenga información de todas las características de los fármacos.*
- *Realizar un registro de proveedores para su fácil ubicación.*
- *Dar información rápida a la dirección para que la toma de decisiones se tomen más rápidamente.*
- *Pronosticación exacta para el abastecimiento de los fármacos.*

El árbol de objetivos se muestra lo antes mencionado de forma estructural y se puede identificar el objetivo general. (ver anexos)

1.6 JUSTIFICACIÓN

Las justificaciones son desarrolladas de acuerdo a tres aspectos técnica, económica y social.

1.6.1 JUSTIFICACIÓN ECONÓMICA

Un sistema de control y seguimiento de inventario de fármacos, permitirá que la clínica optimice sus principales tareas, mejorando el tiempo de servicio por el sistema que será implementado en red, permitiendo al personal de la clínica realizar consultas desde su oficina.

1.6.2 JUSTIFICACIÓN SOCIAL

La facilidad de consulta demandada de los fármacos beneficiara tanto al personal operativo como al directivo, el mismo permitirá búsquedas de información rápida y oportuna al momento que se la requiera.

El sistema que controla el inventario de fármacos proporciona información rápida y oportuna a la dirección y particularmente a Almacén.

1.6.3 JUSTIFICACIÓN TÉCNICA

El Proyecto a desarrollar, se realiza por la necesidad que tiene la Clínica, ya que no cuenta con un buen control de los fármacos de almacén, optimizando así los servicios que presta el mismo.

El sistema realiza un control de inventarios, utilizando para ello la metodología Orientada a Objetos y el método RUP.

1.7 ALCANCES Y LIMITES

El presente proyecto se pretende analizar la situación actual de las operaciones dentro del Almacén, la relación con otras áreas dentro la clínica “San Damián”.

De ahí que se identifican los siguientes límites y alcances:

- *Registro y control de ingreso de los fármacos a Almacén de distintas sucursales.*
- *Registro y control de salidas de los fármacos de Almacene a distintas sucursales.*
- *El listado detallado contendrá información de los fármacos y su uso en formato texto.*
- *Emisión de reportes de stock.*

1.7.1 APORTES

Los aportes que ofrecerá este proyecto será automatizar sus procesos rutinarios, minimizar y optimizar tiempos de ejecución generando información que coadyuvé a la fácil y correcta toma de decisiones de la Clínica.

- *El usuario contara con una herramienta de automatización a la medida de sus requerimientos, para un óptimo control de las operaciones en el área operativa.*

- *Informes y reportes que emite el sistema con información acertada y oportuna que ayude al director en la toma de decisiones. El director contara con toda la información acerca del movimiento de los fármacos.*
- *El modulo de inventario facilitara las tareas de los empleados, permitiendo realizar un control adecuado de los medicamentos: caducidad, forma de almacenaje, medicamentos sin movimiento, cálculo automático del stock, evitando la situación de tener falta de fármacos.*
- *El modulo de seguridad realizara el control y registro de cualquier acceso de usuarios al sistema, además de proveer códigos de acceso y mantenimiento de los mismos, esta información es muy valiosa para la clínica para futuras auditorias.*
- *En este entendido se muestra a través de la metodología RUP y el lenguaje UML el diseño del sistema de control y seguimiento de inventario de fármacos.*

1.8 METODOLOGÍA DE INVESTIGACIÓN

Los métodos preliminares utilizados para determinar los requerimientos de los usuarios y los procesos a automatizar son:

- *Entrevistas personales con los médicos y la dirección encargada de almacén.*
- *Análisis de la situación actual.*

- *Determinación y definición de áreas.*
- *Método del marco lógico orientado a la planificación de proyectos.*
- *Metodología Orientada a Objetos.*
- *Método basado en el proceso unificado racional RUP.*
- *Diseño de la base de datos confiable de inventarios, compra y venta de fármacos.*
- *ISO-9126 para la calidad.*

1.8.1 MARCO LÓGICO

El marco lógico es una herramienta para facilitar el proceso de conceptualización, diseño y ejecución de proyectos. Su propósito es de brindar estructura al proceso de planificación y de comunicar información esencial relativa al proyecto, facilitando una visión rápida e inmediata de la estructura del proyecto.

Puede utilizarse en todas las etapas de preparación de proyecto: programación identificación, orientación, análisis, presentación ante los comités de revisión, ejecución y evaluación. [QCN, 2004]

El marco lógico que sea diseñado para este sistema se encuentra en anexos

1.8.2 HERRAMIENTAS Y TÉCNICAS

Las herramientas que se utilizarán en el desarrollo e implementación del proyecto harán uso de los siguientes elementos tanto del software como hardware;

Tabla 1.2 Especificación del Software

| SOFTWARE | |
|---------------------------------|------------------------------------|
| <i>Sistema Operativo</i> | <i>Windows XP</i> |
| <i>Lenguaje de Programación</i> | <i>PHP</i> <i>My SQL Server</i> |

Fuente: Creación Propia

Tabla 1.3 Especificación de Hardware

| HARDWARE | | |
|---|---|--|
| Almacén | Directivo | Oficinas |
| Microprocesador Pentium D Memoria 512 MB Disco Duro 80 G Monitor 15 plg. Lector de CD 52X Teclado Multimedia Mouse Impresora | Microprocesador Pentium D Memoria 512 MB Disco Duro 80 G Monitor 15 plg. Lector de CD 52X Teclado Mouse | Microprocesador Pentium IV Memoria 512 MB Disco Duro 40 G Monitor 15 plg. Teclado Mouse |

Fuente: Creación Propia

CAPÍTULO II
MARCO TEÓRICO

MARCO TEÓRICO

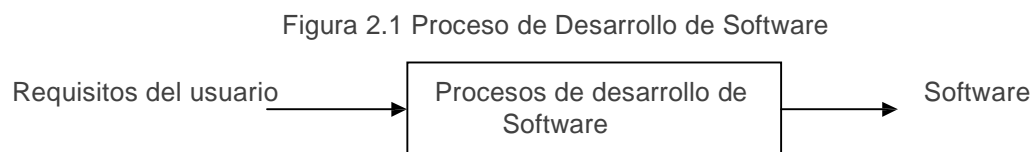
2.1 INTRODUCCIÓN

En este capítulo se introducirán los conceptos más relevantes sobre las metodologías, métodos y herramientas utilizadas para el desarrollo del presente proyecto de grado, pero no se puede dar una teoría completa es así que se tratara de presentar una base para su fácil comprensión.

2.2 PROCESO UNIFICADO DE RATIONAL (RUP)

Un proceso define quien está haciendo que, y cuando, además dice como alcanzar un determinado objetivo. En la ingeniería de software el objetivo es construir un producto de software [Jacob,2000], vale decir, que todos los proyectos necesitan de un proceso que guíe sus actividades.

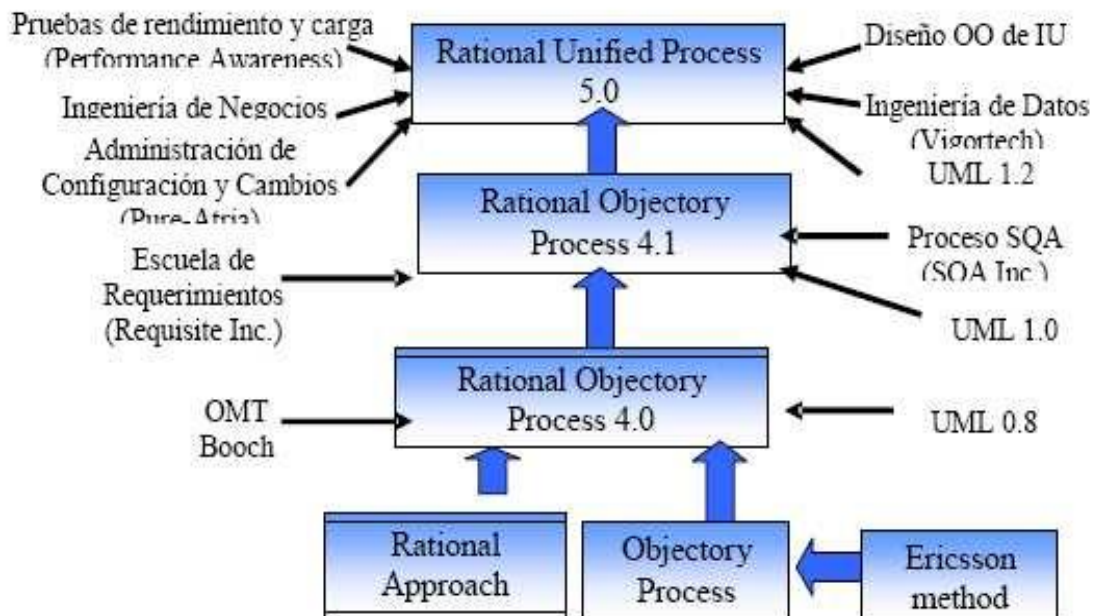
Según Jacobson en sus libros “El procesos Unificado de desarrollo de Software”, unos procesos efectivos proporcionan normas para el desarrollo eficiente de Software de calidad, captura y presenta las mejores practicas que la tecnología permite. Por tanto reduce el riesgo y hace el proyecto más predecible (ver Figura. 2.1).



Fuente: [Jacob, 2000]

Entre muchos investigadores de la orientación a objetos hay tres autores que se han destacado por sus contribuciones al uso del paradigma en todo el proceso de desarrollo: Ivar Jacobson, Grady Booch y James Rumbaugh. Luego de muchos años de trabajo individual desarrollado y difundido sus propios métodos, han unido sus teorías y su experiencia, y se han puesto a la cabeza de un formidable grupo de investigadores para contribuir dos herramientas con las cuales buscan estandarizar y por ende facilitar el uso de los objetos en la programación: El lenguaje Unificado de Modelo (UML Unified Modeling Language) y el proceso unificado rotacional para el desarrollo de programas (RUP, Rational Unified Process) mientras que UML, es ya un lenguaje maduro que ha logrado la aceptación de amplios sectores de las industria y la academia, RUP sigue siendo aún una propuesta que deberá depurarse y templarse al calor de la experiencia de su aplicación en el campo y los portes de los casos de estudio [Jacob,2000] (ver Figura. 2.2).

Figura 2.2 Historial de Procesos Unificados



Fuente: [Jacob, 2000]

RUP y UML están estrechamente relacionados entre si, pues mientras el primero establece las actividades y los criterios para conducir un sistema desde su máximo nivel de abstracción, el segundo ofrece la notación gráfica necesaria para representar los sucesos, modelos, que se obtienen de procesos de refinamiento.

RUP se define como un procesos dirigido por:

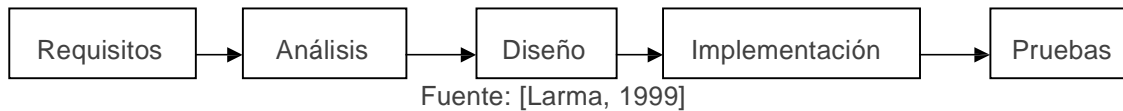
- Casos de Uso
- Centrado en la Arquitectura
- Iterativo e Incremental

2.2.1 DIRIGIDO POR CASOS DE USO

Procesos de desarrollo de software utiliza los casos de uso como una herramienta para la obtención de requisitos de usuario. Donde los casos de uso son para definir la funcionalidad del sistema, y guían al desarrollador en la construcción de la arquitectura del sistema.

La descripción obtenida de los requerimientos debe ser comprendida por casos de uso que nos ayudan a recopilar la información acerca de la interacción que tiene los usuarios en este caso actores con el sistema. Un caso de uso es una secuencia, reacciones que el sistema lleva a cabo para ofrecer un resultado de valor a algún actor, que sirven para realizar pruebas sobre los componentes desarrollados (ver Figura. 2.3). Los casos de uso enlazan los flujos de trabajo fundamentales. El proyecto progresa a través de estos flujos de trabajo, que inician en los casos de uso[Jacob 2000].

Figura 2.3 Casos de uso que en laza los flujos de trabajo fundamental



2.2.2 CENTRADO EN LA ARQUITECTURA

En el caso de software la arquitectura se refiere a un conjunto de decisiones significativas acerca de la organización de un sistema, la elección de los elementos acerca de la organización de un sistema software, la selección de los elementos estructurales a partir de las cuales se componen el sistema con su respectivo comportamiento y las interacciones entre esos elementos y la composición de esos elementos estructurales.

La necesidad de una arquitectura radica en poder comprender el sistema, es decir que todos los que están involucrados con su desarrollo deben entender el problema al cual va enfocado el sistema de software para satisfacer las demandas individuales y de la organización mediante la utilización de los diagramas definidos por UML.

La organización es un punto muy importante ya que cuanto mayor sea la organización del proyecto software mayor será la comunicación entre los desarrolladores para coordinar sus esfuerzos dividiendo el sistema en subsistemas definiendo las interfaces correctas de diseño.

Al conocer el dominio de problema y con que componentes se piensa en como conectar esos componentes para cumplir con los requisitos del sistema y realizar los modelos de casos de uso reutilizando dichos componentes.

En la arquitectura de la construcción, antes de construir un edificio, este se completa desde varios puntos de vista: estructura, condiciones el éctricas, fontanería, etc.

2.2.3 ITERATIVO E INCREMENTAL

Jacobson en su libro "El Proceso Unificado de Desarrollo de Software", explica que en esta fase proporciona la estrategia para desarrollar un producto de software en pasos pequeños manejables:

- *Planificar un poco*
- *Especificar, diseñar e implementar un poco*
- *Integrar, probar y ejecutar un poco cada iteración.*

"Un ciclo de vida iterativo se basa en el agrandamiento y perfeccionamiento secuencial de un sistema a través de múltiples ciclos de desarrollo de análisis, diseño implementación y pruebas".

El modelo incremental entrega el software en partes pequeñas pero utilizables, llamadas "incrementos" [Press, 1998]. En general, cada incremento se construye sobre aquel que ha sido entregado.

Las ventajas de un desarrollo de software con un ciclo de vida iterativo se dan gracias a la retroalimentación en cada ciclo por lo cual se crea un sistema más robusto. En cada incremento que y tiene el sistema se va perfeccionando a ún mas, lo cual permite al usuario realizar las modificaciones requeridas en el transcurso del tiempo [Press, 1998].

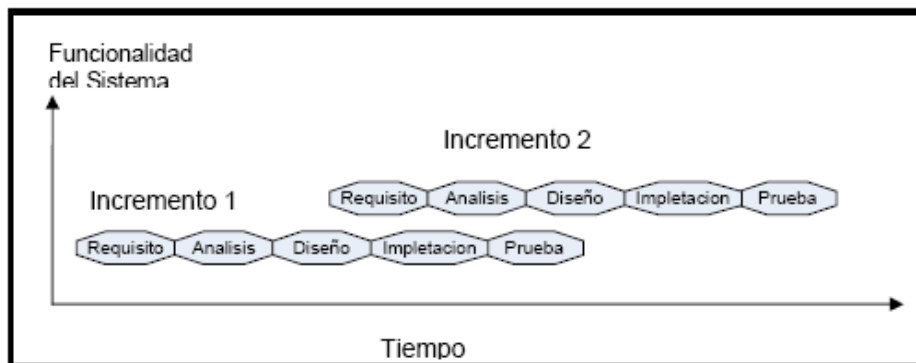
Todo sistema informático complejo supone un gran esfuerzo que puede durar desde varios meses hasta años, por lo tanto, lo más práctico es decidir en varias fases. Actualmente se puede hablar de ciclos de vida en los que se realiza varios recorridos por todas las fases.

Las ventajas de un desarrollo de software con un ciclo de vida iterativo se da gracias a la retroalimentación en cada ciclo por lo cual se crea un sistema más robusto. En cada incremento que tiene el sistema se va perfeccionando a un más, lo cual permite al usuario realizar las modificaciones requeridas en el transcurso del tiempo.

Todos los sistemas informáticos complejos suponen de un gran esfuerzo que puede durar desde varios meses hasta años; por lo tanto, lo más práctico es dividirla en varias fases.

Actualmente se puede hablar de ciclos de vida en los que se realizan varios recorridos por todas las fases. Cada recorrido por las fases se denomina iterativo del proyecto en la que se realizan varios tipos de trabajo (denominados flujos). Además cada iteración parte de la anterior incrementado o revisando la funcionalidad implementada. Se suele denominar proceso (ver Figura 2.4).

Figura 2.4 Proceso Iterativo e Incremental



Fuente: [Ferrer, 2005]

2.3 LENGUAJE DE MODELADO UNIFICADO (UML)

UML, emergió en los '90 luego de la búsqueda de un lenguaje de modelamiento que unificara a la industria, que siguió a la "guerra de métodos" de los '70 y '80. A pesar de que UML evolucionó primeramente de varios métodos orientados al objeto de segunda generación (en nivel de notación), UML no es simplemente un lenguaje para modelamiento orientado al objeto de tercera generación. Su alcance extiende su uso más allá de sus predecesores. Y es la experiencia, experimentación y una gradual adopción del estándar lo que revelará su verdadero potencial y posibilitará a las organizaciones darse cuenta de sus beneficios.

2.3.1 Funciones

Son acciones o procesos a ser realizados para lograr alcanzar un objetivo que presenta el proyecto. Las funciones pueden ser organizadas de dos tipos.

Tabla 2.1 Categoría de las Funciones

| CATEGORIA DE LA FUNCIÓN | SIGNIFICADO |
|-------------------------|--|
| EVIDENTE | Debe realizarse y el usuario debería saber que se ha realizado |
| OCULTA | Debe realizarse, aunque no es visible para el usuario. Esto hace muchos servicios técnicos subyacentes, como guardar información en un mecanismo persistente |

Fuente: [LARMAN, 1999]

2.3.2 Diagramas de Casos de Uso

Los casos de uso no son propiamente un caso de análisis, se limitan a describir

Procesos de dominio que pueden expresarse en forma narrativa en un formato estructurado de prosa y pueden ser eficaces en un proyecto de tecnología no orientada a objetos. No obstante, constituyen un paso preliminar muy útil porque describen las especificaciones de un sistema. [Jacob, 2000]

a) Actores

El actor es una entidad externa del sistema que de alguna manera participa en la historia del caso de uso. Por lo regular, estimula el sistema con eventos de entrada o recibe algo de él. Los actores están representados por el papel que desempeñan en el caso: Cliente, técnico u otro. Conviene escribir su nombre con mayúscula en la narrativa del caso para facilitar la identificación (ver Figura 2.5).

Figura 2.5 Actor

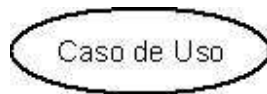


Fuente: [LARMAN, 1999]

b) Caso de uso

El caso de uso es un documento narrativo que describe la secuencia de eventos de un actor (agente externo) que utiliza un sistema para completar un proceso [Jacob, 2000] Los casos de uso son historias o casos de utilización de un sistema, no son exactamente los requerimientos, ni las especificaciones funcionales, sino que ejemplifican e incluyen tácticamente los requerimientos en las historias que narran (ver Figura. 2.6).

Figura 2.6 Caso de Uso



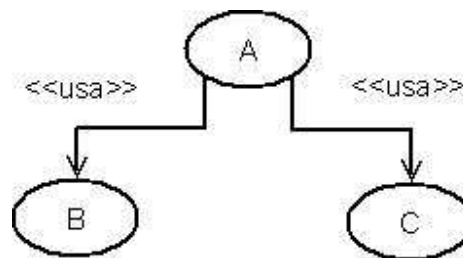
Fuente: [LARMAN, 1999]

c) Relación

Si un caso de uso inicia o contiene el comportamiento de otro se dice que usa el segundo caso, eso es una relación unidireccional [Jacob 92]. Esta relación puede presentar uno de los siguientes tipos:

- *La relación “usa”, se utiliza cuando se quiere reflejar un comportamiento común en varios casos de uso (ver Figura. 2.7).*
- *La relación “extiende”, se utiliza cuando se requiere reflejar un comportamiento opcional de un caso de uso, es decir, es cuando tiene un caso similar a otro, cuyo contexto tiene mucho más detalle.*

Figura 2.7 Relación de usos



Fuente: [Larma, 1999]

2.3.2.1 Diagramas de caso de uso de alto nivel

Son muy breves, generalmente, son descripciones de un proceso de dos o tres oraciones.

Tabla 2.2 Formato de Casos de Uso de Alto nivel

| | |
|---------------------|--|
| CASO DE USO: | Nombre del caso de uso |
| ACTORES: | Lista de actores |
| TIPO: | Primario, Secundario u opcional, Esencial o real |
| DESCRIPCIÓN: | Explica brevemente el proceso que realizan los actores en el caso de uso |

Fuente: [LARMAN, 1999]

2.3.2.2 Diagramas de Caso de Uso Expandido

Son descripciones externas que pueden contener cientos de oraciones con las cuales se realiza la descripción.

Tabla 2.3 Formato de Caso de Uso Expandido

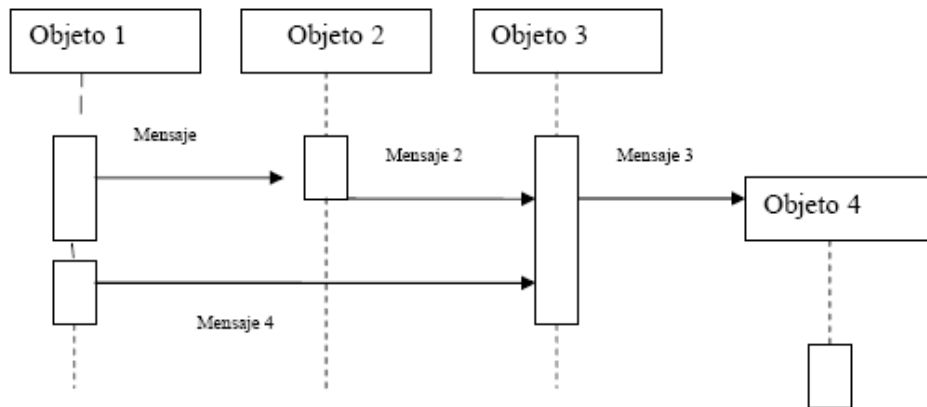
| | | | |
|-----------------------------------|---|--|--|
| Caso de Uso: | Registro de entrada de medicamentos | | |
| Actores: | Lista de actores | | |
| Propósito: | Intención del caso de Uso | | |
| RESUMEN: | Explica brevemente el proceso que realiza los actores en el caso de uso | | |
| Tipo: | Primario secundario u opcional | | |
| Referencias cruzadas: | Casos de uso y funciones también relacionadas del sistema | | |
| Curso normal de Eventos | | | |
| Acción de los actores | Respuesta de Sistema | | |
| Acciones numeradas de los actores | Descripciones numeradas de las respuestas del sistema | | |

Fuente: [LARMAN, 1999]

2.3.3 Diagrama de Secuencias

El diagrama de secuencia es un tipo de diagrama de interacción cuyo objetivo es describir el comportamiento dinámico del sistema de información haciendo énfasis en la secuencia de los mensajes intercambiados por los objetos [Jacob, 2000]. El diagrama de secuencia tiene dos dimensiones, el eje vertical representa el tiempo y el eje horizontal los diferentes objetos. El tiempo avanza desde la parte superior hacia el interior, cada objeto tiene asociado una línea de vida y focos de control. La línea de vida indica el intervalo de tiempo durante el que existe ese objeto. Un foco de control o activación muestra el periodo de tiempo (ver Figura. 2.8).

Figura 2.8 Diagrama de Secuencia



Fuente: [Jacob, 2000]

2.3.4 Descripción de Casos de Uso Real

Un caso de uso real describe el diseño concreto del mismo a partir de una tecnología particular de entrada y salida, así como de su implementación global.

Tabla 2.4 Caso de Uso Real

| | |
|---------------------|--|
| Caso de Uso: | Nombre del caso de uso |
| Actores: | Lista de Actores |
| Propósito: | Intención del caso de uso |
| Resumen: | Repetición del caso de uso de alto nivel a alguna síntesis similar |
| Tipo: | 1.- Primario, secundario u opcional. 2.- Esencial o real. |

Fuente: [LARMAN, 1999]

2.3.5 Diagrama de Colaboración

El diseño orientado a objetos tiene como primicia definir las especificaciones lógicas del software que cumplan con los requisitos funcionales, un paso esencial de esta fase es la asignación de responsabilidades entre los objetos y mostrar como interactúan a través de mensajes, expresados en diagramas de colaboración, cuyo objetivo es describir el comportamiento dinámico de información, como interactúan los objetos entre si, con que otros objetos tienen vinculados o intercambian mensajes a un determinado objeto.

Un diagrama de colaboración muestra la misma información que un diagrama de secuencia, pero de forma diferente. En los diagramas de colaboración coexiste una secuencia temporal en el eje vertical es decir, la colaboración de los mensajes en el diagrama no indica cual es el orden en el que sucede. Además la colaboración de los objetos es más flexible y permite mostrar de forma más clara cuáles son las colaboraciones entre ellos.

2.3.6 Diagrama de Clases

Es la representación de los aspectos estáticos del sistema, utilizando diversos mecanismos de abstracción (clasificación, generalización, agregación).

El diagrama de clases recoge las clases de objetos y sus asociaciones. En este diagrama se representa la estructura y el comportamiento de cada uno de los objetos del sistema y sus relaciones con los demás objetos, pero no muestran información temporal [Ferre, 2005].

Se advierten los siguientes tipos de relación:

- 1. La Asociación que representan un conjunto de enlaces entre objetos o instancias de clases.*
- 2. Herencia que indica que una subclase hereda los métodos y atributos especificados por una superclase hereda los métodos por ende la subclase, además de poseer sus propios métodos y atributos poseerá las características y atributos visibles de la superclase.*
- 3. La agregación que es un tipo de relación jerárquica entre un objeto que representa la totalidad de ese objeto y las partes que la componen. Permite el agrupamiento físico de estructuras relacionadas lógicamente.*

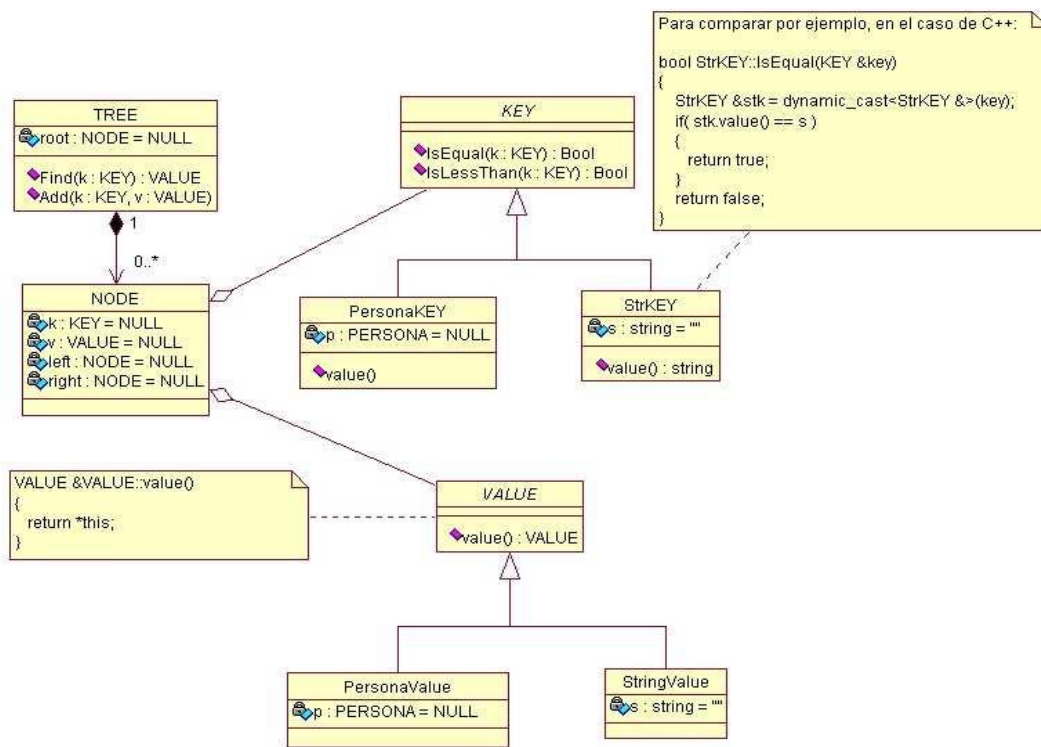
En el siguiente ejemplo se muestran las distintas clases relacionadas entre si:

- Se muestra la asociación entre la clase TREE (árbol) y la clase NODE (nodo), es decir la asociación que existe entre ambas, y la cardinalidad de uno a muchos que significa que un árbol tiene n nodos.*
- La herencia que se da entre las clases Personavalue, StringValue, cuyo padre sería la clase VALUE, dando a entender que existe un valor (value)*

en común que puede heredarse de la clase padre. Y de la misma forma se da con las clases KEY, PersonaKey, StrKey

- Cabe la importancia de notar que en la clase PersonaKey y StrKey están denotados los métodos y atributos que debería tener toda clase.

Figura 2.9 Diagrama de Clases



Fuente: [Ferre, 2005]

2.4 DEFINICIÓN DEL ESQUEMA DE BASE DE DATOS

Consiste en determinar el esquema de base de datos que se utilizará, en este caso se vio conveniente el estudio de bases de datos objeto -relacionales.

En la generalidad de las aplicaciones es necesario guardar y recuperar la información en un mecanismo de almacenamiento persistente, una base de datos relacional por ejemplo. Dado el predominio de estas últimas, a menudo se requiere su uso en vez de otras bases más manejables orientadas a objetos. De ser así, surgen varios problemas a causa de la desigualdad entre representaciones de datos orientadas a registros, y las que se orientan a objetos, se requieren servicios especiales de ambos tipos en las bases de datos relacionales [Larma, 1999].

¿Cómo mapear un objeto a un archivo o a un esquema de base de datos relacional? El patrón de representación de objetos a tablas propone definir una tabla para cada clase de objeto persistente. Los atributos de objetos que contienen tipos primitivos de datos (número, booleano, cadena y otros) se mapean en las columnas. [Larma, 1999].

Conviene contar con un medio que relacione los objetos con los registros y de asegurarse de que la repetición de la materialización de un registro no culmine en la duplicación de objetos. El patrón identificador de objetos (IDO) se propone asignar un IDO a cada registro y objeto (o al agente de un objeto). Un identificador de objetos suele ser un valor alfanumérico, es único para un objeto en específico, toda tabla de base de datos relacional tiene un IDO como clave primaria, y los objetos también contarán (directa o indirectamente) con un identificador [Larma, 1999].

Si todos los objetos se asocian a un IDO y si todas las tablas poseen una clave primaria IDO los objetos podrán mapearse de modo singular en una región de alguna tabla.

¿Cómo representar las relaciones de objetos en una tabla de una base de datos relacional? La respuesta se da en el patrón de representación de las relaciones de objetos como tablas que propone lo siguiente:

➤ *Asociación uno a uno*

- *Colocar una clave foránea de identificador de objetos en una o en las dos tablas que representan los objetos en la relación.*
- *Crear una tabla asociativa que registre los identificadores de objetos de cada relación.*

➤ *Asociaciones de uno a muchos, una colección por ejemplo*

- *Crear una tabla asociativa que registre los identificadores de cada objeto en la relación.*

➤ *Asociaciones de muchos a muchos*

- *Crear una tabla asociativa que registre todos los identificadores de objetos en la relación*

2.5 INVENTARIOS

2.5.1 INTRODUCCIÓN

La base de toda empresa es la compra o venta de servicios, ahí la importancia del manejo de inventario por parte de la misma. Este manejo contable permite a la empresa mantener el control oportuno de los productos (en nuestro caso medicamentos).

Los inventarios comprenden, además de las materias primas, productos en proceso y productos terminados o mercancías para la venta, bienes reparados para la prestación de servicios.

Ahora bien, el inventario constituye las partidas del activo corriente que están listas para la venta, es decir, toda aquella mercancía que posee una empresa en el almacén valorada al costo de adquisición, para la venta o actividades productivas. Por medio del siguiente trabajo se darán a conocer algunos conceptos básicos de todo lo relacionado a los inventarios en una empresa, métodos, sistema y control.

En contabilidad, el término inventario significa una existencia de bienes con propósitos específicos según la naturaleza de la empresa.

2.5.2 CONCEPTOS BÁSICOS DE INVENTARIO

Los inventarios son un puente de unión entre la producción y las ventas. En una empresa manufacturera el inventario equilibra la línea de producción si algunas máquinas operan a diferentes volúmenes de otras, pues una forma de compensar este desequilibrio es proporcionando inventarios temporales o bancos. Los

inventarios de materias primas, producto semiterminados y productos terminados absorben la holgura cuando fluctúan las ventas. Estos tienden a proporcionar un flujo constante de producción, facilitando su programación.

Los inventarios de materia dan flexibilidad al proceso de compra de la empresa. Sin ellos en la empresa existen una situación “de la mano a la boca”, comprándose la materia prima estrictamente necesaria para mantener el plan de producción, es decir comprando y consumiendo.

2.5.3 MODELO DE INVENTARIOS SIN DÉFICIT

Este modelo tiene como base el mantener un inventario sin falta de productos para desarrollar las actividades de cualquier empresa.

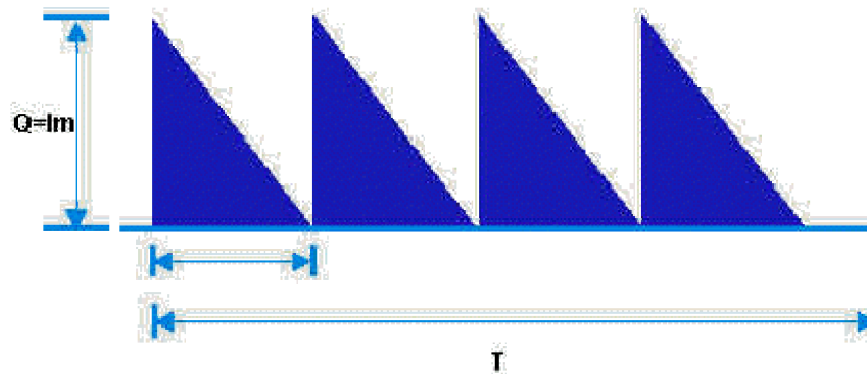
Este es un modelo de inventarios que se encuentra basado en las siguientes suposiciones:

- a. La demanda se efectúa a tasa constante.*
- b. El reemplazo es instantáneo (la tasa de reemplazo es infinita).*
- c. Todos los coeficientes de costos son constantes.*

En este modelo no se permite la falta de productos para la venta, es decir, una empresa que maneje este modelo de inventario no se puede quedar sin mercancías para la venta.

En la siguiente figura se ilustra esquemáticamente este modelo.

Figura 2.10 Modelo de inventario sin déficit



Fuente: Universidad del Centro, Facultad de Ciencias Exactas

Símbolos:

Q = Cantidad optima a pedir

Im = Inventario Máximo

t = Periodo entre pedidos

T = Periodo de Planeación

En este modelo se representan iguales el inventario máximo y la cantidad económica pedida.

Cabe mencionar que esto no siempre es verdadero.

El costo total para un periodo en este modelo esta conformado por tres componentes de costo:

- a. Costo unitario del producto (C_1)*
- b. Costo de ordenar una compra (C_2)*
- c. Costo de mantener un producto en almacén (C_3)*

El costo para un periodo e stará conformado de la siguiente manera:

Costo por periodo = [Costo unitario por periodo] + [Costo de ordenar un pedido] + [Costo de mantener el inventario en un periodo]

El costo total para el periodo de planeación estará conformado de la manera siguiente:

Costo total = Costo por periodo x Numero de pedidos a realizar.

2.5.3.1 ANÁLISIS DE ECUACIONES

Costo unitario por periodo.

El costo unitario por periodo simplemente es el costo de la cantidad óptima a pedir.

$$C_1 Q$$

Costo de ordenar una compra.

Puesto que solo se realiza una compra en un periodo el costo de ordenar una compra esta definido por:

$$C_2$$

El inventario promedio por periodo es [Q / 2]. Por consiguiente el costo de mantenimiento del inventario por periodo es:

$$\frac{Q}{2} C_3$$

Para determinar el costo en un periodo se cuenta con la siguiente ecuación:

$$\text{Costo (Q*)} = C_1Q + C_2 + C_3t \frac{Q}{2}$$

El tiempo de un periodo se expresa de la siguiente manera:

$$t = \frac{Q}{D}$$

Nota: La demanda del artículo en un periodo de planeación se define con la letra

D.

El número de periodos se expresa de la manera siguiente:

$$N = \frac{D}{Q}$$

Si se desea determinar el costo total en el periodo de planeación (T) se multiplica, el costo de un periodo por el número de interperiodos (t) que contenga el periodo de planeación. Para determinar este costo se aplica la siguiente ecuación:

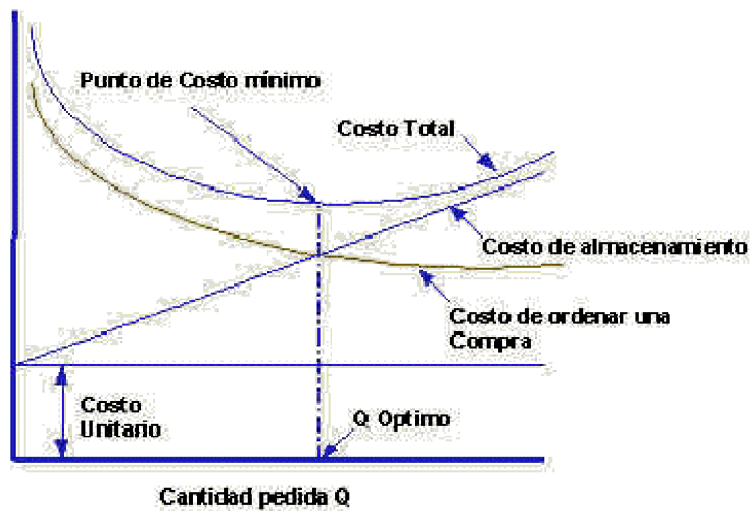
$$\text{Costo Total} = \text{Costo (Q*)}t$$

Otra manera de representar el costo total para el periodo de planeación es por medio de la siguiente ecuación:

$$\text{Costo Total} = C_1D + C_2 \frac{D}{Q} + \frac{Q}{2} C_3$$

Cuando los componentes del costo total se representan gráficamente se obtiene un punto óptimo (de costo mínimo).

Figura 2.11 Componentes de Costo Total



Fuente: Universidad del Centro, Facultad de Ciencias Exactas

Una forma de determinar la cantidad óptima a pedir es suponer diversos valores de Q y sustituir en la ecuación anterior hasta encontrar el punto de costo mínimo. Un procedimiento más sencillo consiste en derivar la ecuación del costo total con respecto a Q e igualar la derivada a cero.

$$\frac{dC}{dQ} = C_1D + C_2 \frac{D}{Q} + \frac{Q}{2} C_3$$

Al resolver esta derivada tenemos la ecuación para determinar la cantidad óptima a pedir.

$$Q = \sqrt{\frac{2C_2D}{C_3}}$$

Esta ecuación ocasiona un costo mínimo y tiene como base un balance entre los dos costos variables (costo de almacenamiento y costo de compra) incluidos en el modelo. Cualquier otra cantidad pedida ocasiona un costo mayor.

2.6 MÉTRICAS DE CALIDAD

Los desarrolladores de software más hastiados estarán de acuerdo que el software de alta calidad es una de las metas más importantes. La calidad del

software es una compleja mezcla de factores que varían a través de diferentes aplicaciones y según el cliente que las pida [Press,1998].

La calidad del software se define también, en términos de ausencia de errores en el funcionamiento del sistema. El ajuste a las necesidades del usuario, el sistema debe ser flexible u susceptible a modificaciones que se puedan realizar de manera rápida y oportuna. El sistema debe alcanzar un desempeño apropiado en términos de tiempo, volumen y espacio. Un sistema debe cumplir de la mejor forma los estándares internacionales establecidos, en lo que a la calidad de software se refiere.

2.6.1 FUNCIONALIDAD

El grado en que el software satisface las necesidades. Los puntos de función (PF) se describen como medidas básicas desde donde se calculan métricas de productividad. Los datos de PF se utilizan de dos formas durante la estimación del proyecto software:

- *Como una variable de estimación que se utiliza para dimensionar cada elemento del software.*

- *Como métricas de línea base recopilada de proyectos anteriores, y utilizados junto con variables de estimación para desarrollar proyecciones de costo y esfuerzo.*

Para estimaciones de PF, la descomposición funciona de la siguiente manera (ver Tabla 2.6):

Tabla 2.5 Dominios de información de Punto Función.

| Dominio de Información | Descripción |
|--------------------------------------|--|
| Número de entradas De usuario | Una petición se define como una entrada interactiva que produce la generación de alguna respuesta del software inmediata en forma de salida interactiva. Se cuenta cada petición por separado. Dentro de un informe no se cuentan de forma separada. |
| Número de archivos | Se cuenta cada archivo maestro lógico (esto es, un grupo lógico de datos que puede ser una parte de una gran base de datos o un archivo independiente). |
| Número de interfaces externas | Se cuenta todas las interfaces legibles por la máquina (por ejemplo: archivos de datos de disco) que se utilizan para transmitir información a otros sistemas. |
| Número de salidas de usuario | Se cuenta cada salida que proporciona al usuario información orientada a la aplicación. En este contexto la salida se refiere a informes, pantallas, mensajes de error, y demás. Los elementos de datos particulares |

Fuente: [Press, 1998]

Los puntos de función se calculan completando la Tabla 2.7

Tabla 2.6 Cálculos de los Punto Función.

| Parámetros de medición | Cuenta | Factor de Ponderación | | | Resultado |
|----------------------------------|--------|-----------------------|-------|----------|-------------|
| | | Simple | Medio | Complejo | |
| Número de entradas de usuario | N1 | 3 | 4 | 6 | N1 *factor |
| Número de salidas de usuario. | N1 | 4 | 5 | 7 | N2 *factor |
| Número de peticiones de usuario. | N1 | 3 | 4 | 6 | N3 *factor |
| Número de archivos. | N1 | 7 | 10 | 15 | N4 *factor |
| Número de interfaces externas. | N1 | 5 | 7 | 10 | N5 *factor |
| CUENTA TOTAL | | | | | (Ni*factor) |

Fuente: [Press, 1998]

Para calcular los PF, se utiliza la relación siguiente:

$$PF = CUENTA\ TOTAL * (0.65 + 0.01 * (Fi))$$

CUENTA TOTAL = Sumatoria de todas las entradas de la Tabla .2.7.

Fi = Son valores de ajuste a la complejidad según las respuestas a las preguntas siguientes.

1. ¿Requiere el sistema copias de seguridad y recuperación flexible?
2. ¿Se requiere comunicación de datos?
3. ¿Existen funciones del procedimiento distribuido?
4. ¿Es crítico el rendimiento?
5. ¿Se ejecutara el sistema en un entorno operativo existente y fuertemente utilizado?
6. ¿Requiere el sistema entrada de datos interactiva?
7. ¿Requiere la entrada de datos interactiva que las transiciones de entrada se lleven a cabo sobre múltiples pantallas u operaciones?
8. ¿Se actualiza los archivos maestros de forma interactiva?
9. ¿Son complejas las entradas, las salidas, los archivos y las peticiones?
10. ¿Es complejo el procesamiento interno?
11. ¿Se ha diseñado el código para ser reutilizable?
12. ¿Están concluidas en el diseño la conversión y la instalación?
13. ¿Se ha desarrollado el sistema para soportar múltiples instalaciones en diferentes organizaciones?
14. ¿Se ha diseñado la aplicación para facilitar los cambios y para ser fácilmente utilizado por el usuario?

Cada una de las preguntas, es respondida usando una escala con rangos desde 0 (no importante), hasta 5 (absolutamente esencial).

2.6.2 CONFIABILIDAD

Cantidad de tiempo que el software está disponible para su uso. La confiabilidad está definida como la probabilidad del sistema libre de fallos en un contexto determinado y durante un periodo de tiempo [Plati,2003].

La confiabilidad se expresa como una probabilidad, esta se encuentra en una escala de 0 a 1.

Si la probabilidad se acerca a uno, mas confiabilidad brindar á el sistema. Para el usuario, el sistema será mas confiable mientras presenta menos errores.

La confiabilidad de un sistema se calcula mediante la siguiente función:

Probabilidad de hallar una falla: $P(T \leq t) = F(t)$

Probabilidad de no hallar una falla: $P(T > t) = 1 - F(t)$

Con $F(t) = F_c * (e^{-\lambda * t})$

Donde :

$F_c = 0.87$: Funcionalidad del sistema.

$\lambda = 1$: Tasa de fallos en 7 ejecuciones dentro de un mes.

| Hallando confiabilidad | Prob. Hallar una falla(F(t)) | Prob. de no hallar una falla |
|-----------------------------------|------------------------------|------------------------------|
| $F(t) = F_c * (e^{-\lambda * t})$ | $P(T \leq t) = F(t)$ | $1 - F(t)$ |

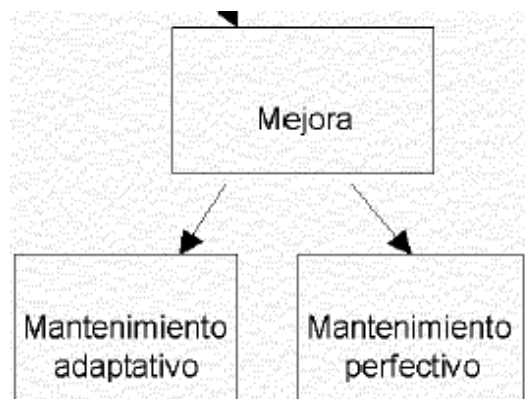
2.6.3 PORTABILIDAD

La portabilidad de un sistema de información, se define como la factibilidad de transferir un producto a diferentes entornos de hardware/software, sin necesidad de aplicar acciones o mecanismos distintos. También es considerado como la capacidad del producto software para ser usado en lugar de otro producto software, para el mismo propósito dentro del mismo entorno. Las características más importantes que se consideran para este factor son: la facilidad de instalación, facilidad de ajuste y facilidad de adaptación al cambio [Plati,2003].

2.6.4 MANTENIBILIDAD

Dentro de los tipos de mantenimiento para la mejora, se muestra dos: el mantenimiento adaptativo y el mantenimiento perfectivo [Juan C,1995], como se muestra en la figura 2.12

Figura 2.12 Clasificación de tipos de mantenimiento



Fuente: [Juan C, 1995]

- **Adaptativo.** *Modificación de un producto software, después de su entrega, para conseguir que sea utilizable en un nuevo entorno.*
- **Perfectivo.** *Modificación de un producto software después de su entrega, para mejorar su rendimiento o su mantenibilidad.*

2.7 TECNOLOGÍAS

2.7.1 SEGURIDAD DEL SISTEMA

Intranet es una infraestructura basada en los estándares y tecnologías de Internet que soporta el compartir información dentro de un grupo bien definido y limitado [Geoci, 2000].

Aunque una Intranet sea una red privada en la que se tengan grupos bien definidos y limitados ésta no se encuentra exenta de ataques que pudiesen poner en riesgo la información que maneja, ya que la mayoría de éstos son provocados por sus mismos usuarios.

La mayoría de las estadísticas de seguridad en cómputo indican que cerca del 80% de los fraudes relacionados con las computadoras provienen de los usuarios internos, por esto las intranets son las más vulnerables a ataques de ésta índole.

Por tal es importante establecer normas y políticas de seguridad a fin de tener un sistema confiable, y seguro. Un punto muy importante dentro de las políticas es el que tienen que ir acompañadas de sanciones, las cuales deberán también ser redactadas, revisadas, autorizadas, aplicadas y actualizadas [Geoci,2000] .

- **Políticas de contraseñas:** Son una de las políticas más importantes, ya que por lo general, las contraseñas constituyen la primera y tal vez única manera de autenticación y, por tanto, la única línea de defensa contra ataques. Éstas establecen quién asignará la contraseña, qué longitud debe tener, a qué formato deberá apegarse, cómo será comunicada.

En PHP se utiliza la función MD5 (Message Digest 5), que es una función hash irreversible (de un sólo sentido), es decir, inscripta el password tecleado por el usuario y es imposible que partiendo desde la cadena encriptada se vuelva a la contraseña origen. Por esto mismo no hay problema de que alguien pueda acceder al campo encriptado de la base de datos.

- **Políticas de uso adecuado:** Especifican lo que se considera un uso adecuado o inadecuado del sistema por parte de los usuarios, así como lo que está permitido y lo que está prohibido dentro del sistema de cómputo.
- **Políticas de respaldos:** Especifican qué información debe respaldarse, con qué periodicidad, qué medios de respaldo utilizar, cómo deberá ser restaurada la información, dónde deberán almacenarse los respaldos.

2.7.1.1 ¿QUE ES PHP?

Figura 2.13 Funcionalidad del PHP



Fuente: Ing. Joel Gonzáles Estrada, Desarrollo Web con PHP y MySQL

El lenguaje PHP es un lenguaje de programación de estilo clásico, es decir, es un lenguaje de programación con variables, sentencias condicionales, ciclos (bucles), funciones.... No es un lenguaje de marcado como podría ser HTML, XML o WML. Está más cercano a Java Script o a C, para aquellos que conocen estos lenguajes.

Recursos que tenga el servidor como por ejemplo podría ser una base de datos. El programa PHP es ejecutado en el servidor y el resultado enviado al navegador. El resultado es normalmente una página HTML pero igualmente podría ser una pagina WML.

Al ser PHP un lenguaje que se ejecuta en el servidor no es necesario que su navegador lo soporte, es independiente del browser, pero sin embargo para que las páginas PHP funcionen, el servidor donde están alojadas debe soportar PHP.

2.7.1.2 CARACTERÍSTICAS DE PHP

- *Es un lenguaje multiplataforma.*
- *Capacidad de conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL.*
- *Capacidad de expandir su potencial utilizando la enorme cantidad de módulos (llamados ext's o extensiones).*
- *Posee una amplia documentación, entre la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda.*

- *Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.*
- *Permite las técnicas de Programación Orientada a Objetos .*
- *Biblioteca nativa de funciones sumamente amplia e incluida.*
- *No requiere definición de tipos de variables.*
- *Tiene manejo de excepciones (desde PHP5).*

2.7.2 CONCEPTOS BÁSICOS ASOCIADOS A REDES

2.7.2.1 CLASIFICACIÓN DE LAS REDES

Clasificación por área de cobertura

El universo de las redes, puede clasificarse según la extensión que abarcan.

Cada uno de los tipos requiere de tecnologías y topologías específicas. Se distinguen en general 3 categorías: [Mariano 03]

- *REDES LAN o Local Área Networks: Son las que no exceden 1 Km. de extensión. Lo más habitual es que abarquen un edificio o varios dentro de una manzana o un área limitada*
- *REDES MAN o Metropolitan Área Network: Hasta 10 Km., es decir, distintos puntos dentro de una misma ciudad.*

- *REDES WAN o Wide Área Networks: Más de 10 Km. Distintas ciudades dentro de un mismo país o distintos países.*

2.7.2.2 TOPOLOGÍAS DE REDES

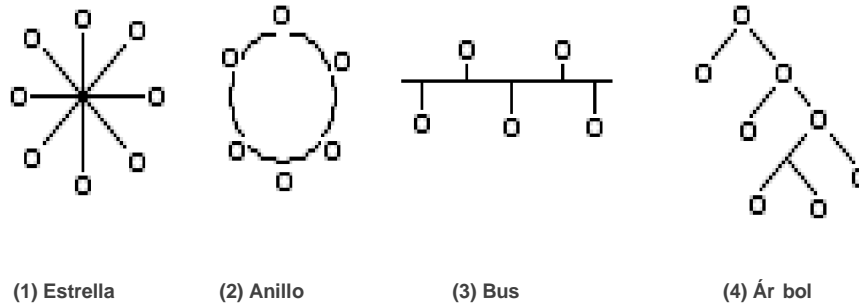
Topologías físicas

La topología física es la conexión real del cableado entre los dispositivos.

Hay una gran variedad, nosotros enumeraremos las 4 principales. [Mariano 03]

- **ESTRELLA:** *Las terminales se conectan todas directamente a un dispositivo central*
- **ANILLO:** *El cable de interconexión recorre uno a uno las terminales cerrándose en un lazo*
- **BUS:** *Un único cable recorre todas las terminales desde un extremo a otro. También se la conoce como "**topología horizontal**"*
- **ÁRBOL:** *Partiendo de un dispositivo central los equipos se van ramificando. También se la conoce como "**estructura jerárquica**".*

Figura 2.14 Topologías Físicas



(1) Estrella

(2) Anillo

(3) Bus

(4) Árbol

Fuente: Mariano López Figuerola

2.7.2.3 COMPONENTES DE HARDWARE DE UNA RED

PLACAS DE RED o NIC's: proporcionan la interfaz entre las PCs o terminales y el medio físico. [Mariano 03]

- **REPETIDORES:** son elementos activos que se utilizan como "refuerzo" de la señal. Permiten incorporar nuevos segmentos de cableado.
- **CONCENTRADORES O HUBS:** se utilizan como punto de partida del cableado UTP. De allí salen los cables a cada una de los terminales. Su funcionamiento se basa en "repetir" la señal que llega por una boca en las demás. Pueden conectarse en cascada constituyendo una estructura tipo "árbol".
- **SWITCHES:** cumplen la misma función que los hubs pero poseen una cierta inteligencia que los hace más eficientes. En vez de repetir la señal a todas las bocas sólo la envía a la salida correspondiente. Esto permite reducir el tráfico en la red.
- **BRIDGES:** interconectan 2 redes iguales.

- **ROUTERS:** encaminan la información hacia otras redes. Son la piedra fundamental de Internet.

- **GATEWAYS:** igual que los routers pero permite n conectar redes de diferentes tipos.

2.8 PRESUPUESTO

2.8.1 ANÁLISIS DE COSTOS

En la siguiente tabla se mostrará el monto aproximado de cuanto se tendría que gastar en la realización del sistema, entonces el total reflejaría el costo del sistema en general.

Tabla 2.7 Costos de la clínica con respecto al sistema a realizar.

| | MONTO \$us | TOTAL |
|--|-------------------|---------------|
| Costo de Construir el Sistema | | |
| Analista de Sistemas | 80 | 80 |
| Costo de Implementar el Sistema | | |
| Capacitación | 50 | |
| Realización de Base de Datos | 50 | |
| Gastos de Equipo | | |
| Gastos de Instalación | 300 | |
| | | 400 |
| Costos Operacionales | | |
| Hardware y Software | 900 | |
| Mantenimiento | 50 | |
| | | 950 |
| Total de Costos | | \$1430 |

Fuente: Elaboración Propia

CAPÍTULO III
MARCO APLICATIVO

MARCO APLICATIVO

3.1 INTRODUCCIÓN

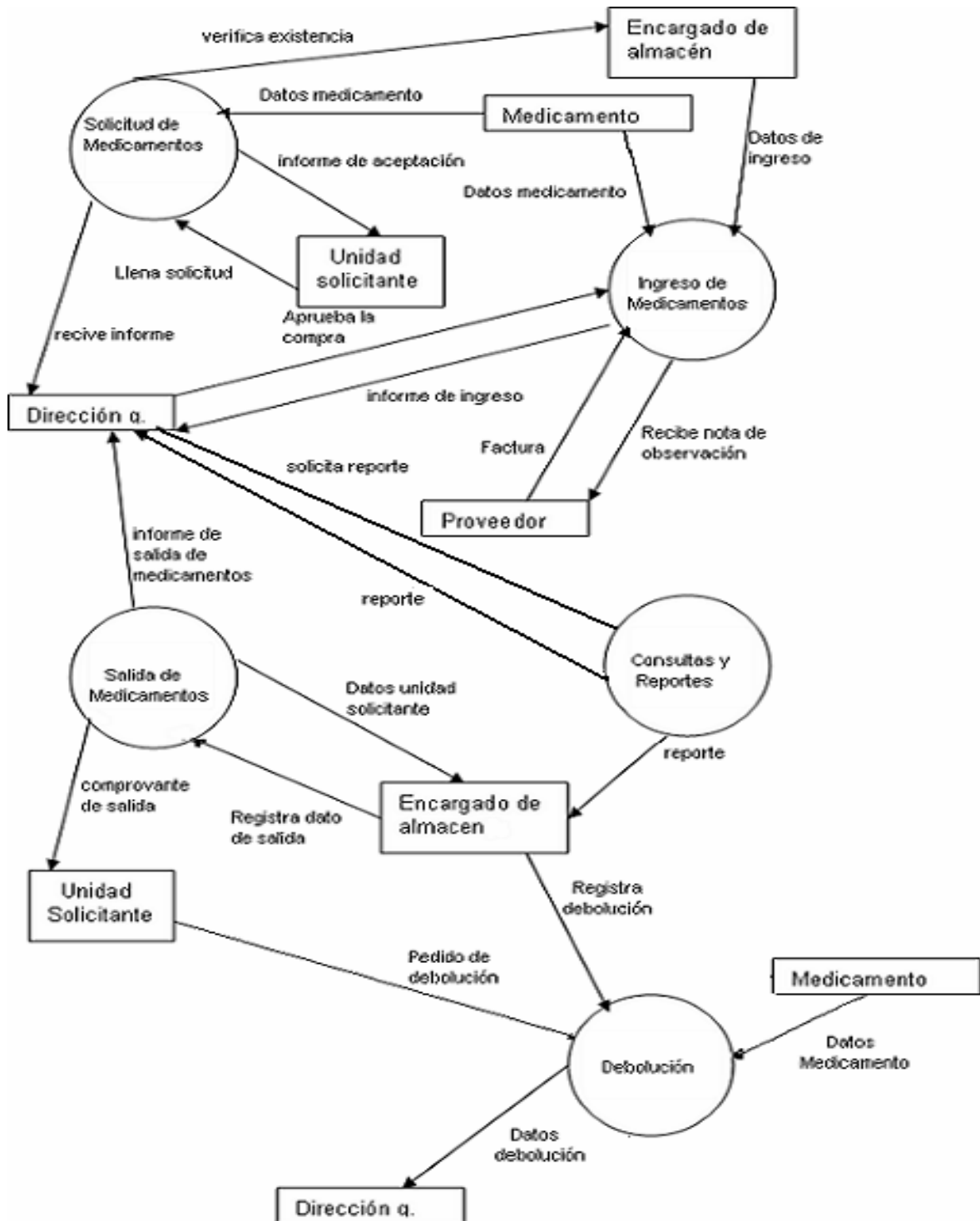
El análisis del " Sistema de Control y Seguimiento de Inventario de Fármacos ", hace uso de los instrumentos métodos y técnicas descritas en el capítulo anterior "Marco teórico" brindando a la Clínica el soporte administrativo para la toma de decisiones que se aduce a sus necesidades tales como el control en el inventario de medicamentos.

El análisis sigue los lineamientos del Proceso Unificado de Desarrollo de Software (RUP), utilizando el Lenguaje de Modelado Unificado (UML) representándose todos los esquemas de un sistemas de software de acuerdo los modelos descritos por RUP.

3.2 ANÁLISIS DEL SISTEMA ACTUAL

El área de inventarios de la clínica se encarga de registrar, clasificar todos los movimientos de almacén. Además de controlar las adquisiciones y venta de los mismos. Estos procesos se realizan manualmente, lo que representa un esfuerzo por parte de los encargados de esta área por que cada vez se tarda más en la ubicación de un medicamento además de sus características básicas, si contaran con un sistema de información, se realizaría sus tareas en menor tiempo

Figura 3.1: diagrama de flujo de datos



Fuente: Elaboración Propia

3.2 FASE DE PLANEACIÓN

3.2.1 DESCRIPCIÓN DE LOS ACTORES

La identificación de actores en términos generales son usuarios del sistema los cuales interactúan, aportan y reciben información del sistema para coadyuvar a sus tareas cotidianas o necesidades demandadas.

A continuación se da una lista de los actores o usuarios identificados.

Director General *Es la persona que necesita estar mas informado teniendo un control y seguimiento de las actividades de la Clínica. Sus funciones son:*

- *Solicita reportes de almacén para la toma de decisiones.*
- *Aprueba las solicitudes que llegan a la oficina de la unidad solicitante.*
- *Ordena la ejecución de transacciones económicas.*

Encargado de almacén

- *Realiza el Inventario de medicamentos dentro del Almacén.*
- *Elabora reportes de inventario.*
- *Elabora comprobantes de Salida y entrada de medicamentos.*

- *Elabora la solicitud de compra de medicamentos.*
- *Realiza la compra de medicamentos.*
- *Realiza el registro de medicamentos que entran y salen de Almacén.*
- *Elabora una lista detallada de los medicamentos.*

Unidad Solicitante

- *Realiza pedidos de medicamentos para su abastecimiento. Mediante un formulario de Solicitud.*

Proveedor

- *Encargadas de proveer medicamentos a Almacén de la Clínica.*
- *Emite facturas o notas de venta.*

3.2.2 IDENTIFICACIÓN DE LOS CASOS DE USO

El sistema de control y seguimiento de inventario de fármacos está constituido por los siguientes casos de uso que son:

Tabla 3.1 Identificación de casos de uso

| ACTOR | CASOS DE USO |
|-----------------------------|--|
| ENCARGADO DE ALMACEN | <ul style="list-style-type: none"> ➤ Registro de ingreso de medicamentos. ➤ Registro de Solicitud de Medicamentos. ➤ Registro de Salidas de medicamentos. ➤ Control de inventario. ➤ Emisión de reportes y consultas. |
| UNIDAD SOLICITANTE | <ul style="list-style-type: none"> ➤ Registro de ingreso de medicamentos. ➤ Registro de solicitud de medicamentos. ➤ Registro de Salida de medicamentos. |
| DIRECCION | <ul style="list-style-type: none"> ➤ Registro de ingreso de medicamentos. ➤ Registro de solicitud de medicamentos. ➤ Registro de salida de medicamentos. ➤ Control de inventario. ➤ Emisión de reportes y consultas. |
| PROVEEDORES | <ul style="list-style-type: none"> ➤ Registro de ingreso de medicamentos. |

Fuente: Elaboración Propia.

3.2.3 CATÁLOGO DE REQUERIMIENTOS DEL SISTEMA

Un proyecto no puede ser exitoso sin una especificación correcta y exhaustiva de los requerimientos, donde describe las necesidades o deseos de un producto.

- *Registro de las solicitudes, entrantes y salientes de medicamentos a Almacén.*

- *Verificación rápida de la existencia de medicamentos.*

- *Realizar el seguimiento y control de la compra de medicamentos.*
- *Conocer cuales son los proveedores y clientes de compra y venta de medicamentos.*
- *Realizar reportes de movimiento de inventario.*
- *Realizar comprobantes de ingreso y salida de medicamentos para las Unidades Solicitantes.*

3.2.4 FUNCIONES BASICAS

Las funciones del sistema son lo que esta deberá hacer. Estas funciones o requerimientos del sistema se detallan a continuación asignándoles además la categoría de evidente y oculta.

En las siguientes tablas se reflejan las funciones del sistema, donde la primera columna hace referencia a la cantidad de funciones para una tarea o módulo específico, la segunda columna describe las funciones en si que engloba un módulo, la tercera columna muestra las clasificaciones que pueden tener cada función, y entre ellas están:

- **Evidente:** *Función que debe realizarse, y el usuario debería saber que se ha realizado.*
- **Oculto:** *Debe realizarse, aunque no es visible para los usuarios.*
- **Superflua:** *Opcionales, su inclusión no repercute significativamente en el costo ni en otras funciones.*

En la tabla 3.2 se especifica la funcionalidad que tiene el sistema para el ámbito de registro de solicitud de medicamentos.

Tabla 3.2 Registro de Solicitud de Medicamentos

| Ref: # | FUNCIÓN | CATEGORÍA |
|---------------|--|------------------|
| R1.1 | Se llena solicitud de requerimiento de medicamentos. | Evidente |
| R1.2 | Se verifica la existencia de medicamentos en almacén. | Oculto |
| R1.3 | El sistema registra los medicamentos solicitados a almacén | Evidente |
| R1.4 | Genera reporte de la solicitud de medicamentos. | Oculto |

Fuente: Elaboración Propia

La tabla 3.3 muestra el Registro de ingreso de Medicamentos a almacén.

Tabla 3.3 Registro de salida de Medicamentos

| Ref: # | FUNCIÓN | CATEGORÍA |
|---------------|--|------------------|
| R2.1 | Registra la compra de medicamentos. | Evidente |
| R2.2 | Verifica el estado y cantidad de medicamentos. | Evidente |
| R2.3 | Incrementa las cantidades del inventario cuando realiza una compra | Oculto |
| R2.4 | Genera comprobante de entrada de medicamento para unidad solicitante | Oculto |
| R2.5 | Genera reporte de los ingresos de medicamentos | Oculto |

Fuente: Elaboración Propia

La tabla 3.4 muestra el Registro de salida de Medicamentos.

Tabla 3.4 Registro de salida de Medicamentos

| Ref: # | FUNCIÓN | CATEGORÍA |
|-----------|---|-----------|
| R3.1 | Recibe el número de solicitud de la unidad solicitante que desea la salida medicamentos | Evidente |
| R3.2 | Selecciona medicamentos disponibles. | Evidente |
| R3.3 | El sistema registra los medicamentos que salen de almacén. | Evidente |
| R3.4 | Reduce el stock en inventario. | Oculto |
| R3.5 | El sistema realiza el cálculo de medicamentos para la unidad solicitante. | Evidente |
| R3.6 | El sistema realiza el comprobante de salida de medicamentos para la unidad solicitante. | Oculto |
| R3.7 | Realiza reporte de salida de medicamentos que salieron a Dirección. | Oculto |

Fuente: Elaboración Propia

La tabla 3.5 muestra el Control del inventario.

Tabla 3.5 Control de Inventario

| Ref: # | FUNCIÓN | CATEGORÍA |
|-----------|--|-----------|
| R4.1 | Verifica cantidades existentes por medicamentos. | Evidente |
| R4.2 | Registra faltantes. | Oculto |
| R4.3 | Revisar y verificar kardex de medicamentos. | Evidente |
| R4.5 | Genera reporte detallado. | Oculto |

Fuente: Elaboración Propia

3.3 ANÁLISIS

3.3.1 DISEÑO DE LOS CASOS DE USO

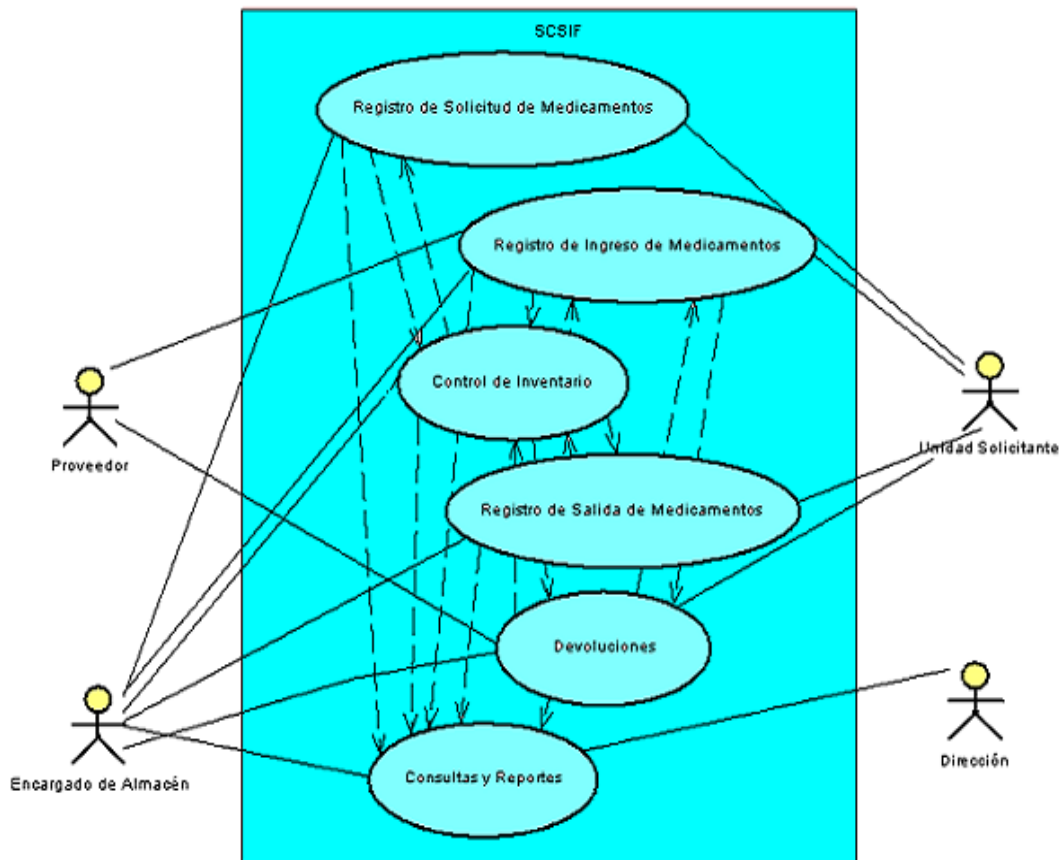
3.3.1.1 DIAGRAMA CASOS DE USO DE ALTO NIVEL

En un caso de uso descrito a alto nivel la descripción es muy general, normalmente se condensa en dos o tres frases. Es útil para comprender el ámbito y el grado de complejidad del sistema [Larma, 1999]

Los casos de uso de alto nivel que a continuación pondré a consideración son concisos y no detalla los procesos sin embargo explican los requerimientos de los actores.

Dentro la estructura de casos de uso de alto nivel se identifica el tipo, este puede ser primario secundario u opcional de acuerdo a su prioridad

Figura 3.2 Diagrama de caso de uso principal



Fuente: Elaboración Propia

3.3.1.2 DESCRIPCIÓN CASOS DE USO DE ALTO NIVEL

A continuación se muestran las tablas referentes a la descripción del caso de uso, cada tabla muestra la descripción por cada caso de uso a fin de entender rápidamente el grado de complejidad y funcionalidad del sistema (ver Tablas 3.6,.....,3.10).

Tabla 3.6 Registro de solicitud de Medicamentos

| | |
|---------------------|---|
| CASO DE USO: | REGISTRO DE SOLICITUD DE MEDICAMENTOS |
| ACTORES: | Encargado de almacén, dirección y unidad solicitante |
| TIPO: | Primario |
| DESCRIPCIÓN: | La unidad solicitante incluyendo el encargado de almacén realiza el registro de la solicitud de medicamentos al almacén, verifica su existencia, luego genera un reporte y detalle de los medicamentos aceptados o rechazados por la dirección general para ser enviados a la unidad solicitante. |

Fuente: Elaboración Propia

Tabla 3.7 Ingreso de Medicamentos

| | |
|---------------------|--|
| CASO DE USO: | REGISTRO DE INGRESO DE MEDICAMENTOS |
| ACTORES: | Encargado de almacén, dirección y unidad solicitante |
| TIPO: | Primario |
| DESCRIPCIÓN: | El encargado de almacenes realiza el registro de entrada de medicamentos al almacén verifica cantidad, estado y características, luego genera un comprobante de entrada de los medicamentos. |

Fuente: elaboración Propia

Tabla 3.8 Registro de salida de Medicamentos

| | |
|---------------------|--|
| CASO DE USO: | REGISTRO DE SALIDA DE MEDICAMENTOS |
| ACTORES: | Encargado de almacén, dirección y unidad solicitante |
| TIPO: | Primario |
| DESCRIPCIÓN: | El encargado de almacén realiza el registro de salida de medicamentos del almacén verifica cantidad, luego se genera un reporte y comprobante de los medicamentos que han salido de almacén para alguna finalidad. |

Fuente: Elaboración Propia

Tabla 3.9 Registro de Control de Inventario

| | |
|---------------------|--|
| CASO DE USO: | CONTROL DE INVENTARIO |
| ACTORES: | Encargado de almacén y Dirección |
| TIPO: | Primario |
| DESCRIPCIÓN: | La dirección solicita a almacén un reporte de saldos de existentes, es así que el encargado de almacén consulta la existencia por Medicamento y procede a la elaboración del reporte que es impreso. |

Fuente: Elaboración Propia

Tabla 3.10 Devolución de Medicamentos

| | |
|---------------------|--|
| CASO DE USO: | REGISTRO DE DEVOLUCION DE MEDICAMENTOS |
| ACTORES: | Encargado de almacén, dirección y unidad solicitante |
| TIPO: | Primario |
| DESCRIPCIÓN: | El encargado de almacenes realiza el registro de devolución de medicamentos al almacén verifica cantidad, estado, luego genera un comprobante de devolución de medicamentos. |

Fuente: elaboración Propia

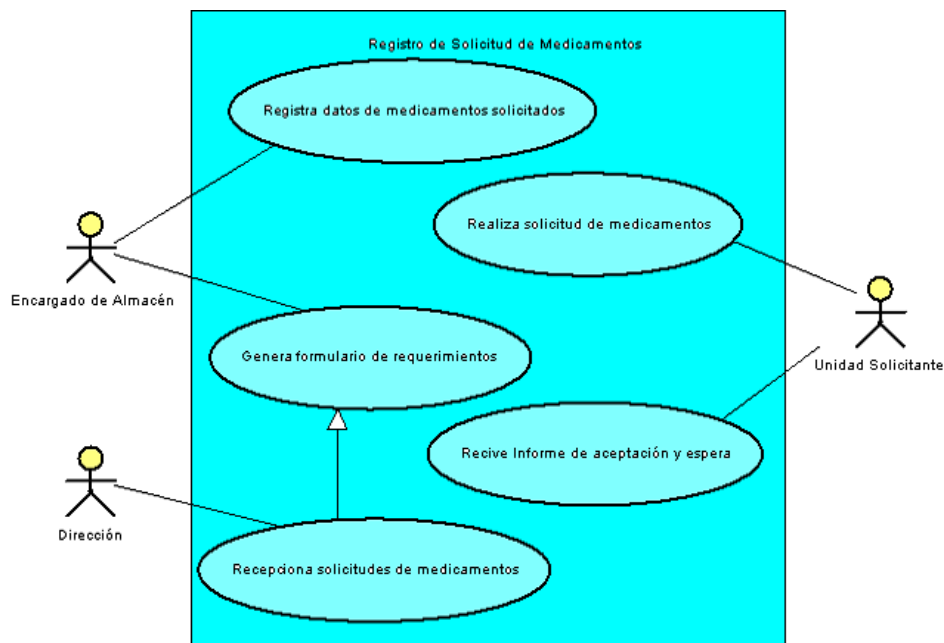
3.3.1.3 CASOS DE USO ESENCIAL O EXPANDIDO

Los casos de uso expandido muestran a detalle los procesos a ntes mencionados, tienen información breve que describe el proceso, el curso normal de los eventos

que detalla la interacción de los actores y el sistema, además de cursos alternativos que pueden presentarse en el curso de los eventos (ver acápite 2.3.2.2).

A continuación se mostrarán los casos de uso expandidos, de los procesos reflejados en el diagrama de casos de alto nivel (ver Figura. 3.3,....., Figura. 3.8), consecutivamente a cada diagrama se muestra una tabla describiendo a detalle en que consiste cada una de ellas (ver Tablas 3.11,....., 3.15).

Figura 3.3 Diagrama de caso de uso de Solicitud de Medicamentos



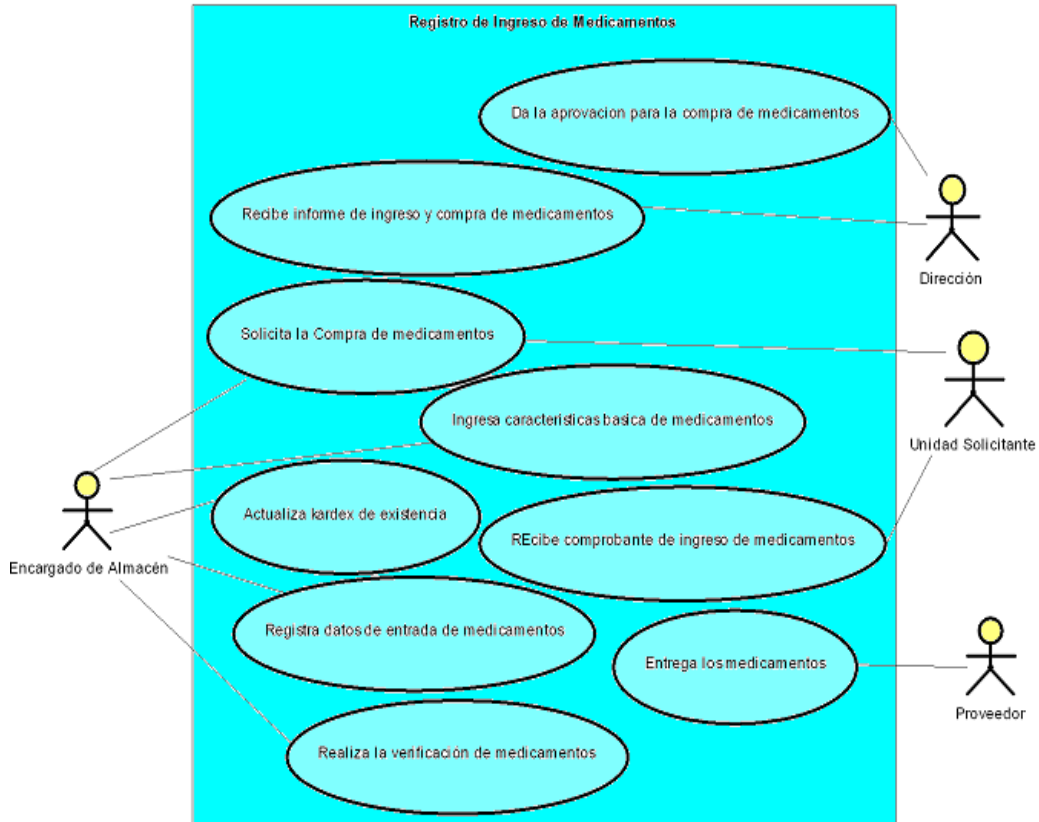
Fuente: Elaboración Propia

Tabla 3.11 Caso de uso Esencial de registro de solicitud de medicamentos

| CASO DE USO | REGISTRO DE SOLICITUD DE MEDICAMENTOS |
|--|---|
| Actores: | Encargado de Almacén, Unidad Solicitante y Dirección |
| Propósito: | Registrar las solicitudes de medicamentos por la unidad solicitantes |
| Resumen: | El encargado de almacén registra las solicitudes de medicamentos que llegan a almacén hechas por alguna unidad solicitante |
| Tipo: | Primario y esencial |
| Referencias Cruzadas: | R1.1, R1.2, R1.3, R1.4, R1.5, R1.6 y R1.7 |
| Curso Normal de Eventos | |
| Acción de Actores | Respuesta del Sistema |
| <p>1.- Este caso de uso comienza cuando la unidad solicitante llena la solicitud de medicamentos y la envía a la Dirección y traspasa a almacén</p> <p>4.- El actor acepta la opción Grabar.</p> | <p>2.- El sistema ya al ingresar colocara la fecha, Nro de solicitud y Nombre de la unidad solicitante por el código ingresado, y además solicita los siguientes datos de solicitud de medicamentos:</p> <ul style="list-style-type: none"> ➤ Medicamento ➤ Nro serie ➤ Código de medicamento ➤ Nro parte ➤ Cantidad solicitada ➤ Finalidad <p>3.- Determina el saldo existente del medicamento y añade la información correspondiente al requerimiento. El sistema muestra las siguientes opciones:</p> <ul style="list-style-type: none"> ➤ Grabar ➤ Cancelar ➤ Repote ➤ Salir <p>5.- Registra solicitud concluida. Genera los comprobantes respectivos de medicamentos aceptados y de espera a unidad solicitante y del proceso de solicitud a la dirección.</p> |

Fuente: Elaboración Propia

Figura 3.4 Diagrama de caso de Registro de Ingreso de medicamentos

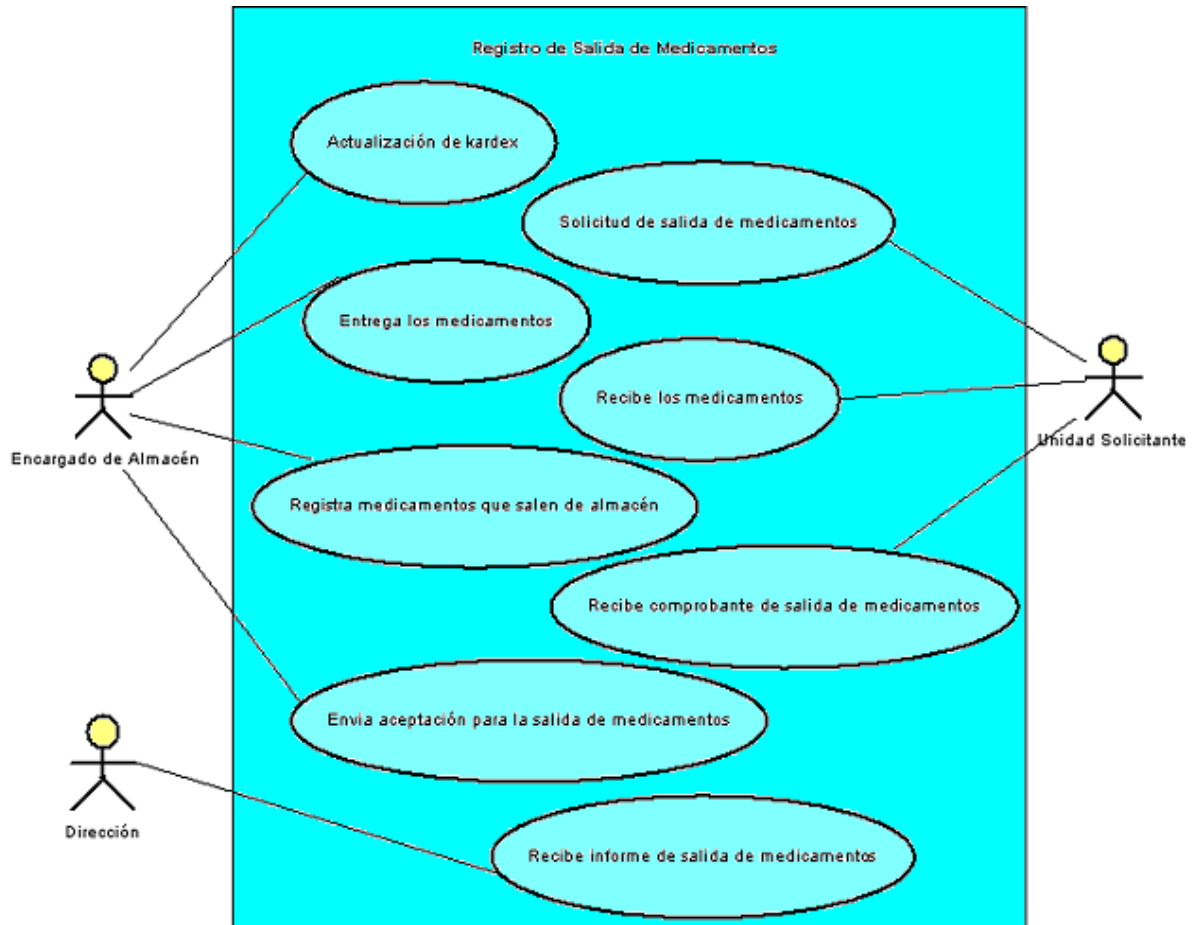


Fuente: Elaboración propia

Tabla 3.12 Caso de uso de registro de ingreso de medicamentos

| CASO DE USO | REGISTRO DE INGRESO DE MEDICAMENTOS |
|------------------------------|---|
| Actores: | Encargado de Almacén, Dirección, Unidad Solicitante y Proveedor. |
| Propósito: | Registrar ingreso de medicamentos a almacén |
| Resumen: | El encargado de almacén registra las entradas de medicamentos que se han comprado para la unidad solicitante. |
| Tipo: | Primario y esencial |
| Referencias Cruzadas: | Funciones R2.1, R2.2, R2.3, R2.4 y R2.5 |
| | |

Figura 3.5 Diagrama de casos de uso de Registro de salida de medicamentos



Fuente: Elaboración Propia

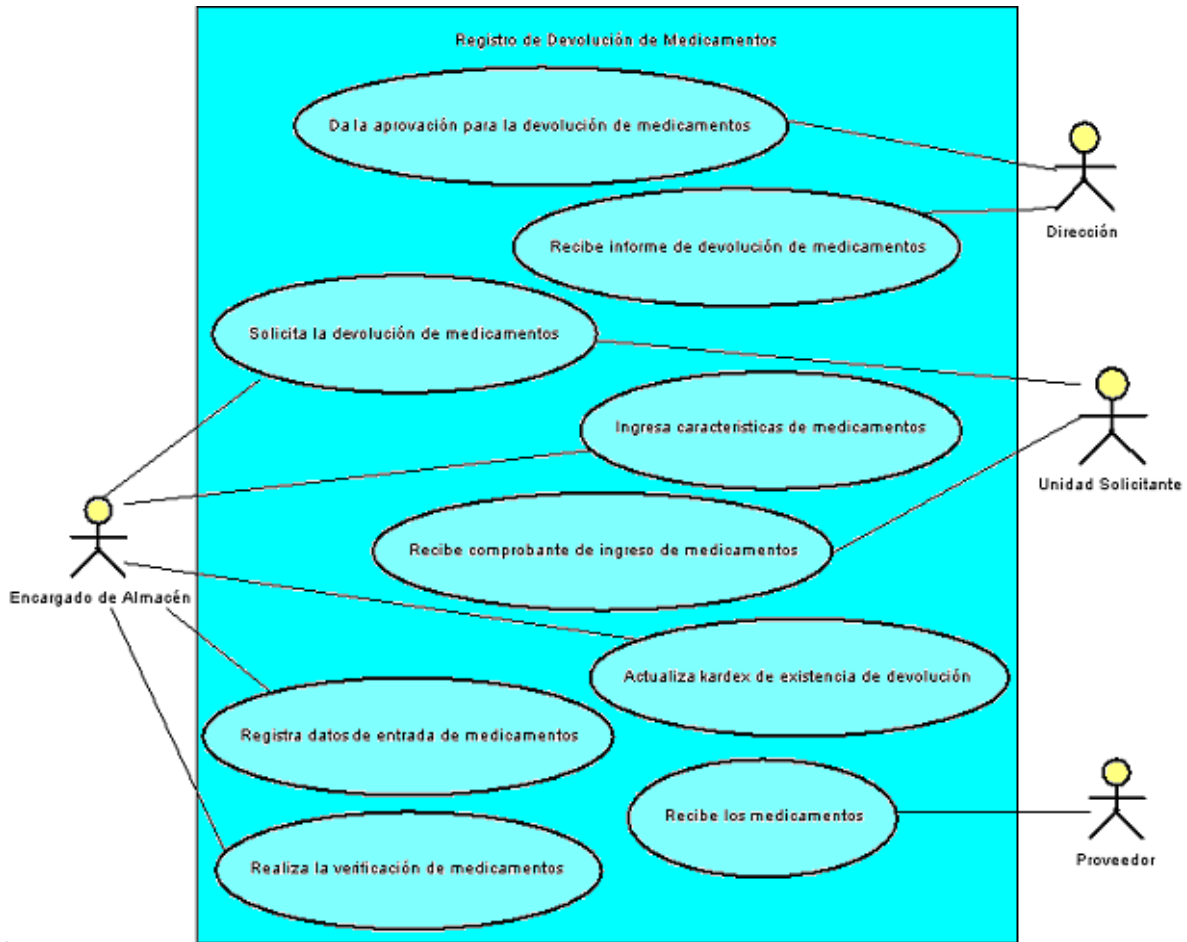
Tabla 3.13 Caso de uso de Registro de Salida de medicamentos

| CASO DE USO | REGISTRO DE SALIDA DE MEDICAMENTOS |
|------------------------------|---|
| Actores: | Encargado de almacén, Dirección y Unidad solicitante |
| Propósito: | Registrar la salida de medicamentos |
| Resumen: | El encargado de almacén registra las salidas de medicamentos por la venta o consumo que realiza la unidad solicitante |
| Tipo: | Primario y esencial |
| Referencias Cruzadas: | R3.1, R3.2, R3.3 y R3.4 |
| | |

| Curso Normal de Eventos | |
|--|---|
| Acción de los actores | Respuesta de sistema |
| <p>1.- Comienza cuando el encargado de almacén registra la salida de medicamentos requeridos y solicitados con anticipación por la unidad solicitante.</p> <p>4.- El encargado de almacén termina de introducir los datos y acepta la opción grabar.</p> <p>7.- El encargado de almacén entrega a la unidad solicitante los medicamentos requeridos.</p> <p>8.- El encargado de almacén envía el comprobante a contabilidad y a la dirección</p> <p>9.- La unidad solicitante se marcha con lo medicamentos.</p> | <p>2.- El sistema al ingresar colocara la fecha y solicita el Nro de solicitud con el cual llenara los siguientes campos.</p> <ul style="list-style-type: none"> ➤ Medicamento ➤ Nro serie ➤ Código de medicamento ➤ Nro parte ➤ Cantidad salida. ➤ Precio compra ➤ Precio venta ➤ Unidad solicitante <p>3.- Determinar la cantidad de salida de los medicamentos e incorpora la información y el sistema muestra las siguientes opciones.</p> <ul style="list-style-type: none"> ➤ Cancelar ➤ Comprobante ➤ Salir <p>5.- Reduce el stock den inventario y calcula el total de medicamentos a entregar.</p> <p>6.- Genera comprobante de salida de medicamentos para la dirección.</p> |

Fuente: Elaboración Propia

Figura 3.6 Diagrama de casos de uso Registro de devolución de medicamentos



Fuente: Elaboración Propia

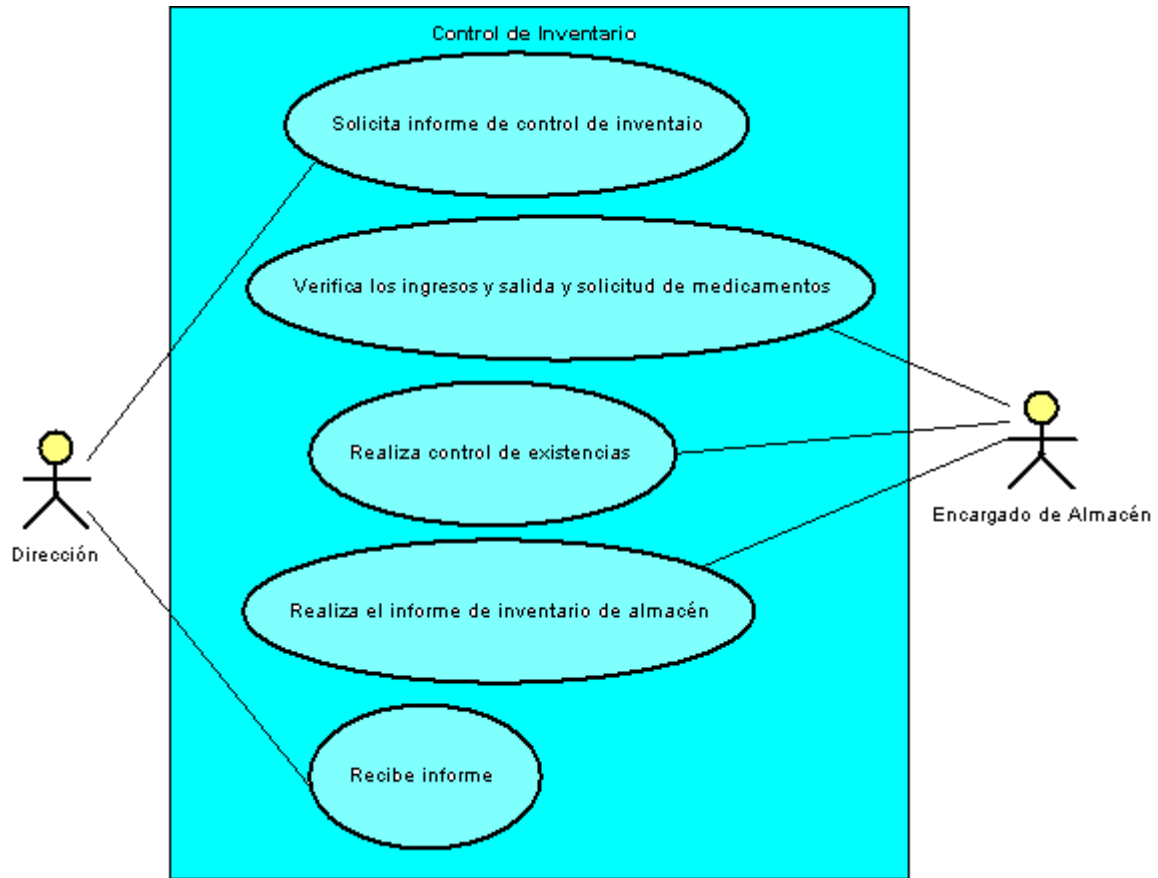
Tabla 3. 14 Caso de uso de Devolución de medicamentos

| CASO DE USO | REGISTRO DE DEVOLUCIÓN DE MEDICAMENTOS |
|------------------------------|--|
| Actores: | Encargado de Almacén, Dirección, Unidad Solicitante y Proveedor. |
| Propósito: | Registrar la devolución de medicamentos a almacén |
| Resumen: | El encargado de almacén registra la devolución de medicamentos de la unidad solicitante. |
| Tipo: | Primario y esencial |
| Referencias Cruzadas: | Funciones R4.1, R4.2, R4.3, R4.4 y R4.5 |
| | |

| Curso Normal de Eventos | |
|--|---|
| Acción de actores | Respuesta de sistema |
| <p>1.- Este caso comienza cuando la unidad solicitante su solicitud de devolución de medicamentos es aceptada.</p> <p>2.- El encargado de almacén empieza a registrar cada medicamento que es devuelto.</p> <p>5.- El encargado de almacén al terminar de introducir los datos correspondientes de cada medicamento acepta la opción grabar.</p> <p>7.- El encargado de almacén le indica al sistema registrar la devolución de medicamento.</p> <p>9.- El encargado de almacén envía el comprobante a contabilidad y el reporte a la dirección.</p> | <p>3.- El sistema al ingresar colocara la fecha y nombre del encargado de almacén que registra el ingreso y además solicita los siguientes datos de entrada de medicamentos.</p> <ul style="list-style-type: none">➤ Nro solicitud <p>Con la cual llenara los siguientes campos</p> <ul style="list-style-type: none">➤ Medicamento➤ Código de medicamento➤ Nro serie➤ Cantidad de ingreso➤ Precio compra➤ Precio venta➤ Nombre proveedor <p>4.- Determina la cantidad que hay del medicamento y agrega la información correspondiente y el sistema muestra las siguientes opciones</p> <ul style="list-style-type: none">➤ Cancelar➤ Comprobante➤ Salir <p>6.- El sistema calcula y presenta el total existente del medicamento</p> <p>8.- Registra la recepción concluida. Genera comprobante de devolución de medicamento.</p> |

Fuente: Elaboración Propia

Figura 3.7 Diagrama de caso de uso de Control de Inventario



Fuente: Elaboración Propia

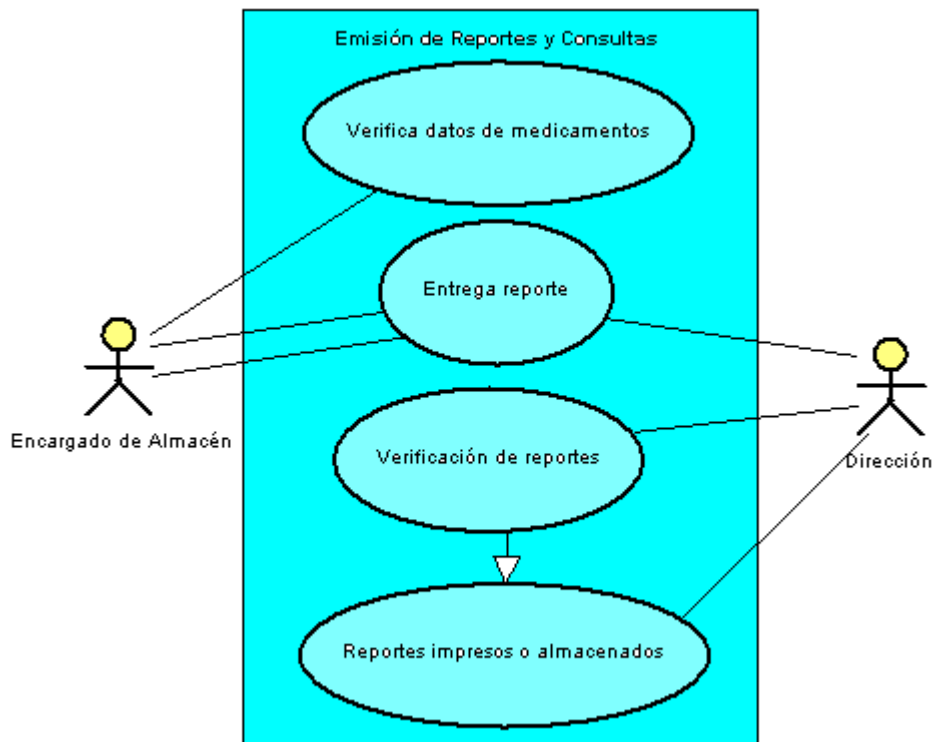
Tabla 3.15 Caso de de Control de inventario

| CASO DE USO | CONTROL DE INVENTARIO |
|------------------------------|---|
| Actores: | Encargado de Almacén y Dirección |
| Propósito: | Realizar el conteo físico y lógico de los medicamentos |
| Resumen: | La dirección solicita un reporte detallado del inventario de los medicamentos en almacén. El encargado de almacén consulta al sistema las cantidades de cada medicamento y realiza una comparación con la existencia física y es impresa y enviada a la dirección |
| Tipo: | Primario y esencial |
| Referencias Cruzadas: | Funciones R5.1, R5.2, R5.3 y R5.4 |
| | |

| Curso Normal de Eventos | |
|--|---|
| Acción de los actores | Respuesta de sistema |
| <p>1.- Este caso de uso comienza cuando la dirección solicita reporte de los medicamentos existentes en almacén.</p> <p>2.- La dirección ingresa al sistema y consulta saldos existentes de los medicamentos realizado por el encargado de almacén.</p> <p>4.- Compara con la existencia física y elabora un reporte.</p> <p>5.- Remite el reporte a la dirección.</p> | <p>3.- Se muestra una lista con la descripción y los saldos de los medicamentos y genera un reporte</p> |

Fuente: Elaboración Propia

Figura 3.8 Diagrama de caso de uso de Emisión de reportes y consultas



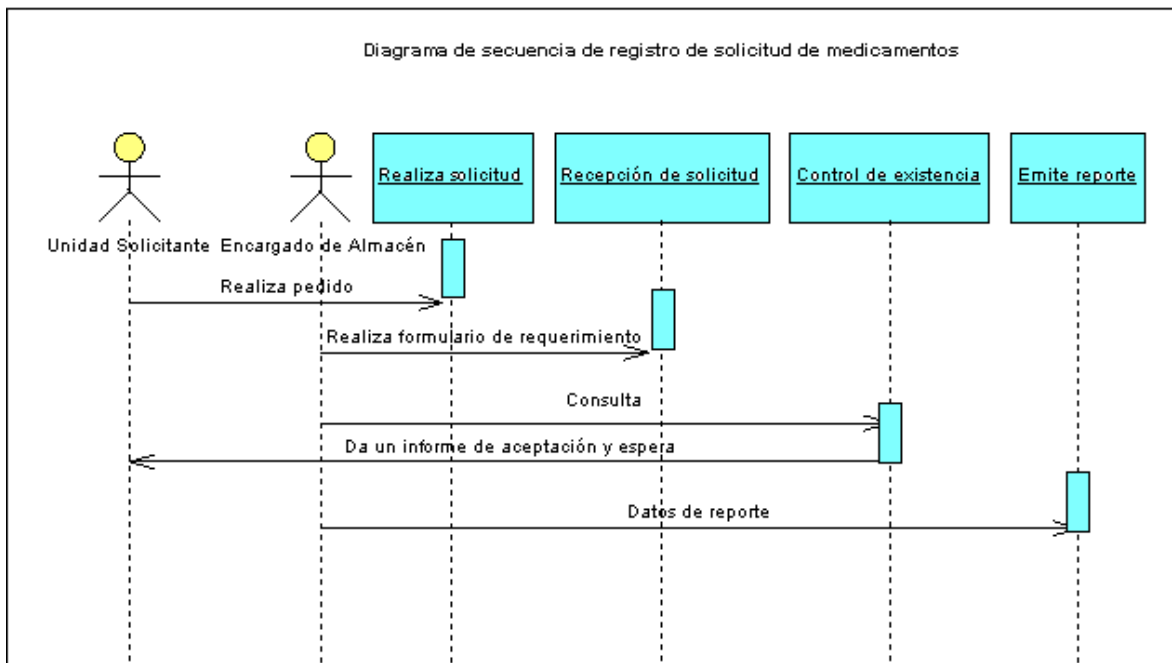
Fuente: Elaboración Propia

3.3.2 DIAGRAMA DE SECUENCIA

El diagrama de secuencia le muestra la forma en que se comunican los objetos al transcurrir el tiempo en el orden de las llamadas/eventos del sistema. El evento del sistema es una entrada externa que origina una operación del sistema como respuesta al evento, representados en secuencias, el detalle del diagrama depende de la fase en la que estemos, lo que pretendamos contar con el diagrama y a quién.

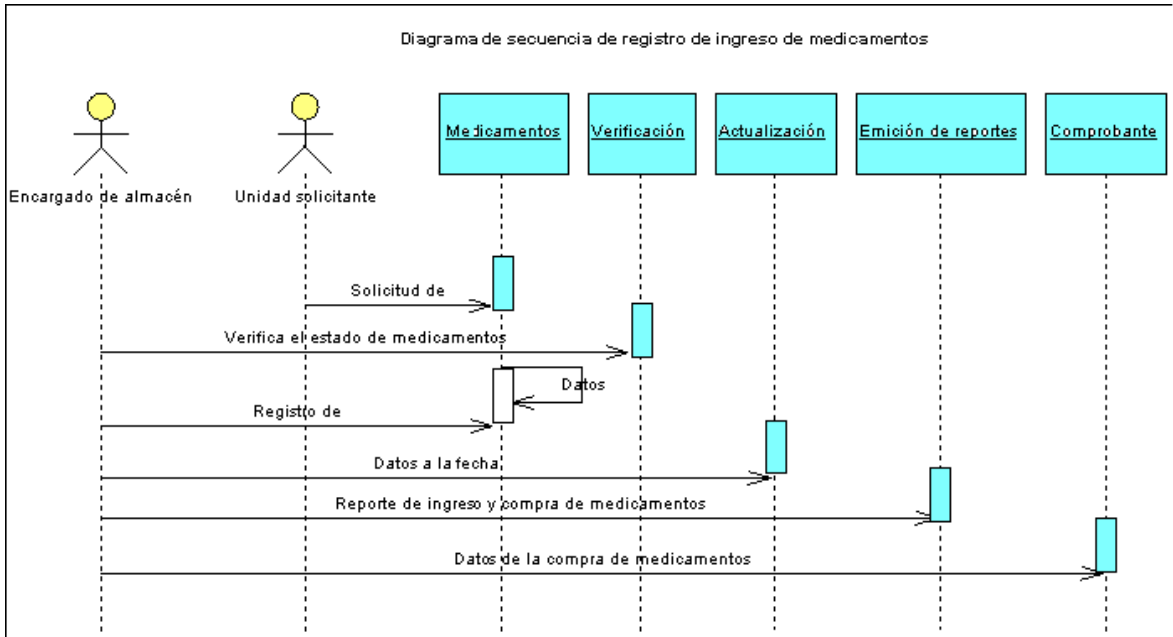
A continuación se muestran los diagramas de secuencia correspondientes al sistema:

Figura 3.9 Diagrama de secuencia de registro de solicitud de medicamentos



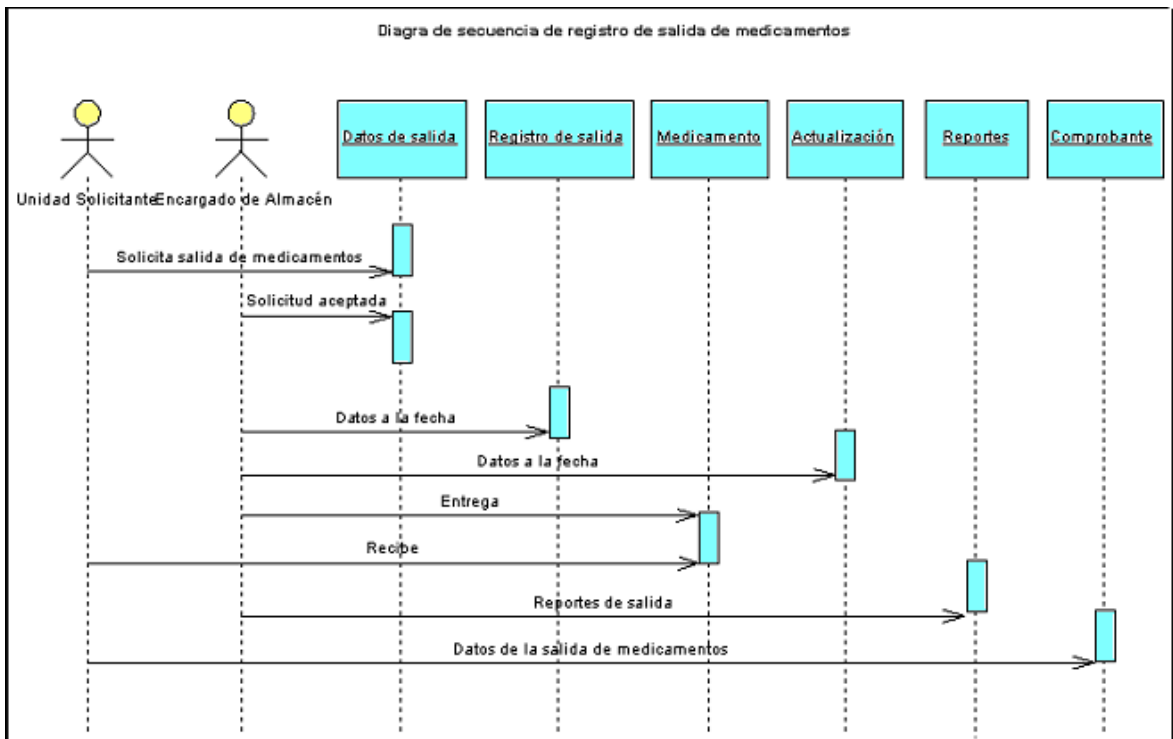
Fuente: Elaboración Propia

Figura 3.10 Diagrama de secuencia de registro de ingreso de medicamentos



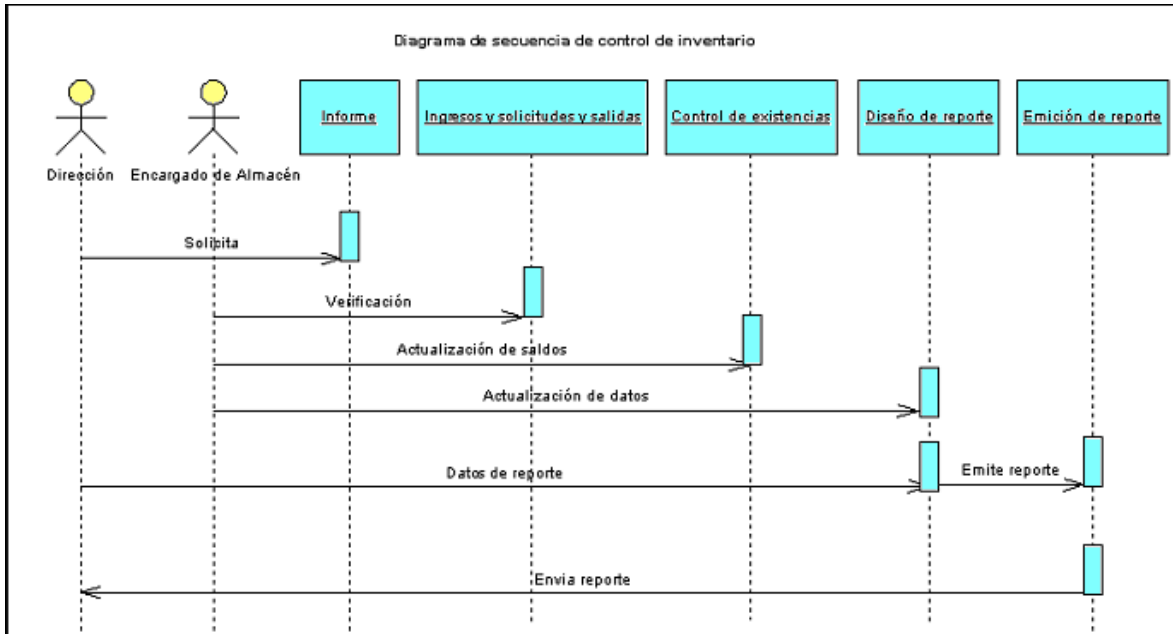
Fuente: Elaboración Propia

Figura 3.11 Diagrama de secuencia de registro de salida de medicamentos



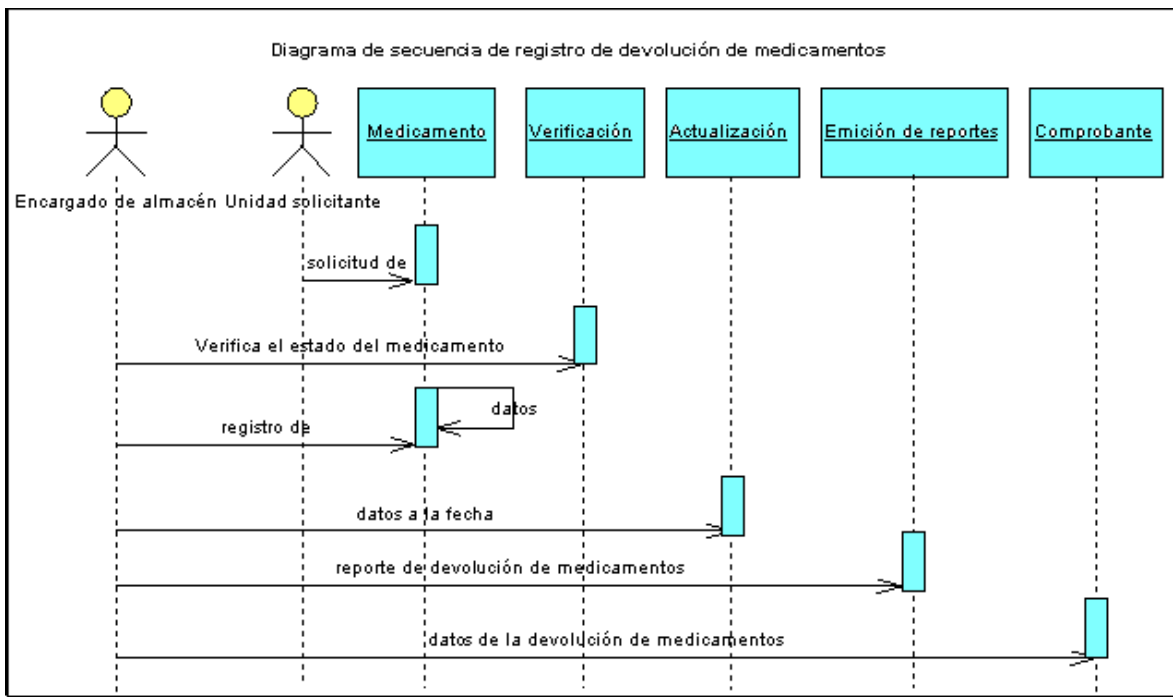
Fuente: Elaboración Propia

Figura 3.12 Diagrama de secuencia de control de inventario



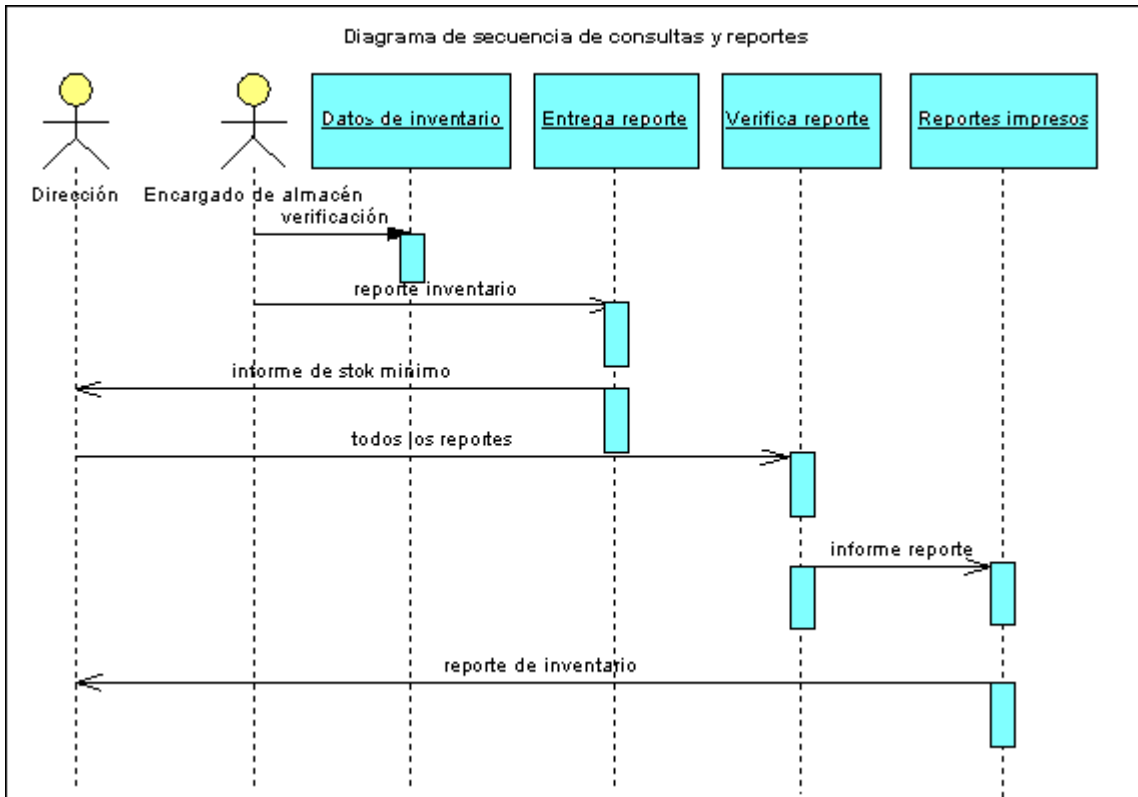
Fuente: Elaboración Propia

Figura 3.13 Diagrama de secuencia de registro de devolución de medicamentos



Fuente: Elaboración Propia

Figura 3.14 Diagrama de secuencia de consulta y reportes



Fuente: Elaboración Propia

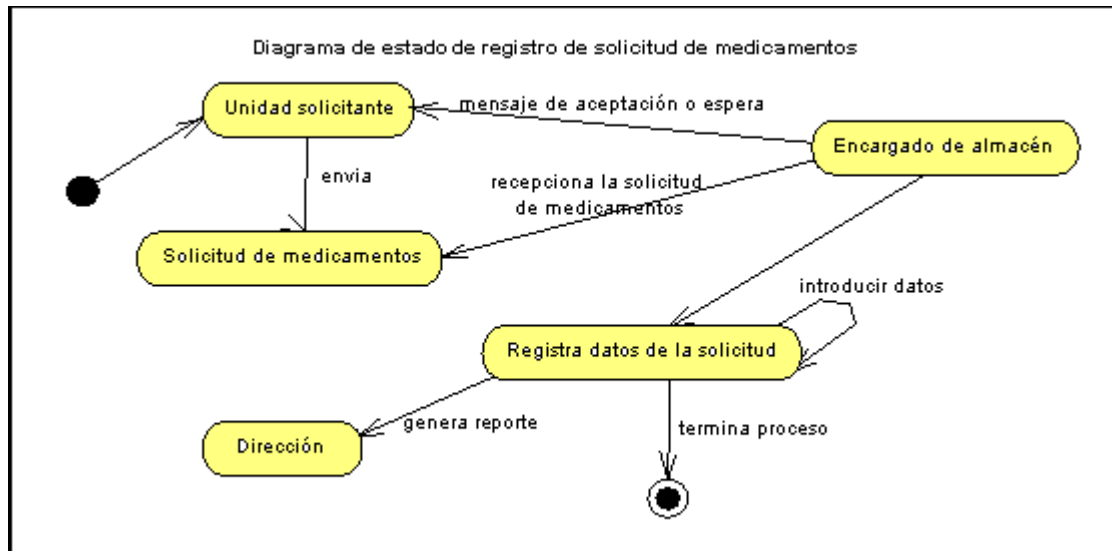
3.3.3 DIAGRAMA DE ESTADOS

Describe visualmente los estados y eventos mas interesados de un objeto, así como su comportamiento ante un evento.

Un diagrama de estado presenta el ciclo de vida de un objeto: los eventos que le ocurren, sus transiciones y los estados que media n entre sus eventos.

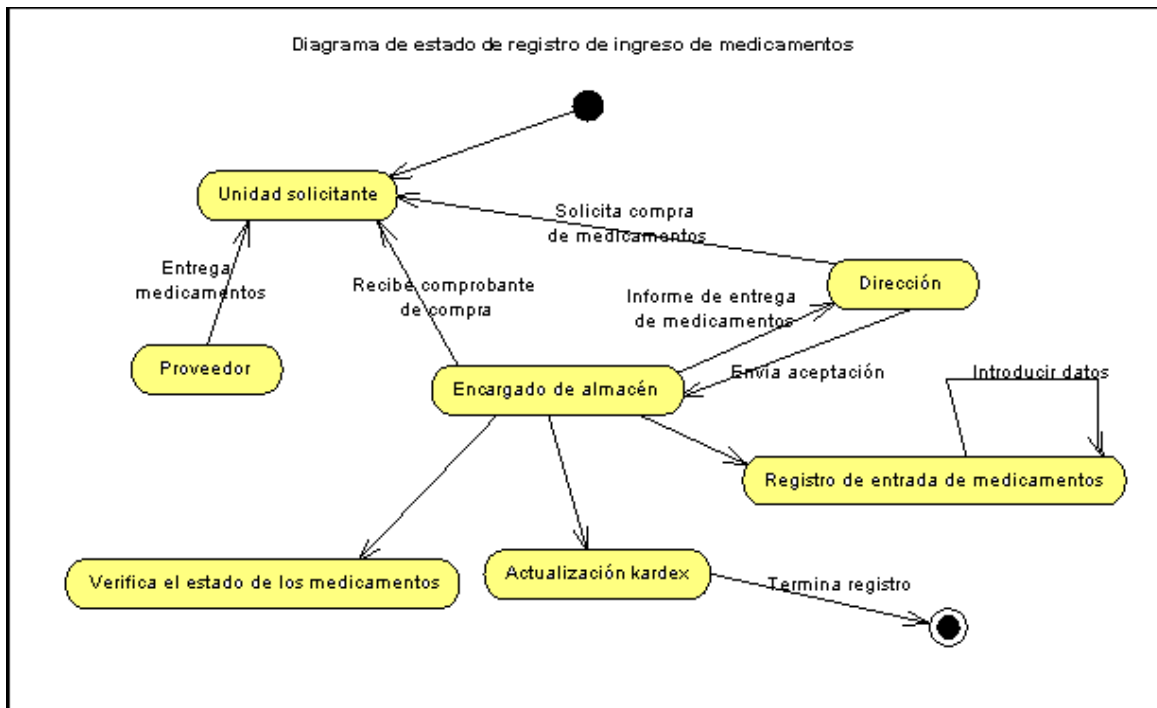
Los diagramas de estados correspondientes a los casos de uso son los siguientes:

Figura 3.15 Diagrama de estado de registro de solicitud de medicamentos



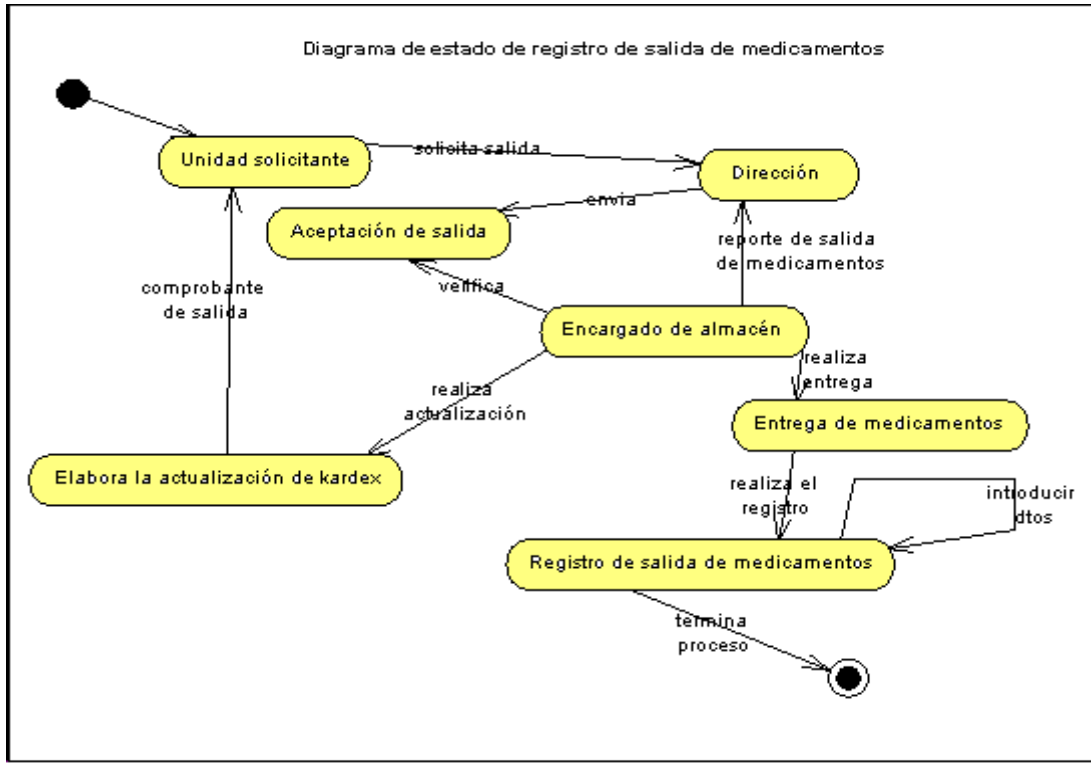
Fuente: Elaboración Propia

Figura 3.16 Diagrama de estado de registro de ingreso de medicamentos



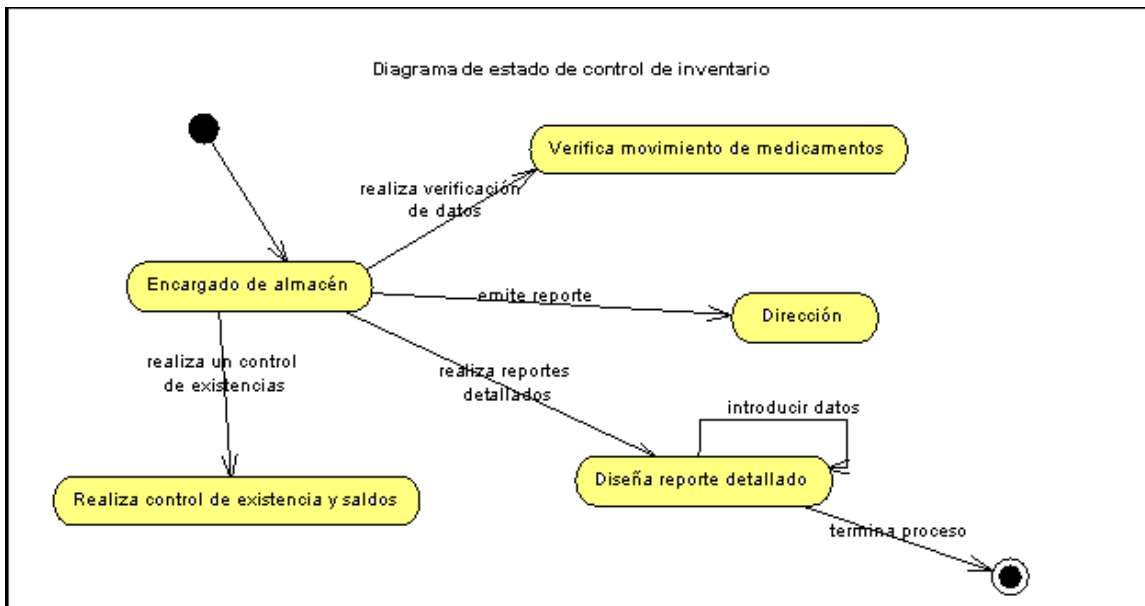
Fuente: Elaboración Propia

Figura 3.17 Diagrama de estado de registro de salida de medicamentos



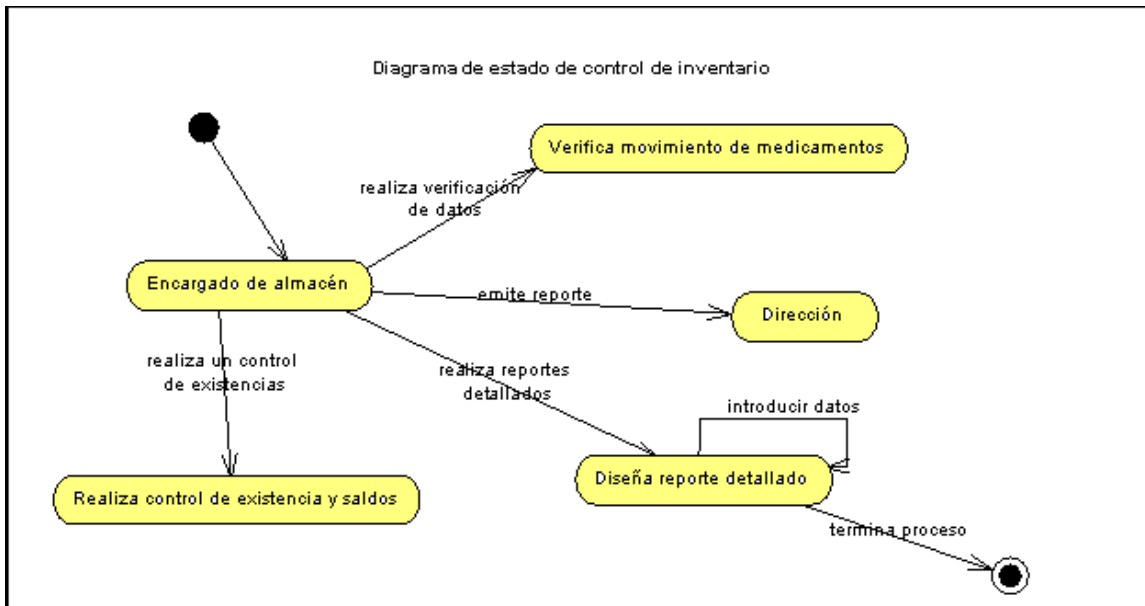
Fuente: Elaboración Propia

Figura 3.18 Diagrama de estado de control de inventario



Fuente: Elaboración Propia

Figura 3.19 Diagrama de estado de consultas y reportes



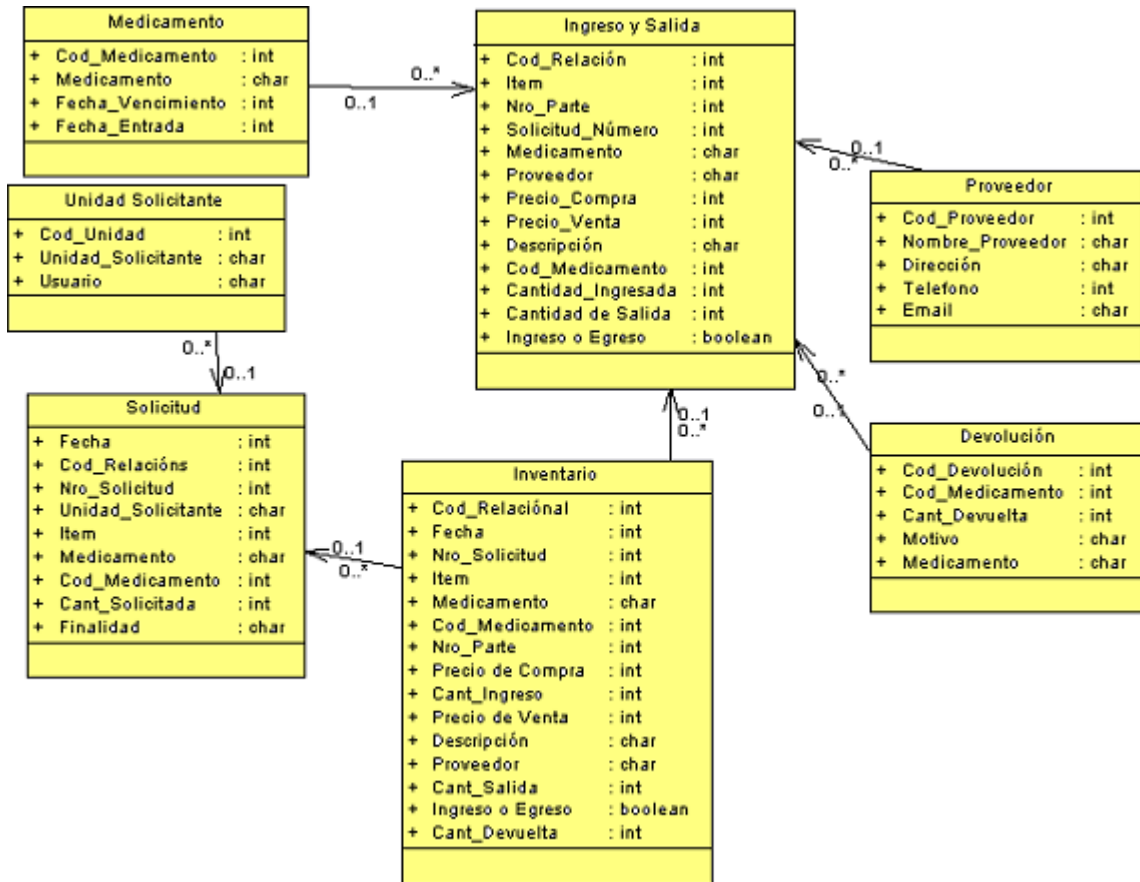
Fuente: Elaboración Propia

3.4 DISEÑO

3.4.1 DIAGRAMA DE CLASES

Este diagrama describe gráficamente la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos. Se define a una clase como categoría o grupo de cosas que tiene atributos o acciones similares (ver ac ápite 2.3.7). A continuación en la figura 3.20 se muestra el diagrama de clases del sistema.

Figura 3.20 Diagrama de clases

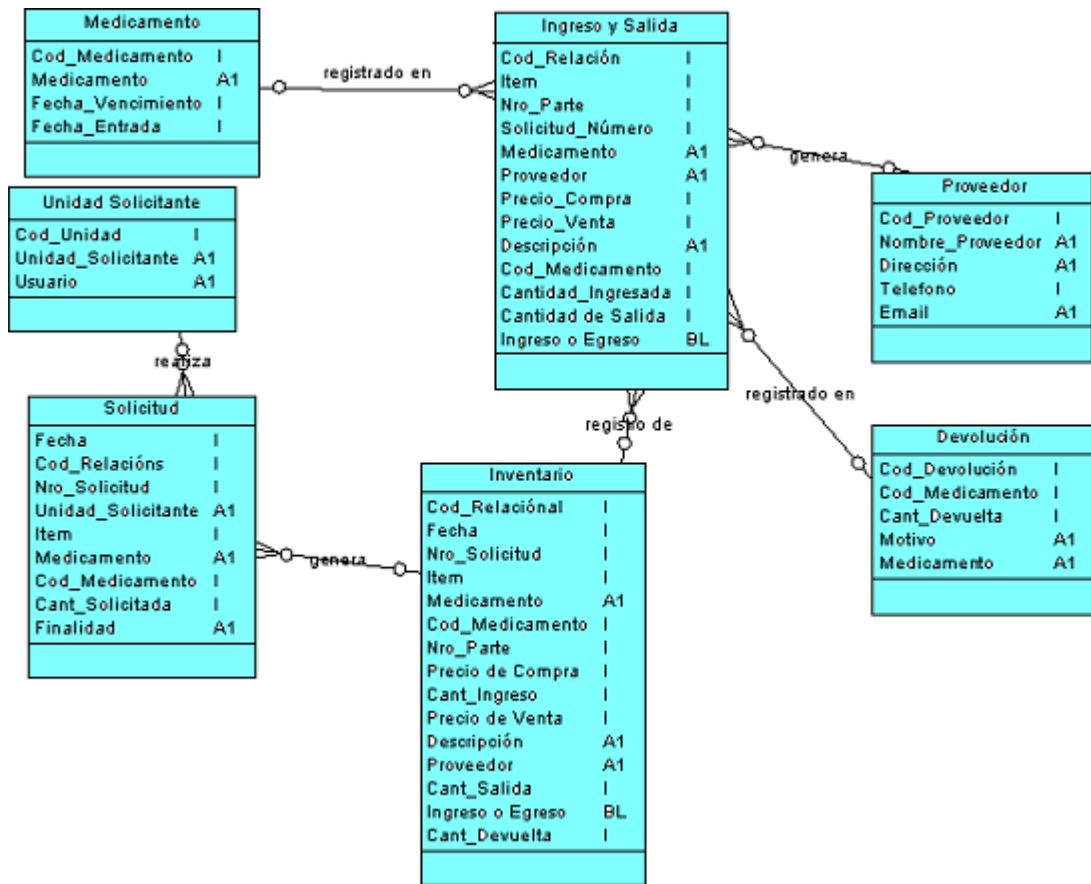


Fuente: Elaboración Propia

3.4.2 DEFINICIÓN DEL ESQUEMA DE BASE DE DATOS

A partir de las especificaciones que se tienen para transformar un diagrama de clases a un modelo entidad relación (ver acápite 2.9), es que a continuación se presenta el modelo que representa de manera gráfica en la figura 3.21 el modelo entidad-relación del sistema

Figura 3.21 Modelo conceptual



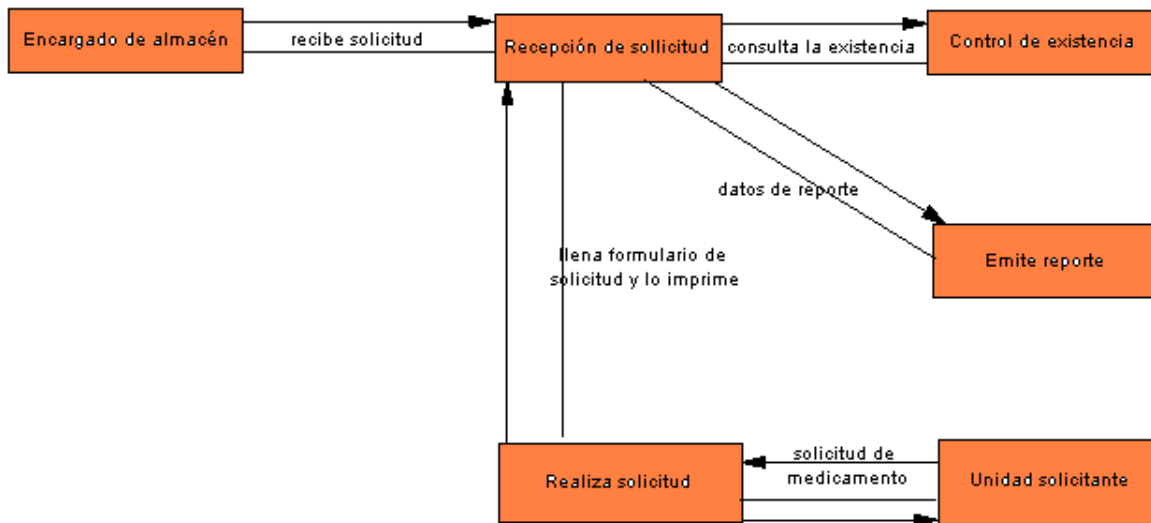
Fuente: Elaboración Propia

3.4.3 DIAGRAMA DE COLABORACION

Los diagramas de colaboración muestra la forma en que los objetos colaboran entre si, mostrando los mensajes que envían entre ellos, destacando el contexto y organización en general de los objetos que interactúan. Dicho de otra manera es la asignación de responsabilidades entre los objetos y mostrar su interacción.

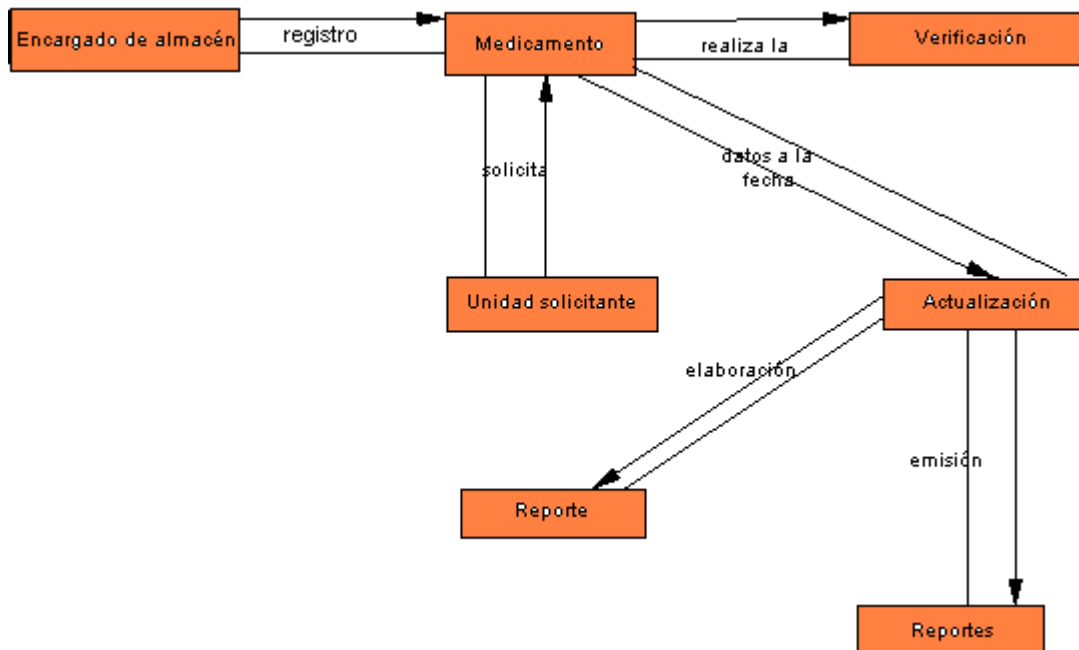
A continuación se muestra los diagramas de colaboración respecto al sistema.

Figura 3.22 Diagrama de colaboración de solicitud de medicamentos



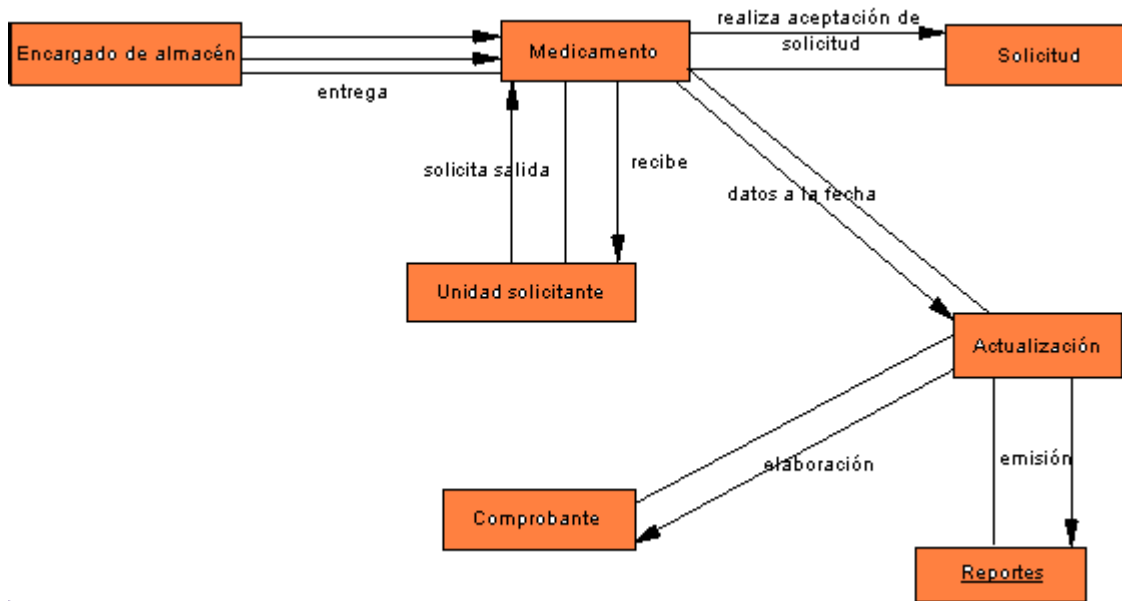
Fuente: Elaboración Propia

Figura 3.23 Diagrama de colaboración ingreso de medicamentos



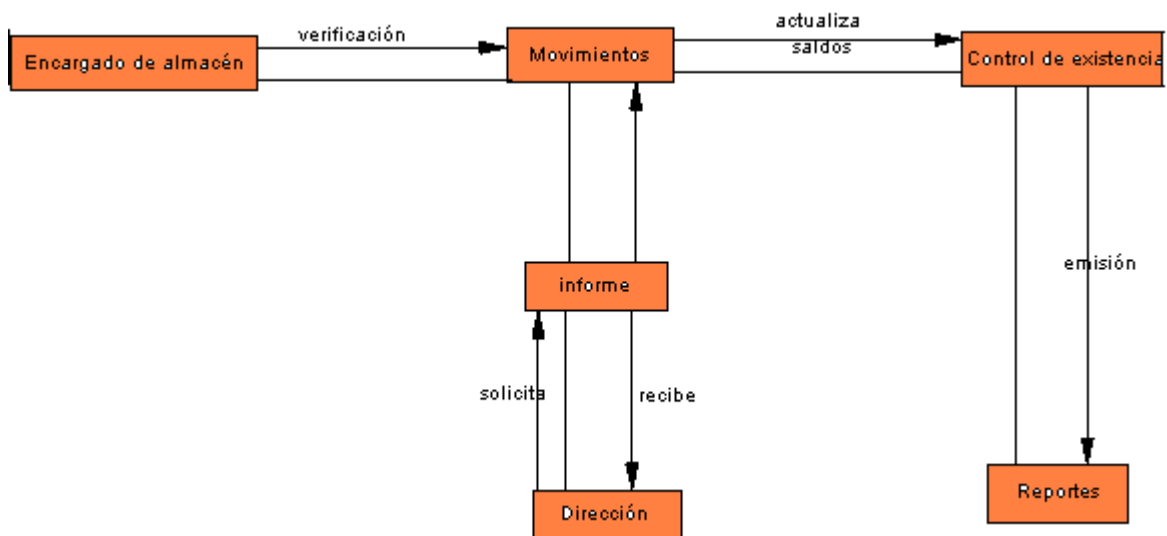
Fuente: Elaboración Propia

Figura 3.24 Diagrama de colaboración de salida de medicamentos



Fuente: Elaboración Propia

Figura 3.25 Diagrama de colaboración de control de inventario



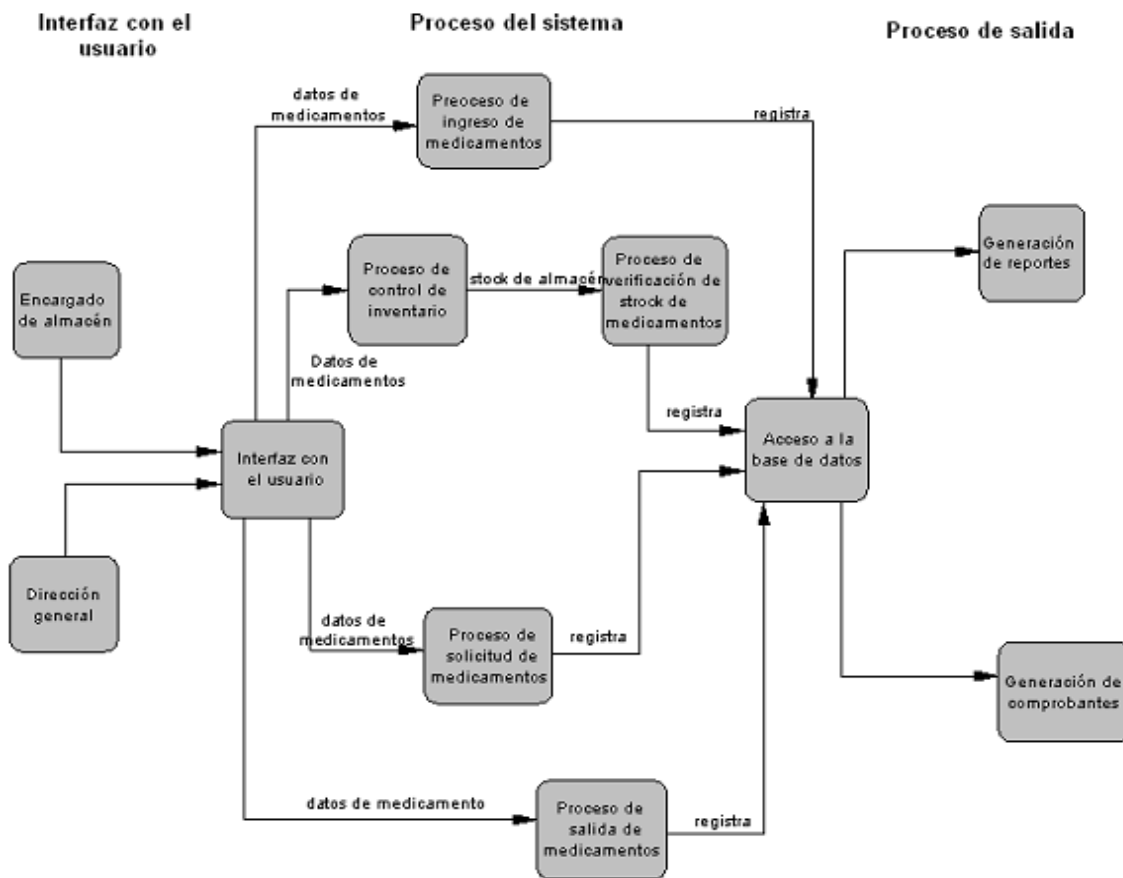
Fuente: Elaboración Propia

3.7 ARQUITECTURA DEL SISTEMA

La arquitectura del sistema esta comprendido tanto en la descripción del diseño y contenido, incluye información sobre la organización fundamental del sistema, que incluye sus componentes, las relaciones entre sí y el ambiente, y los principios que gobiernan su diseño y evolución.

En La figura 3.26 se muestra toda la lógica de aplicación, es decir la arquitectura del software en si.

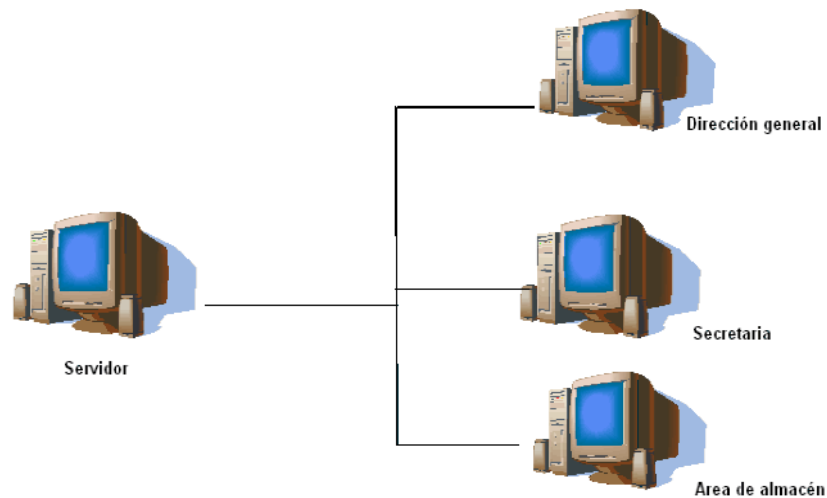
Figura 3.26 Arquitectura del sistema



Fuente: Elaboración Propia

A continuación se presenta el esquema de la arquitectura del sistema en la figura 3.27, donde en cada área se encuentra un computador, y todas ellas se conectan a un servidor común en el cual comparten la información centralizada.

Figura 3.27 Conformación del reparto de equipos



Fuente: Elaboración Propia

3.8 SEGURIDAD DEL SISTEMA

Dado que el sistema corre bajo una intranet se debe considerar por lo menos algunas reglas básicas a cumplirse para que la seguridad del sistema no sea una preocupación para la Clínica (ver acápite 2.4). En la implementación del sistema se considera las siguientes políticas de seguridad:

- **Política de contraseñas.** *El sistema es capaz de realizar la comprobación de contraseñas para los 3 tipos de usuarios que se tiene actualmente, también es capaz de asignar nuevos usuarios con niveles de acceso.*

- **Política de uso adecuado.** *En concreto se especifica que está terminantemente prohibido ejecutar programas que intenten adivinar*

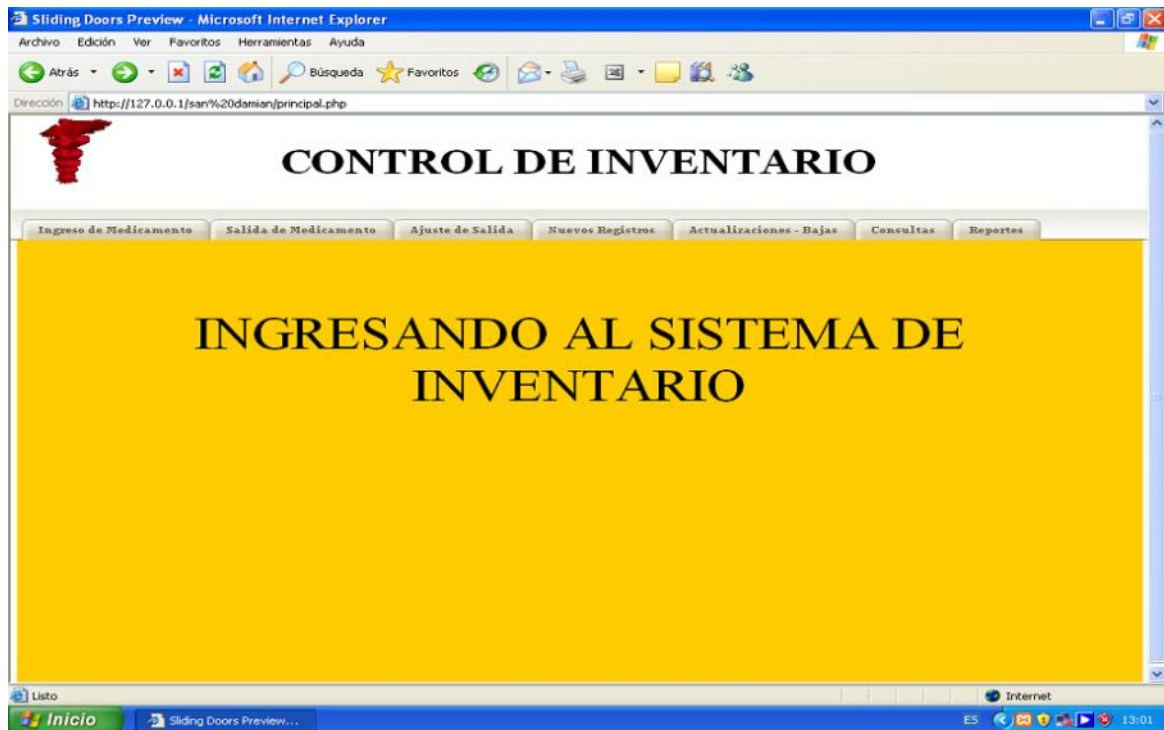
las contraseñas alojadas en las tablas de usuarios de máquinas locales o remotas, o instalar programas maliciosos, insertar dispositivos de almacenamiento secundarios infectados con virus sin previo análisis.

- **Políticas de respaldo.** El administrador del sistema es el responsable de realizar respaldos de la información periódicamente. Cada treinta días deberá efectuarse un respaldo completo del sistema y también deberán ser respaldados todos los archivos que fueron modificados o creados.

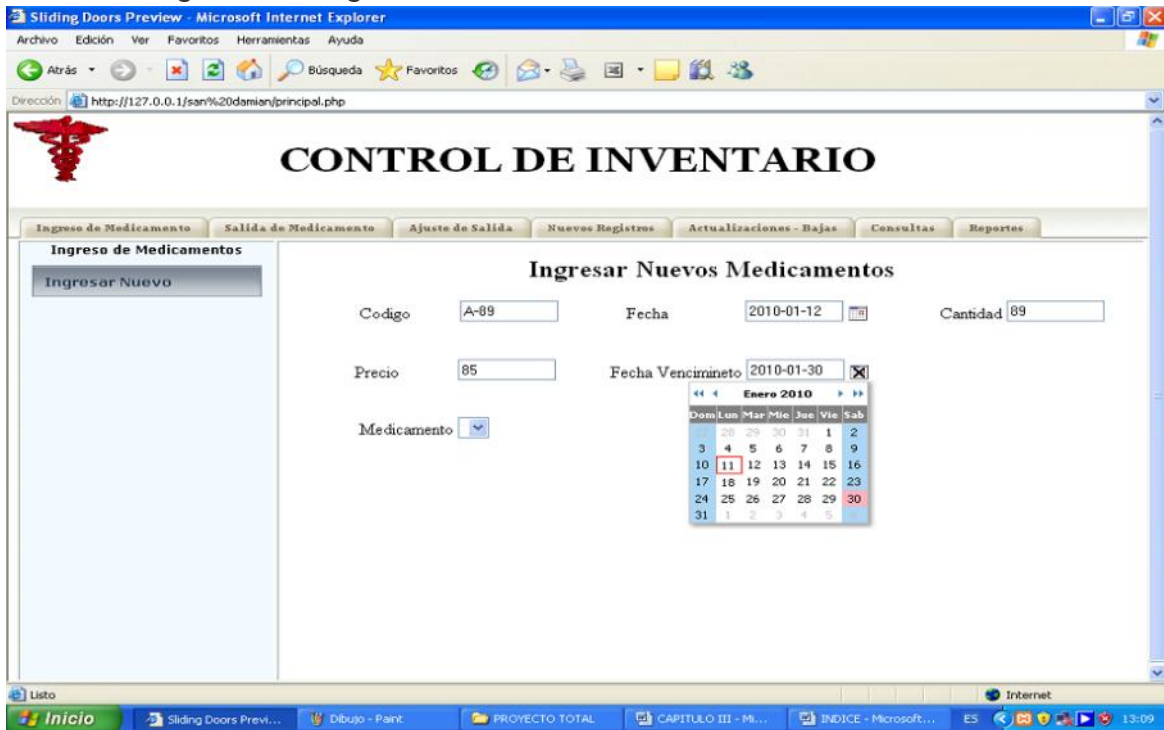
3.9 DISEÑO DE INTERFAZ

A continuación se muestra las principales interfaces del sistema

- El Sistema de Control y seguimiento de inventario de fármacos, “Clínica San Damián”, presenta como pantalla de ingreso la figura 3.28:



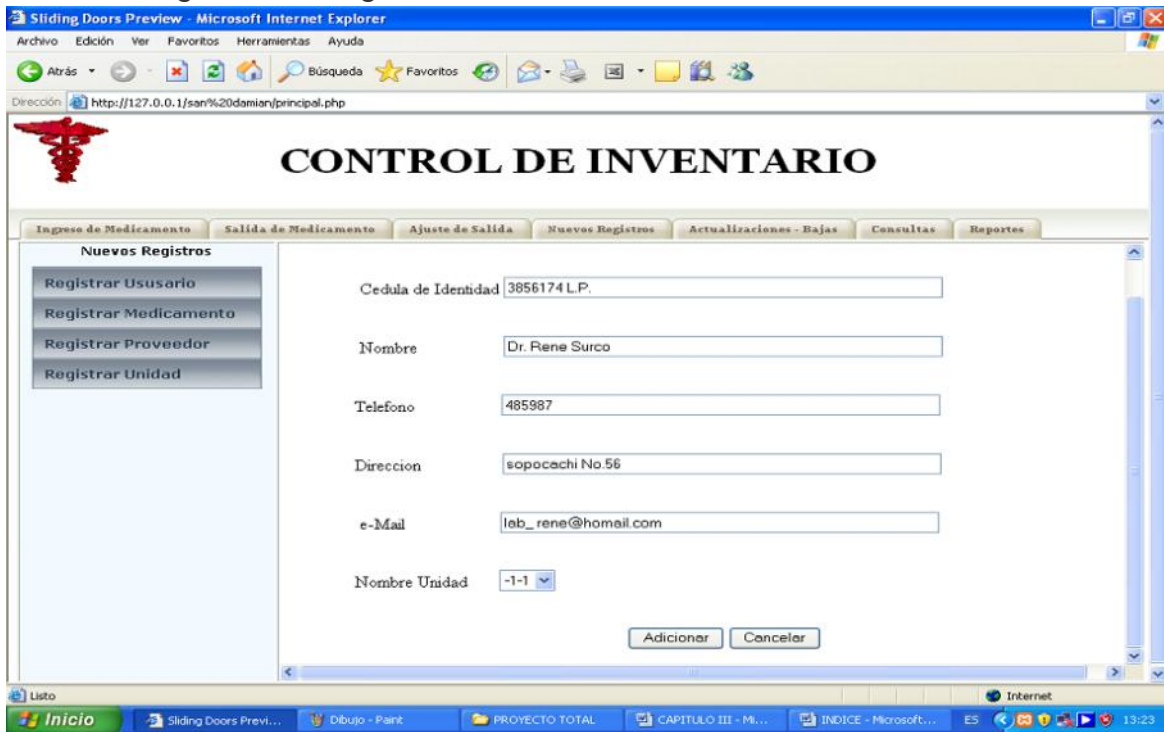
• *Figura 3.29 Ingreso de medicamentos*



• *Figura 3.30 Registro de devolución de medicamentos*



• *Figura 3.31 Registro de la unidad solicitante*



• *Figura 3.32 Consultas*



CAPÍTULO IV

CALIDAD DE SOFTWARE

CALIDAD DE SOFTWARE

4.1 INTRODUCCIÓN

En este capítulo se tratará la medición de la calidad según las métricas mencionadas en el capítulo 2.

En la calidad no es necesariamente llegar al objeto perfecto aunque es deseable, mas al contrario tiene la necesidad y suficiencia para cada contexto de uso en el momento del manejo como ser las métricas internas de la calidad del producto de software, de ahí que se tomo los siguientes criterios de calidad.

- *Aplican a un producto de software no ejecutable.*
- *Aplican durante las etapas de su desarrollo.*
- *Permiten medir la calidad de los entregables intermedios.*
- *Permite predecir la calidad del producto final.*
- *Permiten al usuario iniciar acciones correctivas temprano en el ciclo de desarrollo.*

Estos criterios de calidad están basados en la ISO 9126 – 3 que pueden ser medidos y evaluados por medio de atributos estáticos, se encuentran desarrollados a continuación.

4.2 FUNCIONALIDAD

El punto función es una métrica orientada a la función del software y del

proceso por el cual se desarrolla. Se centra en la funcionalidad o utilidad del programa, los puntos de función se calculan realizando una serie de actividades comenzando por determinar los siguientes números: (ver acápite 2.7.1)

- **Número de entradas de usuarios.** Se cuenta cada entrada de usuario que proporciona o al software diferentes datos orientados a la aplicación.

- **Número de salidas de usuarios.** Estas se refieren a informes, mensajes de error, es decir salidas que proporcionen al usuario información orientada a la aplicación.

- **Número de peticiones de usuario.** Una petición está definida como una entrada interactiva que resulta de la generación de algún tipo de respuesta en forma de salida.

- **Número de archivos.** Se cuenta cada archivo maestro lógico

- **Número de interfaces externas.** Se cuenta todas las interfaces legibles por el ordenador que son solicitados para transmitir información a otro sistema.

De acuerdo a lo mencionado es que se tiene los resultados en la tabla 4.1:

Tabla 4.1. Entradas para el cálculo de funcionalidad

| | |
|----------------------|----|
| Entradas de usuario | 55 |
| Salidas de usuario | 70 |
| Consultas de usuario | 40 |
| Número de archivos | 60 |
| Interfaces externas | 0 |

Fuente: Elaboración Propia

Los puntos de función se calculan rellorando la tabla 4.2 con los datos obtenidos, considerando un factor de ponderación medio.

Tabla 4.2 Calculo de puntos de función sin ajustar.

| Parámetros de medición | Cuenta | | Factor de ponderación MEDIO | = | Totales |
|--------------------------------|--------|---|-----------------------------|---|---------|
| Número de entradas de usuario. | 55 | x | 4 | = | 220 |
| Número de salidas de usuario | 70 | x | 5 | = | 350 |
| Número de consultas de usuario | 40 | x | 4 | = | 80 |
| Número de archivos | 60 | x | 10 | = | 600 |
| Número de interfaces externas | 0 | x | 7 | = | 0 |
| CUENTA TOTAL | | | | | 1250 |

Fuente: Elaboración Propia

La relación que permite calcular los puntos de función es la siguiente:
 $PF = CUENTA_TOTAL * (Grado_de_Confiabilidad + Tasa_de_error * fi)$

Donde:

- *PF = Medida de funcionalidad*
- *CUENTA_TOTAL = Es la suma del valor de las entradas, salidas, peticiones, interfaces externas y archivos*
- *Grado_de_confiabilidad = Es la confiabilidad estimada del sistema.*
- *Tasa_de_error = Probabilidad subjetiva estimada del dominio de la información, este error estimado es del 1%.*

- *Fi = Son valores de ajuste de complejidad que toman los valores de la tabla 4.4 y que dan respuesta a las preguntas de la tabla 4.3*

Tabla 4.3. Ajuste de complejidad del punto función.

| ESCALA | Sin Importancia | Incremental | Moderado | Medio | Significativo | Esencial |
|--|------------------------|--------------------|-----------------|--------------|----------------------|-----------------|
| Factor | 0 | 1 | 2 | 3 | 4 | 5 |
| 1. ¿requiere el sistema copias de seguridad y de recuperación fiables? | | | | | | X |
| 2. ¿Se requiere comunicación de datos? | | | | | X | |
| 3. ¿Existen funciones de proceso distribuidos? | | X | | | | |
| 4. ¿Es crítico el rendimiento? | | X | | | | |
| 5. ¿Será ejecutado el sistema en S.O existente? | | | | | | X |
| 6. ¿Requiere el sistema de entrada interactiva? | | | | X | | |
| 7. ¿Requiere el sistema de entrada de datos interactiva sobre múltiples ventanas? | | | | X | | |
| 8. ¿Se actualizan los archivos maestros de manera interactiva? | | | | X | | |
| 9. ¿Son complejas las entradas, las salidas, los archivos o las peticiones? | | X | | | | |
| 10. ¿Es complejo el procesamiento interno? | | | X | | | |
| 11. ¿Se ha diseñado el código para ser reutilizable? | | | | X | | |
| 12. ¿Están incluidas en el diseño la conversión y la instalación? | | | | | X | |
| 13. ¿Se ha diseñado el sistema para soportar múltiples instalaciones? | | | | | X | |
| 14. ¿Se ha diseñado la aplicación para facilitar los cambios y para ser fácilmente utilizada por el usuario? | | | | | | X |
| Total (Fi) | | | | | | 44 |

Tabla 4.4. Valores de ajuste de complejidad.

| | |
|------------------------|----------|
| Sin importancia | 0 |
| Incremental | 1 |
| Moderado | 2 |
| Medio | 3 |
| Significativo | 4 |
| Esencial | 5 |

Con la obtención de los anteriores datos y considerando un grado de confiabilidad del 65% es que a continuación calculamos el valor de PF:

$$PF = \text{Cuenta_Total} * (\text{Grado_de_Confiabilidad} + \text{Tasa_de_error} * F_i)$$

$$PF = 1250 * (0.65 + 0.01 * 44)$$

$$PF = 1362.5$$

Si consideramos el máximo valor de ajuste de complejidad como $F_i = 70$, se tiene:

$$PF = 1250 * (0.65 + 0.01 * 70)$$

$$PF = 1687.5$$

Entonces si F_i es considerada como el 100%, la relación obtenida entre los puntos será:

$$PF / P_f \text{ máximo} = 1362.5/1687.5 = 0.81$$

Por lo tanto la funcionalidad del sistema es del 81% tomando en cuenta el punto de función máximo.

4.3 CONFIABILIDAD

En la cantidad de tiempo que el software está disponible para su uso, es posible medir la confiabilidad tomando en cuenta la probabilidad del sistema que este libre de fallos en un contexto determinado y durante un periodo de tiempo (ver acápite 2.7.2).

Probabilidad de hallar una falla: $P(T \leq t) = F(t)$

Probabilidad de no hallar una falla: $P(T > t) = 1 - F(t)$

Con $F(t) = F_c * (e^{-1/7 * 12})$

Donde :

$F_c=0.81$: Funcionalidad del sistema.

$=1$: Tasa de fallos en 7 ejecuciones dentro de un mes.

Hallando la confiabilidad:

Tabla 4.5 Cálculo de la confiabilidad.

| Hallando confiabilidad | Prob. Hallar una falla(F(t)) | Prob. de no hallar una falla |
|--|-------------------------------------|---------------------------------|
| $F(t) = F_c * (e^{-1/7 * 12})$ $F(t) = 0.87 * (e^{-1/7 * 12})$ $F(t) = 0.15$ | $P(T \leq t) = F(t)$ 0.15 | $1 - F(t)$ $1 - 0.15 = 0.85$ |

Fuente: Elaboración Propia

Por lo tanto, el sistema presenta una confiabilidad del 0.85, lo que quiere decir que el 85% de las ocasiones, el sistema funciona sin presentar fallos y el resto (15), presenta fallos que no afectan de sobremanera el desempeño global del sistema.

4.4 PORTABILIDAD

Para el presente sistema se tiene que el hardware en el que funciona de manera estable del lado del servidor, esta dado por un equipo Pentium IV, el acceso a este servidor es a través de una intranet donde solo pueden acceder los usuarios autorizados del sistema. Las terminales de donde se accede al servidor tienen características de ser equipos Pentium IV. Por otro lado el Sistema Operativo del lado del servidor es Windows XP, al igual que las terminales de acceso al sistema (ver acápite 2.7.3).

El software es apto para funcionar bajo distintas plataformas, específicamente en Linux debido a que es una aplicación hecha en lenguaje de programación php, Gestor de Base de Datos MySql y servidor Apache que tienen esta característica.

Con respecto al tamaño físico de la aplicación y la Base de Datos, estos ocupan un espacio de 5 Mb aproximadamente. Por lo tanto, podemos concluir que el sistema no requiere de un gran esfuerzo para su traslado de un entorno de Hardware y Software a otro.

4.5 MANTENIBILIDAD

4.5.1 MANTENIMIENTO ADAPTIVO

El mantenimiento adaptativo ocurrirá cuando se cambien las políticas o cuando se cambie la estructura organizacional, o cambie el personal de la Clínica, modificaciones que se harán que el sistema cambie en poca o gran medida, cambios para los cuales el sistema está preparado en adaptarse a algunos de estos casos, pero para otros más complejos se deberá hacer revisión de los procesos y su adaptación con los nuevos cambios que se generen (ver acápite 2.7.4).

4.5.2 MANTENIMIENTO PERFECTIVO

El sistema está completamente abierto a añadir o adicionar nuevas funcionalidades de acuerdo a los nuevos requerimientos del cliente, siempre y cuando sean relacionados con el servicio e información que brinda el sistema (ver acápite 2.7.4).

4.6 FACILIDAD DE USO

La medición de la facilidad de uso se puede entender como la facilidad que el usuario tiene para conocer al sistema, tanto como para comprenderlo, aprenderlo y operarlo. A continuación presentamos en la tabla 4.6 los resultados obtenidos en la enseñanza de la manipulación del software a los tres usuarios.

Tabla 4.6 Resultados para el cálculo de facilidad de uso.

| USUARIOS | FACILIDAD DE COMPRENSIÓN | FACILIDAD DE APRENDIZAJE | FACILIDAD DE OPERACIÓN |
|-----------------|---------------------------------|---------------------------------|-------------------------------|
| Usuario 1 | 91% | 90% | 91% |
| Usuario 2 | 94% | 90% | 92% |
| Usuario 3 | 96% | 93% | 94% |
| Promedio | 93.7% | 91 % | 92.3% |

Por lo tanto de acuerdo a los resultados de la tabla 4.6 se obtuvo que la facilidad de uso del sistema es de un 92 %.

4.7 CONCLUSIONES

Finalmente, se puede apreciar que los objetivos se han satisfecho satisfactoriamente con la implantación del sistema, el cual reemplazó el proceso manual que se administraba dentro de almacén y ha reducido lo que es el manejo de inventarios envió de reporte y comprobantes internos y externos.

CAPÍTULO V
MARCO CONCLUSIVO

MARCO CONCLUSIVO

5.1 CONCLUSIONES

Luego de haber realizado el análisis y diseño del sistema de control y seguimiento de inventario de fármacos. Se logro obtener las siguientes conclusiones:

- *Se logro modernizar las actividades más importantes en el almacén de la clínica como ser el caso de un buen control de inventario de fármacos.*
- *Acortar el tiempo de búsqueda, acceso y suministro de medicamentos requeridos en almacén.*
- *Permite realizar una mejor atención a la Unidades Solicitantes*
- *El control de los medicamentos que son solicitados, ingresan y salen de almacén hacia las diferentes sucursales ya no es un problema gracias a la manera eficaz de como fue encarada por el sistema*
- *Información precisa y confiable*
- *Control de Stocks mínimo de los medicamentos de almacén*
- *Interfaz amigable para el usuario con pantallas comprensibles y de fácil manejo*

5.2 RECOMENDACIONES

Con la finalización del presente proyecto se pueden efectuar las siguientes recomendaciones:

- *Utilizar las herramientas similares para futuras construcciones de software*
- *Se debe tener sumo cuidado respecto a las claves de acceso que son amigables a los usuarios por única vez*
- *Se debe realizar copias de seguridad de la base de datos*
- *Prohibir el ingreso de personas ajenas a almacén*
- *Sacar circulares internas para el buen manejo e higiene del computador e implementos*

REFERENCIAS BIBLIOGRÁFICAS

[Rumbaugh 97] Técnicas de Modelado de Objetos (OMT), Rumbaugh

[González 05] Desarrollo Web con PHP y MySQL Ing. Joel González Estrada

[Mariano 03] Teoría de Redes Informáticas, Mariano López

[Douglas 96] TC / IP Volumen Nro 1. Douglas Comer

[Jacob, 2000] Jacobson/Booch/Rumbaugh El lenguaje Unificado de Modelado Manual de Referencia.

[Larman, 1999] Larman Carig –“UML y patrones” - México 1999 1ra edición

[Press, 1998] Roger S. Pressman 2003, Ingeniería del Software un Enfoque Práctico, Quinta Edición

[Plati, 2003] Platitini Velthuis M. & Garcia Rubio F.J., Calidad en el Desarrollo y Mantenimiento del Software, 2003

[Rosen, 1998] “Ciencia para todos”, Colegio Nacional y fondo de cultura, México

[Merca, 2006] Osvaldo Mercado Torrico Sistema de Información para el Seguimiento y Control del Registro de Bueques

[Horten, 2007] Carla Hortensia Nadeau Inchausti. Sistema integrado de administración y gestión CREATRONIC SRL,

REFERENCIAS ELECTRÓNICAS

[Ferrer,2005] Xavier Ferre Grau, María Isabel Sánchez Segunda. Desarrollo Orientado a Objetos con UML

[Geoci,2000] <http://www.geocities.com/SiliconValley/2208/Insituacion.html>.

[Isfti,2000]<http://www.isftic.mepsyd.es/w3/eos/MaterialesEducativos/mem2003/prog/gramacion/actividades/tecnicas.html>

[Juan C,1995] Juan Cesar Martínez
<http://www.ci.ulsa.mx/~elinodocencia/ctrldesa/ISO14764.pdf>

[Thale,2007] <http://thales.cica.es/rd/Recursos/rd98/Matematicas/29/mixto.html>

[Ref. 1] PHP con MySQL Ing. Cedric Simon <http://www.solucionjava.com>

[Ref. 2] <https://www.pid.dsic.upv.es> Departamento de Sistemas Informáticos y Computación, Universidad Politécnica de Valencia

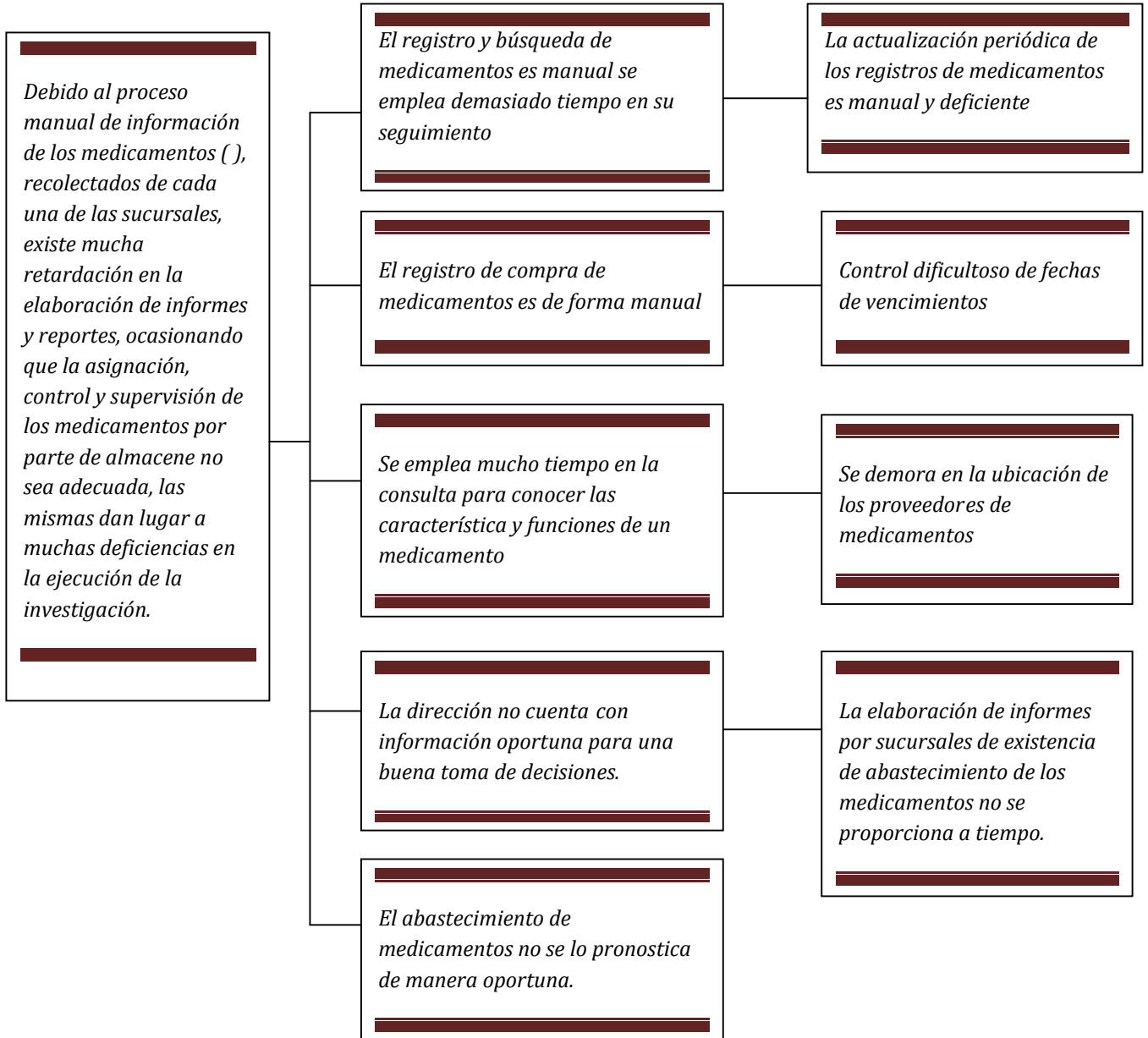
[Ref. 3] <http://www.alarcos.inf-cr.uclm.es> Norma de gestión de la calidad y garantía de la calidad, parte 3.

[Ref. 4] <http://www.informatizate.net> - RUP

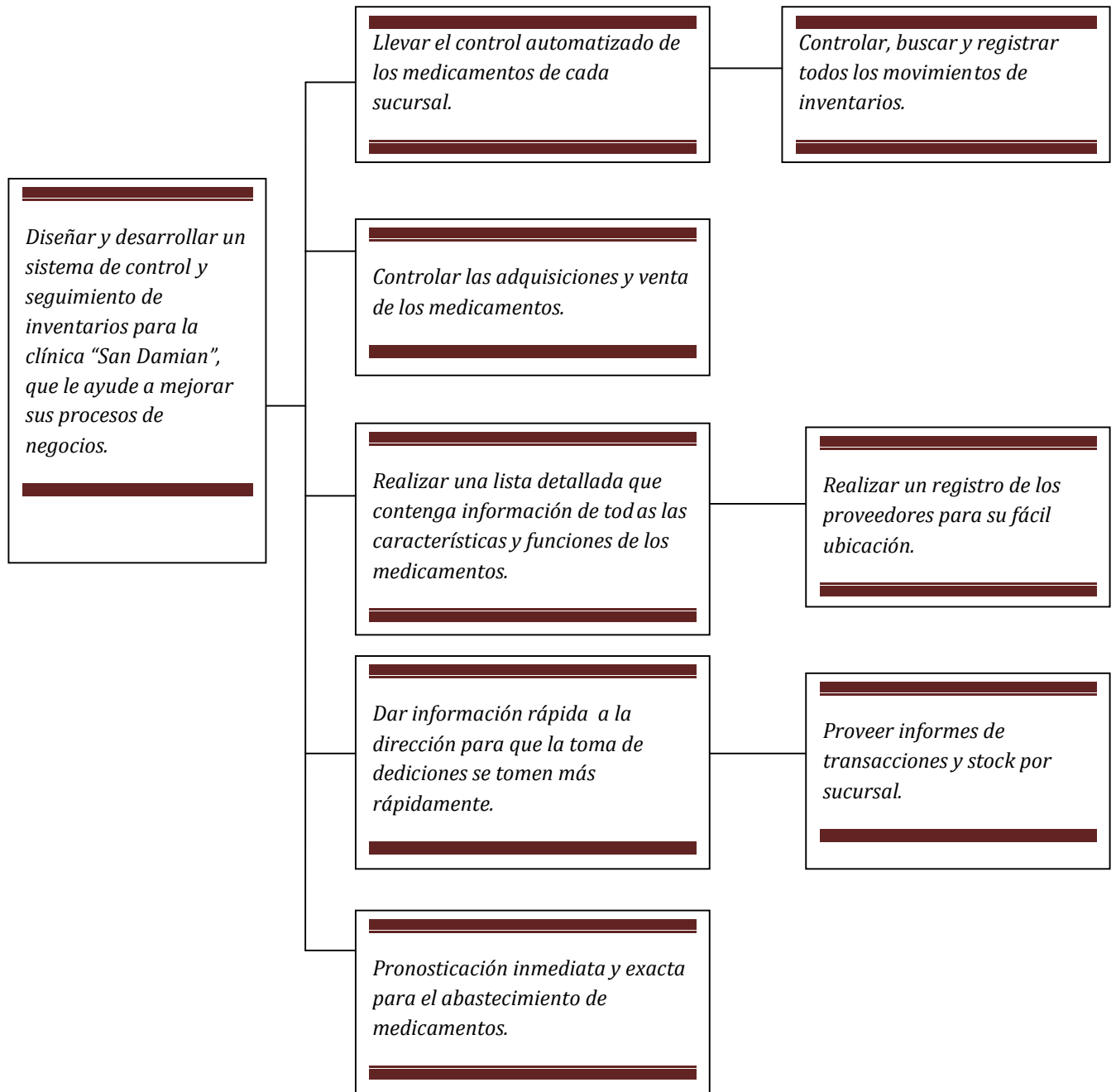
[Ref. 5] <http://www.pid.dsic.upv.es/C1Material/Documentos> - Inventario

ANEXOS

ARBOL DE PROBLEMAS



ARBOL DE OBJETIVOS



DOCUMENTACIÓN

MARCO LÓGICO

| RESUMEN NARRATIVO | INDICADORES VERIFICABLES | MEDIOS DE VERIFICACIÓN | SUPUESTOS |
|---|--|--|---|
| <p><u>Fin.</u>- Mejorar el control de las entradas y saldas de los medicamentos.</p> | <p>-Administración automatizada y en red de la Clínica</p> | <p>-Mejoramiento en las utilidades de la Clínica</p> | <p>- El funcionamiento de la clínica sea normal. - Los médicos que interactúan con el sistema den datos fehacientes necesarios para el trabajo efectivo.</p> |
| <p><u>Propósito.</u>- Diseñar y desarrollar un sistema de control y seguimiento de inventario para mejorar el control de entradas y salidas de los medicamentos. Que le ayude a mejorar sus procesos de negocios.</p> | <p>-Seguimientos de la información que se maneja en la clínica -reducción de tiempo en la realización de informes - el inventario se realiza de acuerdo a su necesidad -El sistema controlara el registro y búsqueda de activos del almacén. Una lista detallada que contenga todas las características básicas de los medicamentos.</p> | <p>- Reportes diarios, semanales y mensuales emitidos por el sistema - toma de decisiones con información confiable segura. -Reportes sobre los productos y su existencia. -Reportes sobre existencia de medicamentos.</p> | <p>-Se utiliza un modelo de inventario adecuado a los requerimientos de la clínica -que el funcionamiento de la clínica sea normal -Se utilizara modelo de inventarios para el control de abastecimiento de los medicamentos aprobado por la Clínica. -La Clínica apruebe el software satisfactoriamente.</p> |

| | | | |
|---|---|--|---|
| <p><u>Componentes o resultados.-</u></p> <ul style="list-style-type: none"> -Disminución de tiempo en la elaboración de reportes -Diseñar una base de datos segura y confiable -Diseñar un modelo de inventarios para el control de entradas y salidas de medicamentos -Diseño de formularios de seguimiento para los medicamentos -La Creación de la lista detallada con información de los medicamentos Controlar, buscar y registrar todos los movimientos de inventario. Realizar un registro de proveedores para su fácil ubicación. Dar información rápida a la dirección para que la toma de decisiones se tome mas rápidamente Pronosticación inmediata y exacta para el abastecimiento de medicamentos. | <ul style="list-style-type: none"> - Los inventarios se manejan de manera mas rápida en un 95% a partir de su implementación -los informes que se emiten reducen un 90% los errores a partir de su implementación | <ul style="list-style-type: none"> - Reportes de estock actualizados -reportes de inventarios, confiable en un menor tiempo - informes y documentos emitidos por el sistema -informes del responsable del departamento de almacén -La información precisa y segura de las características de los medicamentos | <ul style="list-style-type: none"> -capacitación al personal sobre el sistema a ser implementado -se cuente con el equipo de computación con el cual sea capaz de ejecutarse el sistema elaborado - se cuente con todo el material de escritorio para emitir los reportes que brinde el sistema. |
| <p><u>Actividades.-</u></p> <ul style="list-style-type: none"> -recopilación de información -análisis Análisis de datos Diagnostico -Diseño Estructurar el sistema | <ul style="list-style-type: none"> Recopilación Análisis Diseño Implementación Equipo de computación | <ul style="list-style-type: none"> -Orden de los pedidos realizados a la dirección -informes realizados por cada sucursal | <ul style="list-style-type: none"> - Recabar datos necesarios para poder implementar el software y así solucionar el problema identificado. - posibilidad de tener acceso a documentos y informes que permiten su revisión |

