



**UNIVERSIDAD NACIONAL AUTÓNOMA
DE MÉXICO**

FACULTAD DE CIENCIAS

**SISTEMA SIMULADOR DE
HATOS LECHEROS A 7 AÑOS**

T E S I S
QUE PARA OBTENER EL TÍTULO DE:
M A T E M A T I C O
P R E S E N T A :
ÁNGEL REYES GARCÍA

DIRECTOR DE TESIS: M. EN I. MARÍA DE LA LUZ GASCA SOTO



FACULTAD DE CIENCIAS
U.N.A.M.



2005

FACULTAD DE CIENCIAS
SECCIÓN ESCOLAR

m351304



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.



UNIVERSIDAD NACIONAL
AUTÓNOMA DE
MÉXICO

ACT. MAURICIO AGUILAR GONZÁLEZ
Jefe de la División de Estudios Profesionales de la
Facultad de Ciencias
Presente.

Comunicamos a usted que hemos revisado el trabajo escrito:

“Sistema Simulador de Hatos Lecheros a 7 años”

realizado por **Reyes García Ángel**

con número de cuenta **08053301-8**, pasante de la carrera de **Matemáticas**.

Dicho trabajo cuenta con nuestro voto aprobatorio.

Atentamente

Director de Tesis
Propietario

M. en I. **María de Luz Gasca Soto**

Propietario

Dr. **Juan Bautista Climént Bonilla**

Propietario

Dra. **Amparo López Gaona**

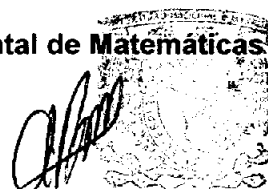
Suplente

Dra. **Elisa Viso Gurovich**

Suplente

M. en C. **María Guadalupe Elena Ibarquingoitia González**

Consejo Departamental de **Matemáticas**



M. en C. **Alejandro Bravo Mojica**

FACULTAD DE CIENCIAS
CONSEJO DEPARTAMENTAL
DE
MATEMÁTICAS

Sistema Simulador de Hatos Lecheros a 7 Años

Ángel Reyes García

2005

Agradecimientos

A la maestra María de Luz Gasca Soto, por darme la oportunidad de realizar este proyecto, al Dr. Juan B. Climent B. quién me confió la responsabilidad de desarrollar el Simulador.

A la Dra. Amparo López Gaona, Dra. Elisa Viso Gurovich, M. en C. María Guadalupe Elena Ibarguengoitia González por sus valiosos comentarios y sugerencias.

- **A la Facultad de Ciencias que me espero pacientemente.**

- ***A la memoria de mis Padres y de mis Hermanos.***

- ***A Elizabeth, Ana Sofía y Pablo que son mis amores y compañeros de mi vida.***

- ***Con mucho cariño a mis herman@s: Berna, Pilar, Rogelio y Victoria.***

Índice General

Prefacio	XIII
I Marco teórico	
Capítulo 1. Introducción.	1
1.1. Sistema de Información	1
1.1.1 Sistemas	1
1.1.2 Tipos de sistemas	2
1.1.3 Evolución del Software	4
1.1.4 Bases de datos	5
1.2 Aspectos del Proceso de Desarrollo	7
1.2.1 El Factor Humano	8
1.2.2 La Organización	9
1.3 Documentación y Mantenimiento	9
1.3.1 Documentación	10
1.3.2 Mantenimiento	11
1.4 Ingeniería del Software	12
1.4.1 Evolución de la Ingeniería del Software.	12
Capítulo 2. Ingeniería del Software.	14
2.1 Ingeniería de software	14
2.2 Ciclo de Vida	15
2.2.1 Elementos del Ciclo de Vida	15
2.2.2 Tipos de Ciclo de Vida	18
Ciclo de Vida Lineal	18
Modelo basado en prototipos desechables	23
Modelo basado en prototipo incremental	24
Ciclo de Vida en Espiral	26
Capítulo 3. UML	28
3.1 Diagramas recomendados	29
3.2 Diagrama de casos de uso	30
3.2.1 Elementos de los diagramas de casos de uso	30
3.2.2 Modelado del contexto	31
3.2.3 Modelado de requisitos	31
3.2.4 Diagrama de casos de uso	31
3.3 Diagramas de clase	32
3.3.1 La Clase	35
3.3.2 Relaciones entre clases	35

3.4 Diagramas de objetos	37
3.5 Diagrama de componentes	38
3.6 Diagramas de despliegue	38
3.7 Diagramas de secuencia	40
3.8 Diagramas de colaboración	40
3.9 Diagramas de paquetes	42
3.10 Diagrama de distribución	43
3.11 Diagrama de estados	44
II Aplicación	
Capítulo 4. Panorama general al sistema de hatos	46
4.1 Introducción	46
4.2 Origen del sistema	47
4.3 Supuestos básicos	48
Capítulo 5. Análisis del Sistema	51
5.1 Introducción	51
5.2 Análisis del sistema	52
5.2.1 Lista de características	52
I. Movimiento de ganado	53
II. Engorda de machos.	53
III. Producción de Leche.	54
IV. Eliminación de ganado	54
V. Ingresos	54
VI. Costos globales.	54
5.2.2 Requerimientos de software	54
5.2.3 Requerimiento de interfaz.	55
5.2.4 Requerimientos Especiales del Sistema	55
5.2.5 Diagramas de Casos de Uso	56
5.2.6 Descripción de Casos de Uso	61
Capítulo 6. Diseño del Sistema	64
Introducción	64
6.1 Definición de Subsistemas	64
6.2 Mecanismos genéricos de diseño	65
6.3 Posibilidades de reuso	65
6.4 Diagrama de diseño	65
6.4.1 Diagrama de paquetes y capas	65
6.4.2 Diagramas de Entidad relación	69

Capítulo 7. Prototipo del Sistema	70
7.1 Prototipo	70
Supuestos Básicos	71
Movimiento de ganado	71
Engorda de machos	73
Producción de leche	74
Eliminación de ganado para rastro	75
Ingresos	75
Costos globales	76
Opción Reportes del Menú principal	77
I.- Movimiento de ganado	78
a.- Compras anuales	79
b.- Ganado no comprado en la etapa (recría del hato)	79
c.- Población inicial del ganado productor	80
d.- Estructura del hato	81
e.- Muertes y desechos del ganado productor	83
f. Población final del ganado productor	83
g. Tamaño del hato	84
h. Capacidad disponible	85
II. Proyección de partos	87
III.- Recría	88
IV.- Engorda de machos	90
V.- Producción de leche	91
VI.- Producción de carne	92
VII.- Excedentes en la producción de vaquillas de reemplazo	92
VIII.- Ingresos	94
IX.- Costos globales	95
X.- Costos de "X" por litro	96
XI.- Utilidades	96
XII.- Razón: beneficio-costo	97
7.2 Evaluación del simulador	99
Funcionamiento general del Simulador.	99
Conclusiones	102
III Apéndices	
Apéndice A	103
Resultados del Simulador de Hatos Lecheros a 7 años para el rancho	103
Movimiento de Ganado	103
Compra de Ganado	103
Costos de producción	104
Eliminación de Ganado para Rastro	104
Ingresos	104

Producción de leche	105
Muertes y desechos	105
Cálculos para la Proyección del Simulador de Hatos	
Lecheros a 7 años	107
Apéndice B	115
Glosario	115
Bibliografía	119

Índice de Figuras

Figura 2.1. Elementos del Ciclo de Vida	16
Figura 2. 2. Descomposición de una fase en subfases.	17
Figura 2. 3. Esquema general de operación de una fase	17
Figura 2. 4. Ciclo de Vida Lineal	19
Figura 2. 5 Ciclo de Vida con Prototipos Desechables	24
Figura 2. 6. Ciclo de Vida Incremental	25
Figura 2. 7. Prototipo Heterogéneo	26
Figura 2. 8. Ciclo de Vida en Espiral	27
Figura 3.1. Diagrama de casos de uso	32
Figura 3.2. La Clase	35
Figura 3.3. Diagrama de Clases	37
Figura 3.4. Diagrama de Objetos	38
Figura 3.5. Diagrama de componentes	39
Figura 3.6. Diagrama de despliegue	40
Figura 3.7. Diagrama de secuencia	41
Figura 3.8. Diagrama de colaboración	41
Figura 3.9. Diagrama de paquetes	43
Figura 3.10. Diagrama de distribución	44
Figura 3.11. Diagrama de estados	44
Figura 5.1. Funcionamiento General del Simulador	57
Figura 5.2. Diagrama de casos de uso del Simulador 1er. nivel	58
Figura 5.3. Diagrama de Casos del Simulador 2do. Nivel Opción "Supuestos básicos"	59
Figura 5.4. Diagrama de Casos del simulador 2do. Nivel Opción "Reportes"	60
Figura 6. 1 Definición de Arquitectura del Simulador de Hatos Lecheros	66
Figura 6. 2 Diagrama de Paquetes	67
Figura 6. 3 Diagrama de modelo de datos generales	68
Figura 6. 4 Diagrama detallado del manejo de datos.	68
Figura 6. 5 Diagrama entidad-Relación	69
Figura 7. 1. Simulador de Hatos Lecheros.	70
Figura 7. 2 Menú Principal	71
Figura 7. 3 Supuestos Básicos.	72
Figura 7. 4 Engorda de Machos	73
Figura 7. 5 Producción de Leche	74
Figura 7. 6 Eliminación de Ganado para Rastro.	75

Figura 7. 7 Ingresos.	76
Figura 7. 8 Costos Globales.	77
Figura 7. 9 Menú de los reportes	78
Figura 7. 10 Reporte de Compras Anuales de Ganado del Primer año al Séptimo año	79
Figura 7. 11 Reporte de Ganado no Comprado en la Etapa	80
Figura 7. 12 Reporte Población Inicial del Ganado Productor	81
Figura 7. 13 Reporte Estructura del Hato	82
Figura 7. 14 Reporte Muertes y Desechos del Ganado Producto.	83
Figura 7. 15 Reporte Población final del ganado pProductor.	84
Figura 7. 16 Reporte Tamaño del Hato.	85
Figura 7. 17 Reporte Capacidad Disponible.	86
Figura 7. 18 Reporte Proyección de Partos	88
Figura 7. 19 Reporte Recría	89
Figura 7. 20 Reporte Engorda de Machos	91
Figura 7. 21 Reporte Producción de Leche	91
Figura 7. 22 Reporte Producción de Carne	92
Figura 7. 23 Reporte Excedente en la Producción de Vaquillas de Reemplazo	94
Figura 7. 24 Reporte Ingresos.	95
Figura 7. 25 Reporte Costos globales.	95
Figura 7. 26 Reporte Costo del Litro de Leche por Concepto "X"	96
Figura 7. 27 Reporte Utilidades.	97
Figura 7. 28 Reporte Razón Beneficio: Costo	99

Resumen

El presente trabajo tiene la finalidad de mostrar el uso de la Ingeniería del Software en el desarrollo de sistemas de información de apoyo a la Academia. Así mismo, aplicar los conocimientos en el proceso de desarrollo y la evolución de sistemas de software, que solucionen problemas dentro de diferentes áreas, utilizando las herramientas adecuadas para optimizar los recursos de tiempo y costo.

Particularmente se presenta una aplicación en un Simulador de Hatos Lecheros con Proyección a siete años, el cual pretende servir a profesores y alumnos de carrera afines a la producción de alimentos agropecuarios, en la integración de algunos de los aspectos administrativos y operativos a las explotaciones de ganado lecheros.

Prefacio

Una de las preocupaciones actuales de la industria de la computación es el desarrollo de software y sistemas de una manera rápida y de bajo costo. Para hacer un buen uso de las bondades de las computadoras, se requiere de un software de mayor complejidad, confiable y de calidad.

La necesidad de crear mejor software se aplica tanto en la industria del software como dentro de las empresas que desarrollan sus propias aplicaciones de computación. Las organizaciones relacionadas con la tecnología de la información requieren crear y modificar aplicaciones en forma más rápida. Si el desarrollo de aplicaciones tarda de dos a tres años, las empresas no pueden crear otras aplicaciones ni reaccionar ante la competencia de manera rápida.

En docencia, por ejemplo, la necesidad de tener un flujo de información rápida y constante, obliga a tener instrumentos de manipulación eficiente. Particularmente, en las escuelas y Facultades de Medicina Veterinaria y Ciencias Agropecuarias, se requiere tener recursos humanos cada vez más competitivos y que sepan adecuarse al manejo de grandes cantidades de información.

El presente trabajo tiene la finalidad de mostrar la importancia de la Ingeniería del Software en el desarrollo de sistemas de información de apoyo a la Academia. Así mismo, aplicar los conocimientos en el proceso de desarrollo y la evolución de sistemas de software, que solucionen problemas dentro de diferentes áreas, utilizando las herramientas adecuadas para optimizar los recursos de tiempo y costo sin menoscabo de la calidad y la confiabilidad de los sistemas.

Además, se pretende explicar y aplicar un proceso de desarrollo de software sistémico acorde a la magnitud y complejidad de proyectos de aplicación, sean o no nuevos desarrollos, teniendo en consideración la evolución y el cambio de los mismos.

Por otro lado, se describen y aplican las herramientas necesarias para la especificación, diseño, verificación, validación e instrumentación de sistemas de software.

Específicamente, el sistema realiza una evaluación de la producción de leche con proyecciones en el tiempo, la cual es un proceso multivariable cuyo análisis es lento y, por ende, costoso por lo que se requiere contar con una herramienta computarizada que modele el proceso y facilite su estudio. La aplicación se centra en Hatos lecheros, con el fin de ver su proyección a 7 años

El diseño y la programación del simulador fueron enfocados para cubrir la mayor cantidad de actividades del área y para manejar las bondades del sistema de una manera fácil y sencilla, sin más necesidad que ser útil.

El trabajo está dividido en tres partes: la primera establece un marco teórico sobre la Ingeniería del Software; en la segunda se presenta la aplicación del Sistema de Hatos Lecheros con Proyección a Siete Años, y en la tercera parte algunos apéndices que complementan el trabajo. Se tiene la siguiente distribución del material por capítulos.

En el Capítulo 1 se presentan los conceptos básicos que se emplean en el trabajo, facilitando su comprensión. Así se definen conceptos como sistema, software y base de datos, entre otros.

El Capítulo 2 reúne los preceptos fundamentales de la ingeniería de software, destacando los modelos de ciclo de vida y prototipos.

En el Capítulo 3, se presenta el desarrollo de UML como notación orientada a objetos, con los diversos diagramas que dan visión estática y dinámica a los desarrollos.

En el Capítulo 4 se proporciona una visión general de la problemática y complejidad de la industria lechera, introducción al lector hacia el sistema de Hatos.

En el Capítulo 5 se realiza el análisis del sistema y el diseño del mismo, a través de varias fases. Asimismo, se definen los conceptos y características del simulador de hatos lecheros.

El Capítulo 6 brinda información sobre el diseño del sistema, definiendo políticas de funcionamiento, subsistemas y diagramas de diseño.

Finalmente, en el Capítulo 7, se presenta el prototipo del simulador, con todas sus pantallas y opciones que permiten establecer la producción de leche y evaluar la rentabilidad de la empresa, por medio de la relación costo/beneficio. Asimismo, se aplica en un proyecto real, determinando las ventajas y desventajas que se tienen en el sistema.

Se presentan conclusiones donde el proyecto se ha llevado a cabo, no sólo con datos reales, sino hipotéticos, derivados de información teórica, sólida (científica), ajustada con sentido práctico, según el buen saber y entender del usuario y sus fuentes de información.

Adicionalmente, la información detallada con respecto a las fórmulas usadas en la proyección, diagramas, pantallas y un glosario, se presentan en apéndices.

I Marco Teórico

Capítulo 1. Introducción.

En este capítulo se presenta de manera resumida una serie de conceptos necesarios para la comprensión del trabajo desarrollado. Asimismo, se da una visión de la evolución de los sistemas de información en las últimas décadas.

1.1. Sistema de Información

La evolución del software está dividida en varias etapas, una de ellas es la llamada *crisis del software*. Esta crisis fue el resultado de la introducción de la tercera generación del hardware. El hardware dejó de ser un obstáculo para el desarrollo de la informática; redujo los costos, mejoró la calidad y eficiencia en el software producido. La crisis se caracterizó por los siguientes problemas:

- Imprecisión en la planificación del proyecto y estimación de los costos.
- Baja calidad del software.
- Dificultad de mantenimiento de programas con un diseño poco estructurado.

A raíz de esta crisis se vio la necesidad de crear estándares de desarrollo dando lugar a lo que hoy llamamos Ingeniería de Software, la cual se define como el establecimiento y uso de principios de la ingeniería a fin de obtener software que sea fiable y que funcione eficientemente.

1.1.1 Sistemas

El concepto de sistema ha sido utilizado por líneas de pensamiento diferente. La primera es la teoría de sistemas generales, corriente iniciada por von Bertalanffy[1]. El segundo movimiento, bastante más práctico, se conoce con el nombre de *Ingeniería de Sistemas* iniciada por la investigación de operaciones y seguida por la administración científica y finalmente por el análisis de sistemas.

Así un sistema es un conjunto de partes coordinadas y en interacción para alcanzar un conjunto de objetivos.

Otra definición señala que un sistema es un grupo de partes y objetos que interactúan y que forman un todo o que se encuentran bajo la influencia de fuerzas en alguna relación definida [2].

El ser humano, por ejemplo, es un sistema con partes diferentes que a su vez está compuesto de subsistemas que contribuyen de distinta forma a mantener su vida, reproducción y acción.

1.1.2 Tipos de sistemas

Esta sección, se basa en parte, en el libro *Ingeniería de Análisis Estructurado Moderno* de Yourdon E. [4].

Los sistemas se dividen en dos categorías: *sistemas naturales* y *sistemas hechos por el hombre*; los cuales se describen a continuación:

Sistemas naturales

La mayoría de los sistemas no están hechos por el hombre: existen en la naturaleza y sirven a sus propios fines. Los sistemas naturales se dividen en dos subcategorías: sistemas físicos y sistemas vivientes. Los sistemas físicos incluyen ejemplos variados como:

- Sistemas estelares: galaxias, sistemas solares, etcétera.
- Sistemas geológicos: ríos, cordilleras, suelos, etcétera.
- Sistemas moleculares: organizaciones complejas de átomos.

Los sistemas vivientes comprenden todos los animales y plantas que nos rodean así, como a los seres humanos.

Sistemas hechos por el hombre

Un buen número de sistemas son construidos, organizados y mantenidos por los humanos, como:

- Sistemas sociales: organizaciones de leyes, doctrinas, costumbres.
- Sistemas de transporte: redes de carretera, canales, aerolíneas, buques cargueros.
- Sistemas financieros: contabilidad, inventarios, libro mayor, bolsa de valores.

Algunos de los sistemas humanos han sido automatizados y controlados por computadoras. Los sistemas automatizados tienen componentes en común: hardware, software, firmware, orgware, humanware, procedimientos y bases de datos.

Hardware.- Es el término aplicado a la computadora y sus componentes físicos. El *hardware* está constituido por uno o más dispositivos de entrada, dispositivos de salida, una unidad central de proceso (CPU) y uno o más dispositivos de almacenamiento.

Software.- Es el nombre genérico que se da a los programas que son utilizados en una computadora. El *software* es el elemento mediante el cual los usuarios utilizan la computadora para resolver problemas.

Firmware.- Se utiliza este término para denominar a cierta parte del *software* que las computadoras tienen grabado desde su fabricación, generalmente en memoria tipo *ROM* o en circuitos incorporados desde el diseño y fabricación de la máquina; el *firmware* generalmente consiste de una parte mínima del sistema operativo, rutinas de arranque, *set up*, microprogramación, circuitos sumadores o aritméticos, etcétera.

Orgware.- El término *orgware* se utiliza para señalar aspectos de organización arquitectónica y construcción de las computadoras, de sus componentes así como de otros elementos del área de sistemas; *orgware* estudia el diseño, el estándar de construcción, los materiales de fabricación, del equipo, del mobiliario y del área de trabajo.

Humanware.- El término *humanware* se utiliza para hacer referencia a diversos aspectos relacionados con el personal que interviene en el trabajo computacional; pueden ser aspectos administrativos o de estructura organizacional, de relaciones humanas, de métodos, procedimientos y estándares de trabajo, aspectos éticos, legales, trabajos en equipo, comportamiento social, etcétera.

Procedimientos.- Son las políticas que definen el uso específico de cada uno de los elementos o componentes del sistema y las reglas de su manejo y mantenimiento.

Base de Datos.- Son una gran colección de información organizada y enlazada al sistema a la que se accede por medio del Software.

Los sistemas se organizan de acuerdo a su aplicación: sistemas de manufacturas, sistemas de contabilidad, sistemas de defensa. Existe otra clasificación según su categoría: sistemas en línea, sistemas en tiempo real, sistemas de apoyo a decisiones, sistemas basados en el conocimiento.

1.1.3 Evolución del Software

En los primeros años de la informática, el hardware tenía una importancia mayor que en la actualidad. Su costo era más alto, su capacidad de almacenamiento así como el procesamiento y su fiabilidad, limitaban las prestaciones de un determinado producto. Al desarrollo se dedicaba poco esfuerzo y no se aplicaba ningún método sistemático. Actualmente el software es el elemento principal y ha desplazado en importancia al hardware. Esto ha hecho que las empresas se formulen las siguientes preguntas:

¿Por qué lleva tanto tiempo terminar los programas?

¿Por qué es tan elevado el costo?

¿Por qué no es posible encontrar todos los errores antes de entregar el software al cliente?

¿Por qué resulta tan difícil verificar el progreso conforme se desarrolla el software?

Éstas y otras preguntas son una manifestación de los mecanismos de programación, que han llevado a la aparición y la adopción paulatina de la ingeniería del software.

Retomando la definición de software, un concepto más integrado se refiere al:

(1) Conjunto de programas, instrucciones y lenguajes que permiten al sistema la ejecución de múltiples tareas[2];

(2) Estructuras de datos que facilitan a los programas manipular adecuadamente la información; y

(3) Documentos que describen la operación y el uso de los programas.

Por lo tanto, el software incluye no sólo los programas de las computadoras, sino también las estructuras de datos que manejan los programas y toda la documentación que debe acompañar al proceso de desarrollo y mantenimiento. El software se diferencia de otros productos por su propia naturaleza lógica [4]. En el desarrollo del hardware, el proceso creativo (análisis, diseño, construcción, prueba) se traduce finalmente en una forma material, en algo físico. Sin embargo, el software es inmaterial y por ello tiene características completamente distintas al hardware.

Existen similitudes en el proceso de desarrollo del software y del hardware. En ambos casos existen fases de análisis, diseño y desarrollo; la calidad del producto final se

obtiene mediante el diseño; en la fase de producción del software pueden generarse problemas que afecten a la calidad; el desarrollo es una de las labores de ingeniería, y la producción a gran o pequeña escala no influye en el impacto que tiene la ingeniería en el costo.

A partir de estas tareas, el software se divide en dos categorías:

Sistemas Operativos que controlan los recursos y administran los trabajos que lleva a cabo la computadora.

Programas de aplicación que llevan a cabo una multitud de tareas para los usuarios de las computadoras [5].

1.1.4 Bases de datos

Una Base de datos es un conjunto de archivos, datos, información; ordenada, organizada y relacionada, con la finalidad de permitir el manejo de la información para su procesamiento. Cada uno de los archivos representa una colección de registros y cada registro está compuesto de una colección de campos. Cada uno de los campos de cada registro permite llevar información de alguna característica o atributo de alguna entidad del mundo real.

El Sistema de Administración de Base de Datos (DBMS) es un conjunto de programas que se encargan de manejar la creación y todos los accesos a las bases de datos. Se compone de un Lenguaje de Definición de Datos (DDL), de un Lenguaje de Manipulación de Datos (DML) y de un Lenguaje de Consulta (SQL).

Sistema de Administración de Base de Datos (DBMS).

Es el nivel de software que permite el acceso a la información a un alto nivel de abstracción.

La secuencia conceptual de operaciones que ocurren para manipular cierta información que contiene una base de datos es la siguiente:

1. El usuario solicita información contenida en la base de datos.
 2. El DBMS intercepta este requerimiento y lo interpreta.
 3. El DBMS realiza las operaciones necesarias para introducir y/o actualizar la información solicitada
-

Administrador de la Base de Datos (DBA). Es la persona encargada de definir y controlar las bases de datos corporativas, además proporciona asesoría a los desarrolladores, usuarios y ejecutivos que la requieran, siendo responsable de:

El Administrador de Bases de Datos es responsable de:

a.- Administrar la estructura de la Base de Datos.

El DBA participa en el diseño inicial de la base de datos y su puesta en práctica así como en el control y administración de sus requerimientos, ayudando a evaluar alternativas, incluyendo los DBMS a utilizar y ayudando en el diseño general de la bases de datos.

b.- Administrar la Actividad de Datos.

El DBA administra la actividad de datos; protege los datos, no los procesa; el DBA proporciona estándares, guías de acción, procedimientos de control y la documentación necesaria para garantizar que los usuarios trabajen en forma cooperativa y complementaria al procesar datos en la bases de datos.

c.- Administrar el Sistema Manejador de Base de Datos.

La concurrencia de múltiples usuarios requiere la estandarización de los procesos de operación; el DBA es responsable de estas especificaciones y de asegurarse que lleguen a quienes concierne. El ámbito de la base de datos se rige por estándares, desde la forma de como se captura la información (tipo de dato, longitud, formato), hasta como es procesada y presentada.

d.- Establecer el Diccionario de Datos.

Cuando se definen estándares sobre la estructura de la base de datos, se deben de registrar en una sección del diccionario de datos a la que todos aquellos usuarios relacionados con ese tipo de proceso pueden acceder.

e.- Asegurar la Confiabilidad de la Base de Datos

Realizar un sistema de bases de datos lo suficientemente robusto, para evitar errores o usos inadecuados. Utilizar herramientas necesarias para la reparación de los posibles errores que las bases de datos presenten.

f.- Confirmar la Seguridad de la Base de Datos.

Alternativas utilizadas por el DBA para resolver o minimizar los problemas que se presentan:

Restringir el acceso a los procedimientos para ciertos usuarios.

Restringir al acceso a los datos para ciertos usuarios, procedimientos y/o datos.

Evitar la coincidencia de horarios para usuarios que comparten.

Los objetivos del administrador de la base de datos son:

Mantener la Integridad, seguridad y disponibilidad de los datos.

Las funciones básicas del administrador de bases de datos son las siguientes:

Crear las bases de datos y tablas.

Especificar las restricciones de integridad de los datos.

Administrar la concurrencia.

Optimizar el acceso a datos.

Definir el esquema conceptual.

Definir el esquema interno.

Vincularse con los usuarios.

Realizar procedimientos de respaldo y recuperación.

Supervisar el desempeño y responder a cambios en los requerimientos.

Conceder autorización para el acceso a los datos.

Definir el esquema.

Definir la estructura de almacenamiento del método de acceso.

1.2 Aspectos del Proceso de Desarrollo

El material presentado en esta sección está basado en el libro *Ingeniería de Sistemas de Software* de Gonzalo León Serrano [5].

El desarrollo de un producto de software es un desafío intelectual tanto para la organización en la que se desarrolla como para cada una de las personas que intervienen. Estos dos factores, humano y organizativo, se cubren en el proceso de gestación del producto. Producto y proceso se concentran en la atención de la Ingeniería de Software. El producto debe incorporar los requisitos que el usuario desea y es el

resultado final del desarrollo; del proceso de desarrollo depende el que esos requisitos sean realmente satisfechos en el producto final dentro de las restricciones de tiempo y costo establecido.

1.2.1 El Factor Humano

El desarrollo de software sigue siendo una actividad intensa en el capital humano. Más que en otras técnicas de desarrollo de sistemas, el desarrollo de software se basa en los equipos humanos de trabajo y, en menor medida, en inversiones materiales.

El objetivo principal de la ingeniería de software es el proceso de desarrollo que integra a un equipo numeroso de personas en el desarrollo de software cuyas actividades están ligadas a los componentes del equipo de trabajo esbozadas de acuerdo a un perfil técnico especializado.

Cada perfil técnico requiere conocimientos asociados a las actividades relacionadas con ese perfil en el proceso de desarrollo. Simultáneamente, el perfil también implica la existencia de unas capacidades de comunicación con otras personas de acuerdo a intercambios de información y protocolos de cooperación entre ellas que deberán estar bien definidos.

El concepto de perfil técnico ha ido evolucionando. Tradicionalmente, se empleaban: analista, diseñador, programador, jefe de proyecto; todos ellos ligados a las etapas en las que se dividía el desarrollo de un determinado sistema. Cuando la actividad ligada a un perfil terminaba su trabajo entonces entraba el siguiente. Era un modelo basado en la división vertical del trabajo.

Los problemas derivados del modelo de desarrollo clásico así como la disponibilidad de nuevas tecnologías de desarrollo, están haciendo obsoletos algunos de los perfiles convencionales. Esta evolución tiende a que cada componente del equipo de trabajo posea una visión más amplia del proceso de desarrollo aunque limitada a una perspectiva concreta sobre el sistema en construcción.

Esta situación implica la aparición de un modelo horizontal de división del trabajo que se superpone parcialmente con el vertical. A cada componente del equipo de trabajo se le va a exigir un conocimiento mayor sobre la totalidad del desarrollo aunque su actividad enfatice aspectos concretos.

1.2.2 La Organización

Las actividades no estrictamente ligadas al desarrollo técnico del producto sino a la forma en la que éste se obtiene, recaen en la estructura de la organización. Un conjunto de actividades, orientadas a un fin concreto, empleando y generando determinada información relativa al sistema en desarrollo se denomina **proceso**. Para cada uno de los **procesos** se requiere fijar actividades, responsables, entradas y salidas, planificación temporal de recursos necesarios y mecanismos para asegurar que se realiza correctamente.

Existen diversas maneras de organizar un desarrollo, debiendo adaptar los procesos necesarios al tipo de software que se desea desarrollar, a la duración del proyecto, o al uso futuro del modelo de desarrollo. Las organizaciones aprenden de los desarrollos actuales y van madurando, las empresas dedicadas al desarrollo de sistemas de software no tienen definidos ni controlados todos los procesos implicados.

1.3 Documentación y Mantenimiento

La documentación tiene un papel importante en todo proyecto de software, pero en la práctica se le presta poca atención, ya sea porque no se la genera o porque no se la utiliza. Sin embargo, su necesidad es evidente. Lo importante no es documentar, lo importante es disponer de la información necesaria cuando hace falta. La documentación es el nexo entre las distintas actividades casi simultáneas que se realizan, y que permite ejercer el control efectivo del proyecto.

La documentación persigue objetivos internos y externos. No sólo documentamos para comunicarnos entre los miembros del grupo de trabajo, sino también para comunicarnos con el usuario del programa respecto a las decisiones del equipo sobre el producto.

Cuando no se documenta resulta difícil modificar con rapidez el procesamiento de una aplicación pues no se tiene idea de donde ocurre el proceso del sistema. Para solucionar este problema, la dirección del proyecto debe imponer normas de documentación y exigir que el equipo de desarrollo se ajuste a ellas.

1.3.1 Documentación

La documentación no es una fase del proyecto, es un subproducto. En la medida en que se debe comunicar algo, se crea un documento.

Los estándares para la documentación se clasifican en seis tipos:

- a) Diseño conceptual.
- b) Especificación funcional.
- c) Especificación del sistema.
- d) Especificación del programa.
- e) Manual de operaciones.
- f) Manual de usuario.

Debe haber al menos tres productos identificados de la documentación:

MANUAL DE OPERACIONES: Los manuales de operaciones son los menos genéricos de los documentos del sistema, dado que su formato y contenido varían según la estructura organizacional del departamento de operaciones, el hardware instalado, el software del sistema y la naturaleza de la aplicación

MANUAL DE USUARIO: Este manual debe redactarse de modo de que todos los que interactúen con el sistema puedan entender fácilmente el rol que les toca desempeñar en la operación.

La documentación para el usuario debe incluir una descripción general de las tareas que realiza el programa, así como una descripción detallada de todas las instrucciones que sean necesarias para su instalación, puesta en marcha y funcionamiento; así como consejos, recomendaciones de uso, explicación de los mensajes de error y modo de solucionarlos, entrada y salida de datos, menú de opciones. El manual de usuario debe cubrir al menos los siguientes puntos:

- a) Instrucciones para ejecutar el programa en memoria, desde el almacenamiento en disco y arrancar su funcionamiento.
 - b) Nombres de los archivos externos a los que accede el programa.
 - c) Formato de todos los mensajes de error o informes.
 - d) Opciones en el funcionamiento del programa.
 - e) Descripción detallada de la función realizada por el programa.
-

- f) Descripción detallada, preferiblemente con ejemplos, de cualquier salida producida por el programa.

MANUAL DEL PROGRAMADOR: cada programa del sistema debe tener un documento correspondiente de especificación. Si las especificaciones del programa se preparan correctamente, un programador sin previo conocimiento podrá realizar los cambios.

Las secciones que debe agregar al manual el programador son:

- a) El algoritmo de resolución en forma de diagrama de flujo.
- b) Diagrama de estructuras o diagramas de bloques.
- c) Flujo de pantallas.
- d) Listado del programa fuente.
- e) Diseño de las pruebas.
- f) Manual de mantenimiento con las restricciones y limitaciones e información de operaciones.

El programador describe el funcionamiento y uso del programa en una documentación técnica y de usuario.

La documentación de un programa debe ser de dos tipos: documentación para personal informático y documentación para usuario.

Existen tres grupos de personas que necesitan conocer la documentación del programa: programadores, operadores y usuarios.

- a) Programadores: manual de mantenimiento del programa.
- b) Operadores: manual del operador.
- g) Usuario: manual del usuario

1.3.2 Mantenimiento

La actividad más cara en la vida del software es el mantenimiento, en la cual se repara, perfecciona o extiende un producto de software. Esto la convierte, además, en la etapa más importante de la vida de un producto de software. El mantenimiento correctivo comienza tras la detección de errores en el proceso de verificación. La tarea básica es la depuración del código.

Los procesos que tienen lugar durante la depuración son:

- a) Reproducir el error.
- b) Diagnosticar la causa.
- c) Corregirla.
- d) Verificar la corrección realizada.

La alteración de un sistema implica grandes riesgos. Una modificación es siempre una corrección.

1.4 Ingeniería del Software

En general, es más conocida la historia del desarrollo del hardware que de la evolución del software, sin embargo, los desarrollos en la ingeniería del software han sido igualmente importantes.

1.4.1 Evolución de la Ingeniería del Software.

La Ingeniería de Software, como ya se menciona al principio del capítulo, es el establecimiento y uso de principios de ingeniería para desarrollar software confiable [7].

La Ingeniería de Software con cuatro décadas de existencia, inicia con las Técnicas de Programación Estructurada, la cual fue seguida por otros métodos estructurados de análisis y de diseño. Se han desarrollado varios métodos para hacer modelos de sistemas de información.

Originalmente el desarrollo de sistemas de programación no se iniciaba con una exhaustiva fase de planeación y definición de requerimientos. Se trataba de producir sistemas sin considerar que éstos, en un futuro, deberían integrarse en un sistema generalizado de información. Los problemas de integración hacían que el sistema, en el mejor de los casos, trabajara ineficientemente. Se olvidaba que para poder planear en forma adecuada un sistema de gran tamaño, es necesario emplear el enfoque sistémico, que es una evolución del método científico. Aquí se indica que la solución de un problema consiste en dar respuesta a las siguientes preguntas:

- ¿Cuál es el problema?
 - ¿Cuáles son las alternativas?
 - ¿Cuál es mejor?
-

El objetivo de la metodología de sistemas es precisamente desarrollar programación con la misma filosofía con la que se diseña un sistema telefónico. El costo cada vez creciente de programación hace necesario seguir un método sistémico para su construcción y no uno de prueba y error, como se ha realizado hasta la fecha.

En la década de los 60's y 70's se realiza una multiprogramación y los sistemas comienzan a ser comercializados por empresa, ampliándose los mercados y los potenciales del software. A fines de los 70's aparece el procesamiento distribuido, haciendo más complejos los sistemas. Aparecen las computadoras personales, aumentando el número de usuarios y ejerciendo una presión hacia la optimización de los flujos de información

El desarrollo de software orientado a objetos comenzó en los años ochenta como una etapa natural de los métodos estructurados, que van desplazando a los lenguajes de tercera generación.

Actualmente, la tecnología de componente es un método para desarrollar sistemas de información que está creciendo. A diferencia de los métodos tradicionales, la tecnología de componente los ensambla para formar una solución de software. Existen hoy día dos estándares de componentes en competencia: JavaBeans de Sun y DCOM de Microsoft®. Los componentes de software son materiales reusables para construir sistemas de software.

La Tecnología Basada en Componentes es un método poderoso para la empresa de la ingeniería de sistemas de información porque es una tecnología que está reduciendo el conflicto entre sistemas de alta complejidad y la búsqueda para la alta calidad y la productividad.

Para ciertos sistemas de información críticos, el uso de métodos formales es recomendado en el ciclo de vida del software, para producir sistemas con la integridad más alta. Los métodos formales confían en las técnicas matemáticas que expresan y modelan los requisitos de cualquier producto en el ciclo vital del software.

Capítulo 2. Ingeniería del Software.

En este capítulo se definen los conceptos fundamentales de la Ingeniería de Software y sus diversas aplicaciones, que incluyen modelos y prototipos.

El software es un elemento básico en la sociedad actual como generador de servicios. Sus costos de desarrollo y, en muchos casos, de adquisición son cada vez más altos respecto al hardware y paulatinamente envejece sin necesidad de estropearse, a pesar de ser una gran paradoja en productos sin partes mecánicas y con un costo de replicación nulo.

2.1 Ingeniería de software

Según la definición del IEEE¹, "**software** es la suma total de los programas de computadora, procedimientos, reglas, documentación asociada y los datos que pertenecen a un sistema de cómputo"[6]. En el mismo documento se afirma que, "un producto de software es un producto diseñado para un usuario". En este contexto, la Ingeniería de Software es un enfoque sistemático del desarrollo, operación, mantenimiento y retiro del software", en otras palabras, se considera que "**la Ingeniería de Software** es la rama de la ingeniería que aplica los principios de la ciencia de la computación y las matemáticas para lograr soluciones costo-efectivas a los problemas de desarrollo de software", es decir, "permite elaborar consistentemente productos correctos, utilizables y costo-efectivos" [7].

El proceso de ingeniería de software se define como "un conjunto de etapas parcialmente ordenadas con la intención de lograr un objetivo, en este caso, la obtención de un producto de software de calidad" [8]. El **proceso de desarrollo de software** "es aquel en que las necesidades del usuario son traducidas en requerimientos de software, estos requerimientos transformados en diseño y el diseño implementado en código, el código es probado, documentado y certificado para su uso operativo". Concretamente "define quién está haciendo qué, cuándo hacerlo y cómo alcanzar un cierto objetivo" [9].

El proceso de desarrollo de software requiere por un lado un conjunto de conceptos, una metodología y un lenguaje propio. A este proceso también se le llama el ciclo de vida

¹Instituto de Ingenieros en Electricidad y Electrónica, Inc. (por sus siglas en ingles)

del software que comprende cuatro grandes fases: concepción, elaboración, construcción y transición. La concepción define el alcance del proyecto y desarrolla un caso de negocio. La elaboración define un plan del proyecto, especifica las características y fundamenta la arquitectura. La construcción crea el producto y la transición transfiere el producto a los usuarios.

2.2 Ciclo de Vida

A partir de los años setenta, se había acumulado suficiente experiencia sobre el desarrollo de software para identificar un conjunto de actividades comunes a todos los proyectos. Asimismo, la estructura organizativa requerida para controlar su ejecución y la forma en la que el control de calidad aseguraba la validez de un determinado producto eran suficientemente conocidas como para definir marcos de referencia utilizables en nuevos proyectos de desarrollo. Esta idea desembocó en el concepto de **ciclo de vida** de software

Por ciclo de vida de software se entiende el proceso de **fases**, **actividades** en cada una de las fases, **controles** para pasar de una fase a otra y **resultados** generados en cada una de ellas; éstos permiten el desarrollo de un producto desde su concepción, la entrega al usuario y en evolución posterior.

2.2.1 Elementos del Ciclo de Vida

Un ciclo de vida para un proyecto se compone de fases sucesivas compuestas por tareas planeadas. Según el ciclo de vida, la sucesión de fases puede ampliarse con ciclos de retroalimentación, de manera que lo que conceptualmente se considera una misma fase se pueda ejecutar más de una vez a lo largo de un proyecto, recibiendo en cada pasada de ejecución aportaciones de los resultados intermedios que se van produciendo (realimentación). La Figura 2.1 ilustra los elementos del ciclo de vida.

Para un adecuado control de la progresión de las fases de un proyecto se hace necesario especificar con suficiente precisión los resultados evaluables, o sea, productos intermedios que deben resultar de las tareas incluidas en cada fase. Normalmente estos productos marcan los hitos entre fases.

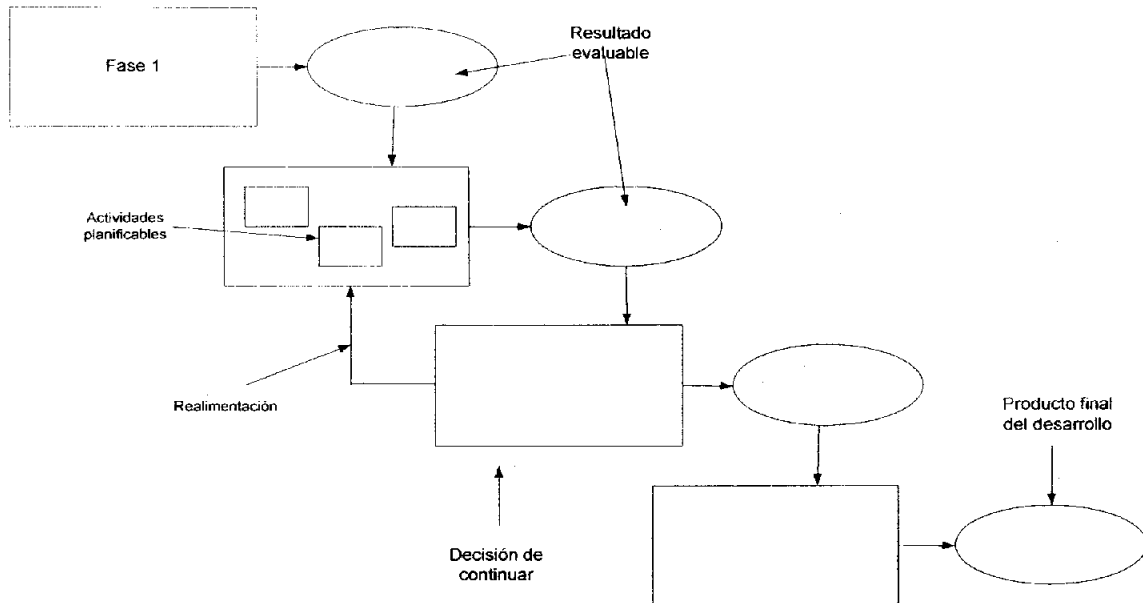


Figura 2.1. Elementos del Ciclo de Vida

- **Fases.** Son un conjunto de actividades relacionadas con un objetivo en el desarrollo de software. Se construye agrupando tareas (actividades elementales) que pueden compartir un tramo determinado del tiempo de vida de un proyecto. La agrupación temporal de tareas impone requisitos temporales correspondientes a la asignación de recursos ya sea humanos, financieros o materiales. Cuanto más grande y complejo sea un proyecto, mayor detalle se necesitará en la definición de las fases para que el contenido de cada una siga siendo manejable. De esta forma, cada fase de un proyecto puede considerarse un "micro-proyecto" en sí mismo, compuesto por un conjunto de micro-fases.

La descomposición de una fase en subfases menores permite separar partes temporales del proyecto que se subcontraten a otras organizaciones, requiriendo distintos procesos de gestión, como se ilustra en la Figura 2.2.

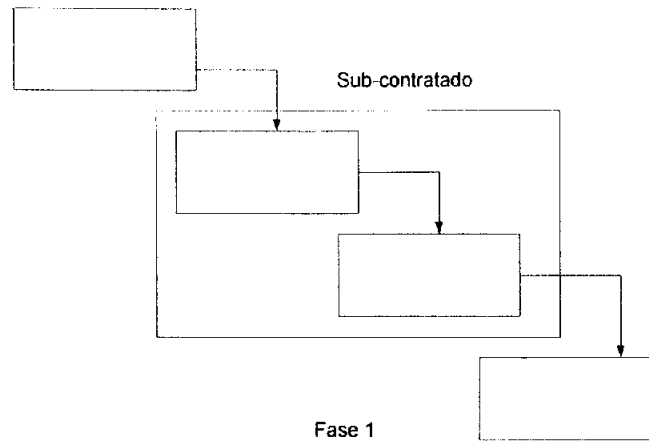


Figura 2. 2. Descomposición de una fase en subfases.

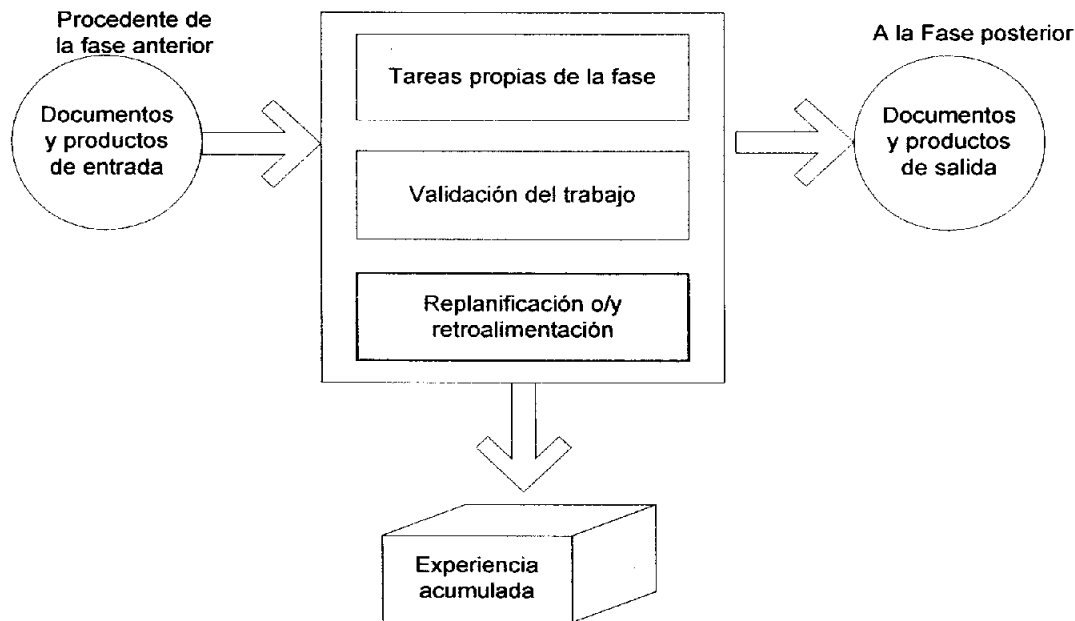


Figura 2. 3. Esquema general de operación de una fase

Cada fase está definida por un conjunto de elementos observados externamente, como son las **actividades** con las que se relaciona, los **datos de entrada**, **datos de salida** y la **estructura interna** de la fase, como pueden verse en la Figura 2.3.

- **Productos intermedios.** Son el resultado en las fases. Pueden ser materiales (componentes, equipos) o inmateriales (documentos, software). Los productos intermedios permiten evaluar la marcha

del proyecto mediante comprobaciones de su adecuación o no a los requisitos funcionales y de condiciones de realización previamente establecidos.

2.2.2 Tipos de Ciclo de Vida

No existe un ciclo de vida único; tanto el tipo, orden y actividades en cada fase pueden cambiar adaptándose a las necesidades del producto a realizar y a la propia estructura de la organización que lo desarrolla a partir de las posibilidades que ofrece la tecnología de software empleada.

Las principales diferencias entre los distintos modelos de ciclo de vida están en:

- El alcance del proyecto, el cual comprende un simple estudio de viabilidad del desarrollo de un producto, su desarrollo completo o, incluso, toda la historia del producto, desde la fabricación y modificaciones posteriores hasta su retirada del mercado.
- Las características (contenidos) del proyecto (no es lo mismo proyectar un avión que un puente) o de la organización.
- La estructura de la sucesión de las fases puede ser:
 - De vida lineal.
 - Cascada o convencional.
 - Prototipos.
 - Incremental.
 - Espiral.
 - Síntesis automática.
 - Iterativo.

Ciclo de Vida Lineal

Consiste en descomponer la actividad global del proyecto en fases que se suceden de manera lineal, es decir, cada una se realiza una sola vez, progresivamente una tras otra como se ilustra en la Figura 2.4 Con un ciclo lineal es fácil dividir las tareas entre equipos sucesivos y prever los tiempos, sumando los de cada fase.

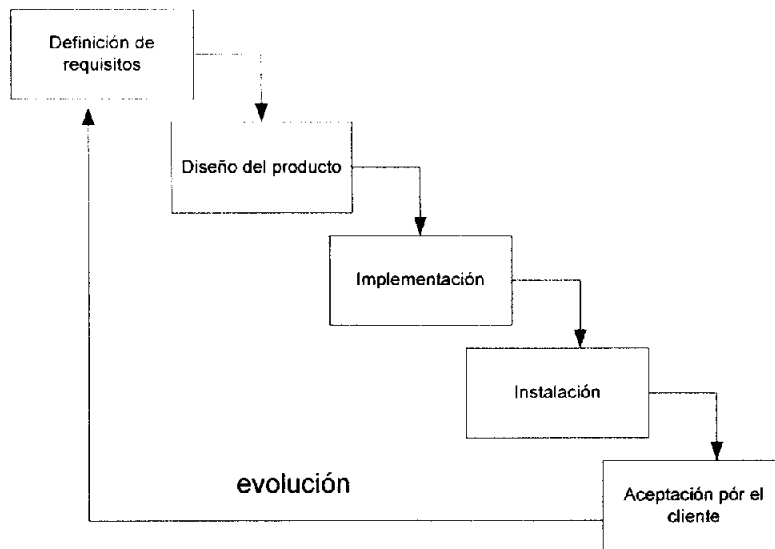


Figura 2. 4. Ciclo de Vida Lineal

Las principales fases a considerar son:

- a. Definición de requisitos.
- b. Diseño.
- c. Implementación.
- d. Instalación.

En este proceso se requiere que cada actividad del proyecto pueda descomponerse de manera que una fase no necesite resultados de las siguientes (retroalimentación), aunque pueden admitirse ciertos supuestos de retroalimentación correctiva. Desde el punto de vista de la gestión, para decisiones de planificación, requiere también que se sepa de antemano lo que va a ocurrir en cada fase antes de empezarla.

a.- Definición de requisitos.

El objetivo de esta fase es entender el problema a resolver, extraer las necesidades del usuario y derivar de ellas las funciones que debe realizar el sistema, la fase se subdivide en: análisis de requisitos del usuario y análisis de requisitos del sistema.

La subfase de análisis de requisitos del usuario tiene como objetivo conocer las necesidades de los usuarios y cuáles deben ser los servicios que un sistema de software debe ofrecerle para satisfacerlas. La fase implica la creación del Documento de Requisitos de Usuario (DRU) que constituye el documento base para que, al final del desarrollo, el sistema sea aceptado por el usuario.

La subfase de análisis de requisitos del sistema consiste en la construcción de un modelo lógico del sistema de software describiendo las funciones que sean necesarias y las relaciones entre ellas suponiendo que no existen limitaciones de recursos.

Por modelo lógico se entiende la identificación de las funciones de software requeridas para satisfacer los requisitos del usuario. Esta identificación se suele realizar en varios niveles de detalle hasta llegar a uno en el que las funciones identificadas estén suficientemente claras de tal forma que no exija un refinamiento.

El producto generado en esta fase es el Documento de Requisitos de Software (DRS). Así mismo, se genera en esta subfase el plan de gestión del desarrollo con estimaciones de costos y recursos más ajustados que en la subfase anterior.

b.- Diseño.

La fase de diseño tiene como objetivo determinar una solución para los requisitos del sistema definidos en la fase anterior. Existen muchas maneras de satisfacer los requisitos y, por tanto, multitud de diseños posibles.

Es conveniente distinguir entre diseño de alto nivel o arquitectónico y diseño detallado. El primero de ellos tiene como objetivo definir la estructura de la solución identificando grandes módulos y sus relaciones. El segundo define los algoritmos empleados y la organización del código para comenzar a instrumentar.

En la subfase de diseño arquitectónico se parte del modelo lógico generado en la fase de definición de requisitos software y se transforma en una arquitectura agrupando las funciones identificadas en componentes de software, así mismo, se define la activación y desactivación de cada uno de los componentes y el intercambio de información entre ellos. El resultado de esta fase es el Documento de Diseño Arquitectónico (DDA).

Definir una buena arquitectura del sistema constituye un elemento básico para asegurar que el sistema sea integrable con otros. Al final de la fase, se genera el documento de Diseño Detallado (DDA).

c.- Implementación.

Su objetivo es producir una solución eficiente en un lenguaje que ejecute las decisiones adoptadas en la fase de diseño. Incluye la codificación y la prueba del sistema hasta obtener un paquete ejecutable sobre la plataforma requerida por el usuario.

En esta fase de implementación se selecciona y utiliza un lenguaje de programación determinado; el conocimiento del lenguaje de implementación orienta en la fase de diseño, relacionando de forma más directa los objetos o módulos identificados con las construcciones del lenguaje.

Una vez obtenida la implementación del sistema se prueba que satisface los requisitos definidos inicialmente. Es necesario realizar pruebas a diferentes niveles hasta que el sistema en su conjunto sea aceptado por el usuario. Al final de la fase, se genera el Manual de Usuario junto con el código fuente del sistema y las pruebas asociadas.

d.- Instalación

La fase de transferencia del producto tiene como objetivo instalar el sistema de software desarrollado en el entorno del cliente y realizar las pruebas de aceptación necesarias. El proceso de transferencia implica un periodo largo en el que se incluye la formación del usuario en el producto y la realización de las pruebas de aceptación junto con el usuario.

En la fase de transferencia se continúan las actividades de prueba iniciadas durante la implementación con la colaboración del usuario.

Se genera en esta fase el documento de Historia del Proyecto que resume las lecciones aprendidas y de cuyo análisis se pueden extraer conclusiones para la mejora de los procesos de desarrollo en futuros proyectos.

Una vez que el software está funcionando en forma regular no es de ningún modo un sistema inmutable. Todo software debe adaptarse a un entorno que va cambiando ya que un software que no evoluciona va haciéndose cada vez menos útil.

La evolución del sistema de software se incluye dentro de la fase de mantenimiento aunque su implicación es mucho más amplia de lo que el término significa en otras ingenierías

Existen tres tipos de mantenimiento:

1. Mantenimiento correctivo, donde se eliminan los problemas surgidos durante la fase de operación del sistema y que no han sido detectados anteriormente.
2. Mantenimiento preventivo, que mejora la funcionalidad del sistema ya sea en relación con la eficiencia en ejecución del mismo (menor tiempo de respuesta, optimización del uso de la memoria), o para facilitar su uso.
3. Mantenimiento evolutivo, que modifica, amplía, elimina o sustituye la funcionalidad del sistema para adaptarla a las nuevas necesidades del usuario o con el objetivo de adaptarlo a nuevas interfaces de hardware o software.

Ciclo de Vida Incremental

En el modelo de ciclo de vida en cascada se obtiene el código como resultado de un proceso de refinamiento a partir de las especificaciones y diseño. Cuando el código es entregado al usuario se dispone de un producto que debe satisfacer sus necesidades. En caso de que esto no sea así, es necesario volver a las fases anteriores.

Existe otro enfoque en donde al usuario se le van exponiendo productos intermedios denominados prototipos que se acercan al sistema final, los cuáles sirven para validar con el usuario el sistema que se está construyendo, antes de realizar la implementación.

Un prototipo puede definirse como un modelo parcial ejecutable de un sistema de software. Por modelo del sistema se entiende una descripción del sistema bajo una cierta perspectiva (por ejemplo arquitectura, datos, comportamiento), empleando notaciones no necesariamente similares al código final; es parcial porque no es necesario que cubra todo el sistema sino aquellas partes o perspectivas que se pretenden analizar; finalmente, debe ser ejecutable para que la validación del sistema pueda hacerse a partir de la experimentación con el prototipo por parte de usuarios y analistas.

Al ser parcial, se debe conocer qué funcionalidad debe ser incluida en el prototipo. Para ello existen dos enfoques básicos: construcción de un prototipo vertical en el que se presta atención a una parte pequeña del sistema pero prácticamente en su forma final, y

prototipo horizontal, en el que la idea es obtener una visión global del sistema sin detallar o refinar ninguna de sus partes.

El prototipo vertical es importante cuando la especificación de requisitos demuestra problemas en conocer qué es lo que se desea sobre un aspecto muy concreto. El prototipo horizontal, por el contrario, pretende conocer mejor la estructura general de la interacción entre el usuario y el sistema de software a diseñar. Ambos son complementarios y pueden aparecer en un caso concreto de aplicación de las técnicas de prototipos.

También la técnica de prototipos puede clasificarse en función del uso que se va a hacer del prototipo a lo largo del desarrollo. Existen dos tipos básicos de uso del prototipo: modelo basado en prototipos desechables y modelo de prototipos incremental.

Modelo basado en prototipos desechables

El modelo con prototipo desechable -Figura 2.6- aborda el problema de la inestabilidad de los requisitos generando un prototipo lo antes posible, que sirva de base al mejor conocimiento de los requisitos de usuario. Este prototipo es desechado cuando los usuarios y desarrolladores acuerdan un documento de requisitos de usuario. Este modelo apoya a la fase de análisis de requisitos tanto de usuario como de sistema, permitiendo confiar que los requisitos identificados son los que realmente desea el usuario y reducir el riesgo de problemas en la aceptación final del sistema de software.

De la Figura 2.5 se concluye que el prototipo se entiende como una mejora de la fase de análisis de requisitos con objeto de conocer mejor éstos y ayudar a usuarios y diseñadores a su definición. Este modelo es útil para detectar **requisitos ocultos** y eliminar inconsistencias entre ellos.

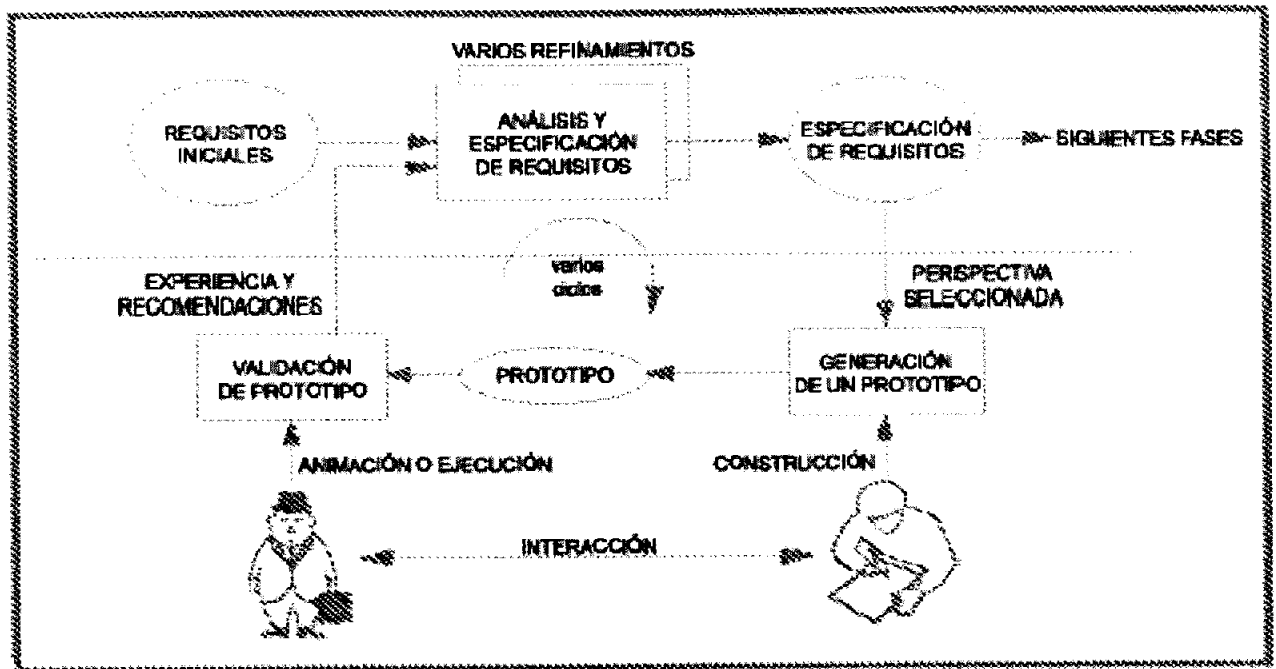


Figura 2. 5 Ciclo de Vida con Prototipos Desechables

Modelo basado en prototipo incremental

Este modelo se basa en la generación de varios modelos parciales ejecutables del sistema antes de proceder a la implementación (durante la especificación y el diseño) con el fin de evaluar sus características y obtener al final el sistema implementado.

Un modelo de especificación es una descripción de las funciones identificadas en el sistema y sus interrelaciones usando una notación gráfica. En un modelo de diseño la descripción se completa incluyendo información sobre la descomposición de tareas y el uso de los recursos del sistema operativo.

Los modelos ejecutables son *prototipos intermedios* generados para realizar análisis sobre ellos. El concepto de ejecución en este caso es el de animación del modelo; por ello, entendemos la visualización dinámica de la evolución del mismo. Algunas herramientas de software, generan automáticamente interfaces gráficas con las que el usuario interacciona. Una vez que usuarios y analistas aceptan el sistema se continúa el desarrollo hacia las siguientes fases. La Figura 2.6 presenta esquemáticamente el desarrollo de acuerdo con este modelo.

Una de las posibilidades que este proceso de ejecución de modelos tiene es la descripción a diferentes niveles, de la abstracción, combinando partes que están definidas a un nivel de detalle cercano al código (o incluso código) con otras que aún corresponden a modelos más abstractos. Un prototipo que combina todas ellas en un sistema ejecutable se denomina prototipo heterogéneo. A la técnica de desarrollo basada en la construcción de prototipos heterogéneos se le denomina prototipo incremental.

La Figura 2.7 presenta esquemáticamente la estructura de un prototipo heterogéneo, combinando modelos de implementación (código) con modelos de diseño y otros que mezclando modelos de diseño con modelos de especificación.

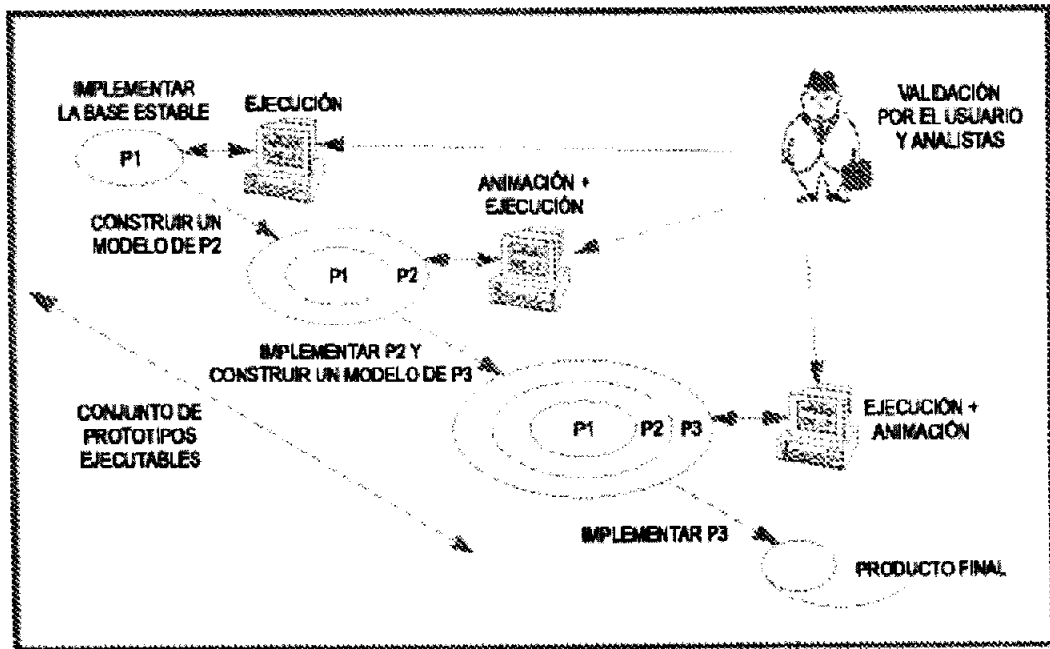


Figura 2. 6. Ciclo de Vida Incremental

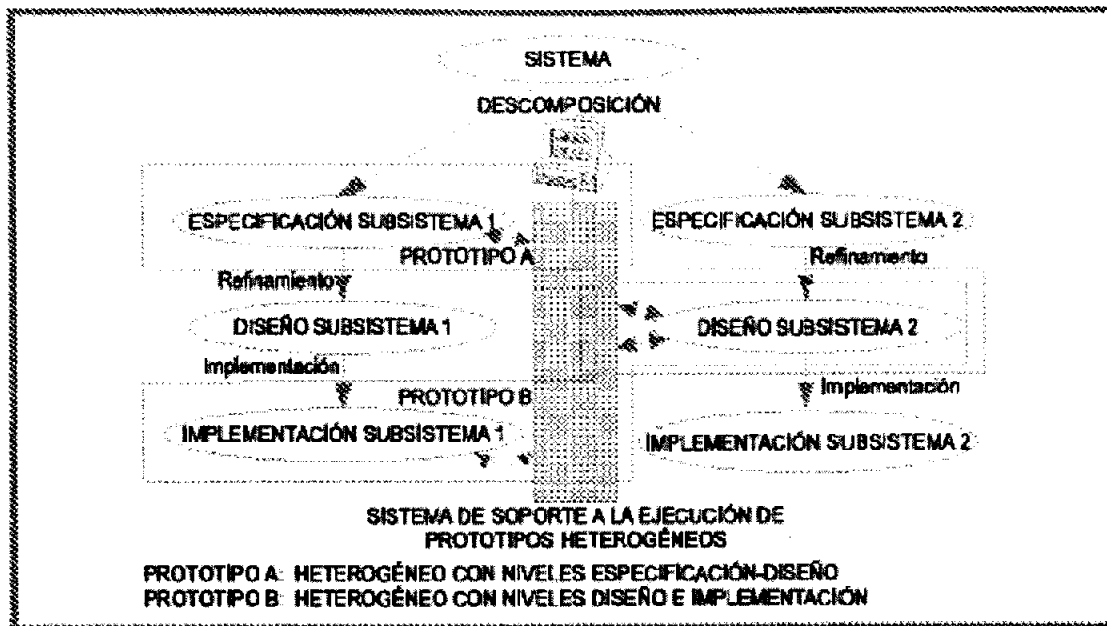


Figura 2. 7. Prototipo Heterogéneo

Las ventajas del modelo incremental se resumen de la siguiente manera:

- Permite incrementar la visibilidad del proceso de desarrollo mediante la experimentación con prototipos ejecutables intermedios.
- La animación de modelos gráficos permite entender el comportamiento dinámico del sistema y las interfaces hombre-máquina.
- La documentación de las fases de análisis y diseño queda reforzada por los resultados del proceso de animación de los modelos facilitando las pruebas de aceptación.
- No se "tira" nada. Los modelos realizados siguen siendo empleados en el siguiente prototipo o se convierten en un modelo con un nivel de mayor detalle, el cuál puede ser incluso código.

Ciclo de Vida en Espiral

El ciclo de vida en espiral puede considerarse como una generalización del ciclo de vida con incremental, para los casos en que no basta con una sola evaluación de un prototipo que permita disminuir o desaparece la incertidumbre. El propio producto a lo largo de su desarrollo puede así considerarse como una sucesión de productos que progresan hasta llegar a alcanzar el estado deseado. En cada ciclo (espiral) las especificaciones del producto se van resolviendo paulatinamente.

Por lo regular la fuente de incertidumbres es el propio cliente, que aunque sepa en términos generales lo que quiere, no es capaz de definirlo en todos sus aspectos sin ver como unos influyen en otros. En estos casos la evaluación de los resultados por el cliente no puede esperar a la entrega final y puede ser necesaria repetidas veces.

El esquema del ciclo de vida se representa por un ciclo en espiral, donde los cuadrantes son, habitualmente, fases de especificación, diseño, realización y evaluación (o conceptos y términos análogos).

En cada vuelta el producto gana en "madurez" (aproximación al final deseado) hasta que en una vuelta la evaluación lo apruebe y el bucle pueda abandonarse.

La Figura 2.8 representa esquemáticamente las fases seguidas en el desarrollo de Ciclo de Vida en Espiral.

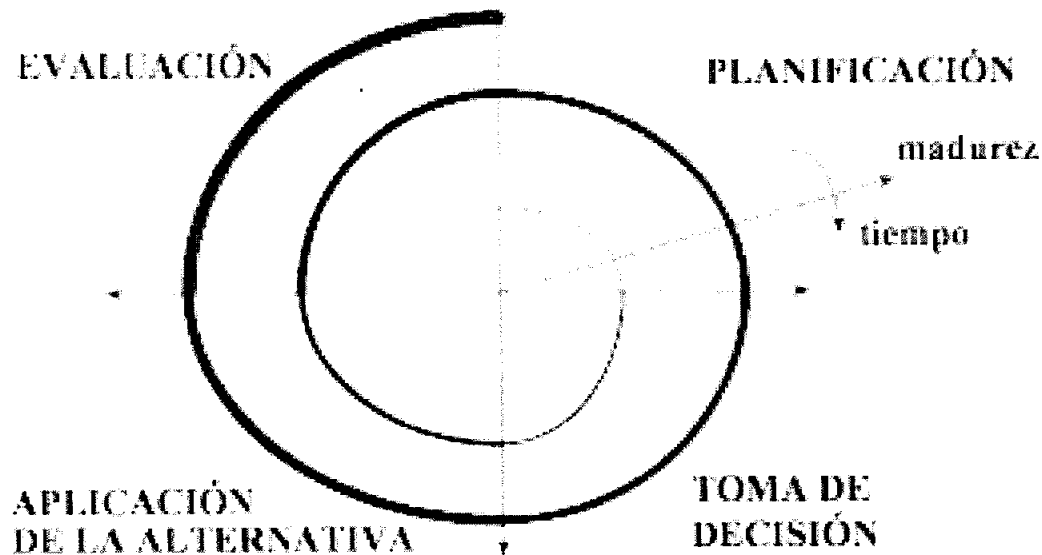


Figura 2. 8. Ciclo de Vida en Espiral

Capítulo 3. UML

En este capítulo se hace referencia al desarrollo con UML.

UML es un lenguaje de modelado unificado y no un método. Se basa en las especificaciones de Booch, Rumbaugh y Jacobson [8]. Modela un proyecto con un número de diagramas que representan las diferentes vistas. Estos diagramas juntos son los que representan a la arquitectura del proyecto.

Cada diagrama usa la anotación pertinente y la suma de estos diagramas crean las diferentes vistas. Las vistas existentes en UML son:

- Vista casos de uso: Se forma con los diagramas de casos de uso, colaboración, estados y actividades.
- Vista de diseño: Se forma con los diagramas de clases, objetos, colaboración, estados y actividades.
- Vista de procesos: Se forma con los diagramas de la vista de diseño.
- Vista de implementación: Se forma con los diagramas de componentes, colaboración, estados y actividades.
- Vista de despliegue: Se forma con los diagramas de despliegue, interacción, estados y actividades

Se dispone de dos tipos diferentes de diagramas, los que dan una vista estática del sistema y los que dan una visión dinámica. Los diagramas estáticos son:

- Diagrama de clases: muestra las clases, interfaces, colaboraciones y sus relaciones.
- Diagrama de objetos: es un diagrama de instancias de las clases mostradas en el diagrama de clases. Muestra las instancias y como se relacionan entre ellas.
- Diagrama de componentes: muestran la organización de los componentes del sistema. Un componente se corresponde con una o varias clases, interfaces o colaboraciones.
- Diagrama de despliegue: muestra los nodos y sus relaciones. Un nodo es un sistema informático o computadora.

- Diagrama de casos de uso: Muestran los casos de uso, actores y sus relaciones. Muestra quien puede hacer qué y qué relaciones existen entre acciones (casos de uso).

Lo diagramas dinámicos son:

- Diagrama de secuencia y diagrama de colaboración: muestran a los diferentes objetos y las relaciones que pueden tener entre ellos, los mensajes que se envían entre ellos. Son dos diagramas equivalentes, que se puede pasar de uno a otro sin pérdida de información, pero que nos dan puntos de vista diferentes del sistema.

- Diagrama de estados: muestra los estados, eventos, transiciones y actividades de los diferentes objetos.
- Diagrama de actividades: es un caso especial del diagrama de estados. Muestra el flujo entre los objetos. Se utilizan para modelar el funcionamiento del sistema y el flujo de control entre objetos.

3.1 Diagramas recomendados

Los diagramas a representar dependerán del sistema a desarrollar, para ello se efectúan las siguientes recomendaciones dependiendo del sistema.

Aplicación monousuario

- Diagrama de casos de uso.
 - Diagrama de clases.
 - Diagrama de interacción.
 - Aplicación monousuario, con registro de eventos:
 - Añadir: diagrama de estados.
 - Aplicación cliente servidor:
 - Añadir: diagrama de despliegue y diagrama de componentes, dependiendo de la complejidad.
 - Aplicación compleja distribuida:
 - Todos.
-

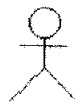
3.2 Diagrama de casos de uso

Esta sección, se basa en parte, en el libro *Ingeniería de software para principiantes*, Ibarguengoitia G., Oktaba H [10].

3.2.1 Elementos de los diagramas de casos de uso

El diagrama de casos de uso de UML describe el comportamiento funcional del sistema. Representa la forma como un usuario (Actor) operará con el sistema, aclara la funcionalidad del sistema.

Un diagrama de casos de uso esta formado por los siguientes elementos:



Actor: Es cualquier cosa externa que intercambia información con el sistema, puede ser un usuario u otro sistema. Se representa con una figura humana con el nombre del actor en singular



Casos de uso: es una funcionalidad o tarea que realiza algún actor para obtener un valor o servicio a un usuario. El nombre del caso de uso deberá nombrarse, con un verbo en infinitivo.

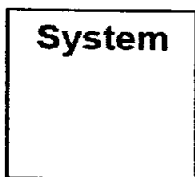
Relaciones



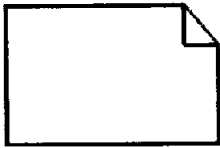
Asociación: Es el tipo de relación más básica que indica la invocación de un actor a una funcionalidad.



Generalización: Se usa para actores o casos de uso, especializa el comportamiento del actor o caso de uso.



Alcance del sistema: contiene los casos de uso que se realizan en cada ciclo de desarrollo definiendo el alcance o frontera del sistema. Se representa por un rectángulo que contiene los casos de uso dentro del alcance del sistema



Notas: Se ponen los comentarios para explicar el diagrama de casos de uso.

3.2.2 Modelado del contexto

Se debe modelar la relación del sistema con los elementos externos, ya que son estos elementos los que forman el contexto del sistema. Los pasos a seguir son:

- Identificar los actores que interactúan con el sistema.
- Encontrar casos de uso para cada actor.
- Organizar a los actores.
- Especificar sus vías de comunicación con el sistema.

3.2.3 Modelado de requisitos

La función principal, o la más conocida del diagrama de casos de uso es documentar los requisitos del sistema o de una parte de él, por lo que se recomienda:

- Establecer que son roles no personas.
- Establecer su contexto, para lo que también podemos usar un diagrama de casos de uso.
- Identificar las necesidades de los elementos del contexto (actores).
- Nombrar esas necesidades y darles forma de caso de uso.
- Identificar que casos de uso pueden ser especializaciones de otros, o buscar especializaciones comunes para los casos de uso ya encontrados.
- Identificar el alcance del sistema para cada ciclo indicando que casos de uso se incluirán en cada ciclo.

3.2.4 Diagrama de casos de uso

Un diagrama de casos de uso incluye los actores identificados, los casos de uso para cada actor y el alcance del sistema para cada ciclo. En un diagrama general se deben poner las funcionalidades o casos de uso más generales y posteriormente en diagramas más detallados se hace el desglose de cada funcionalidad.

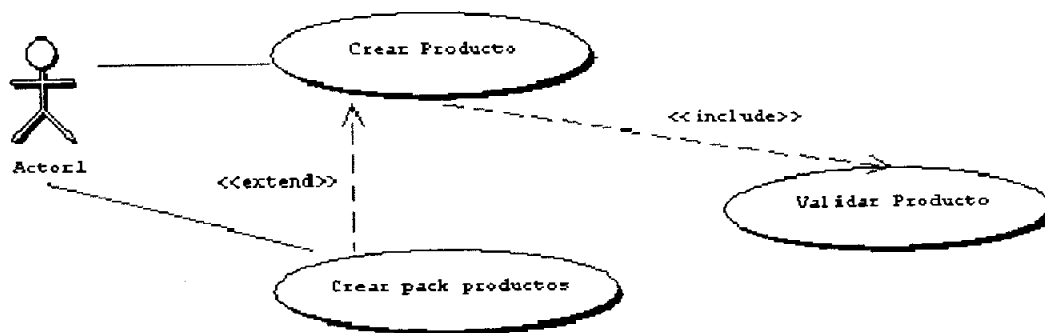


Figura 3.1. Diagrama de casos de uso

En Figura 3.1 se muestran tres casos de usos *Crear producto*, que utiliza *Validar producto*; y *Crear pack productos* que es una especialización de *Crear productos*. Estos diagramas se emplean de dos formas diferentes: para modelar el contexto de un sistema, y para modelar los requisitos del sistema.

3.3 Diagramas de clase

Un diagrama de clase representa las clases importantes de la aplicación con sus responsabilidades (atributos y operaciones), sus relaciones que modelan en uno o varios diagramas de clase.

El diagrama de clase describe los tipos de objetos que hay en el sistema y las diversas clases de relaciones estáticas que existen entre ellos [8]

Para identificar las clases del software, se analiza cada caso de uso y teniendo en cuenta la arquitectura propuesta, se buscan qué clases se necesitan en cada paquete.

Una vez analizados todos los casos de uso, se puede construir uno o varios diagramas de clases en los que se agrupen todas las clases de la lógica de la aplicación.

Para construir esos diagramas de clases se puede usar la siguiente técnica:

Las tarjetas CRC o Clase-Responsabilidad-Colaboración se usan para identificar tanto las clases de control como sus responsabilidades. Para cada responsabilidad se busca con qué clases tiene que colaborar. Estas tarjetas tienen la forma:

Tarjeta CRC

Nombre de clase	
Responsabilidad	Clase con que colabora

El proceso que se sigue para identificar las clases de la lógica de la aplicación es:

1. Selecciona un caso de uso y un escenario o detalle del caso de uso específico.
2. Por cada paso de la interacción del actor con el sistema que se tiene en el detalle del caso de uso, se imagina qué clases se requieren para realizar las acciones.
3. Por cada clase identificada escribe su nombre, digamos A, en una tarjeta, y en la columna Responsabilidad anota la acción R que debe de realizar esta clase.
4. Se analiza la responsabilidad R y se trata de identificar de qué otras clases necesita la colaboración para cumplir con la responsabilidad R de la clase A.
 - a) En caso de identificarlas, digamos B y C, se apuntan sus nombres en la columna de Colaboración de la clase A en la fila correspondiente a la responsabilidad R. Para clases colaboradoras B y C se repite el proceso de crear sus tarjetas y lo indicado en el punto 3.
 - b) No se anota nada en la columna de colaboración si se considera que la clase A puede cumplir sola con su responsabilidad R
5. Se analiza siguiente paso del escenario y se regresa al punto 2.
6. El proceso termina cuando se termina la secuencia de pasos en el escenario del caso de uso.
7. Se toma el siguiente caso de uso y se repite el proceso.

Para traducir cada tarjeta a una clase:

1. Por cada tarjeta se crea una clase con el mismo nombre en singular y con mayúscula.

2. Se identifica que atributos tendrá esa clase. Se puede revisar el prototipo para identificar los datos.
3. Por cada responsabilidad de la clase se incluye la operación correspondiente nombrada como verbo en infinitivo.
4. Para cada colaboración entre clases, dibujar una relación de asociación entre ambas clases.
5. Si en un caso de uso y un escenario se identifican las clases que ya aparecieron en otros casos de uso y /o escenarios , entonces la nueva información (operaciones y relaciones) se agrega al diagrama ya existente.

Otras relaciones que pueden existir entre las clases son herencia o generalización/especialización y agregación.

La herencia sirve para describir una clase general que comparte sus atributos y operaciones con clases que los especializan. La relación entre la clase general y sus especializaciones se denota por una flecha triangular del lado de la general. Se tiene herencia cuando varias clases comparten responsabilidades comunes que pueden tener especializaciones. Las subclasses heredan todos los atributos y operaciones de la superclase. La herencia debe mantenerse a un nivel conceptual de forma que el diagrama sea más fácil de entender.

La agregación es un tipo de asociación que denota pertenencia o que una clase está compuesta por otras. Se denota por un rombo en la clase compuesta. Igual que con la herencia esta relación debe ser conceptual y para facilitar el entendimiento del diagrama.

Una vez teniendo las clases de la lógica de la aplicación, se identifican las clases de la capa de interfaz de usuario. [Coad] propone que para encontrarlas, se revisen las clases anteriores y se identifiquen las clases con las que podrá interactuar el usuario, se construye una nueva clase con estereotipo de interfaz con el mismo nombre y que en general, tendrá como único atributo un objeto de la clase de la lógica de la aplicación.

Para identificarlas se puede revisar el prototipo creado en la fase de requerimientos. Las operaciones de estas clases serán las correspondientes a las interfaces de usuario como Abrir, Cerrar, Minimizar, etc.

De igual manera, se pueden identificar las clases de la capa de almacenamiento, que tendrán el estereotipo de entidad. Se crea una clase para cada clase de la lógica de la aplicación que se quiera sea persistente o esté almacenada. Su único atributo será también un objeto de la clase correspondiente de la capa de la lógica y las operaciones serán las típicas de los almacenamientos como Guardar, Extraer, Borrar, etc.

Una ventaja que tiene la propuesta de que en las clases de las capas de interfaz y de almacenamiento tenga n como único atributo un objeto de la clase correspondiente de la capa de la lógica, es la facilidad de modificación de estos atributos, pues sólo se hacen en la clase de lógica y no en las otras correspondientes.

3.3.1 La Clase

Una clase es una descripción de un conjunto de objetos que comparten los mismos atributos, operaciones, relaciones y semántica

Cada clase se representa con un rectángulo con las siguientes características:

- nombre de la clase
- atributos de la clase
- operaciones de la clase

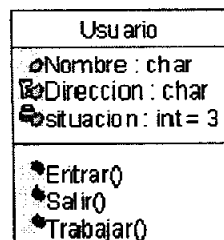


Figura 3.2. La Clase

En la Figura 3.2 se muestra la clase usuario que contiene tres atributos.

3.3.2 Relaciones entre clases

Existen otras relaciones diferentes entre clases: agregación, dependencia, generalización y asociación, y se establecen dependencias entre ellas

Agregación

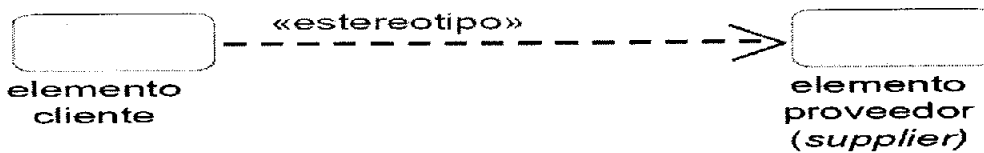
Es una relación estructural entre un todo y sus partes. Se denota por una línea terminada en un "diamante" en el extremo de la clase que representa el todo



Puede ser caracterizada con precisión determinando las relaciones de comportamiento y estructura que existen entre el objeto agregado y cada uno de sus objetos componentes.

Dependencia

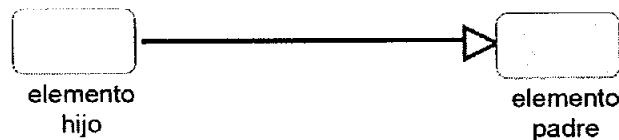
Es una relación semántica entre dos elementos (o dos conjuntos de elementos), en la cual un cambio en un elemento puede afectar a la semántica de otro elemento.



Existen varios tipos de dependencia predefinidas que se indican mediante estereotipos, por ejemplo: extend, e incluye para casos de uso.

Generalización

Es una relación taxonómica entre un elemento más general (el padre) y un elemento más específico (el hijo)



Asociación

Es una relación estructural entre dos elementos, que describe las conexiones entre ellos (suele ser bidireccional)



En la Figura 3.3 se presenta un diagrama de clases que muestra un conjunto de clases y sus relaciones desde un punto de vista lógico

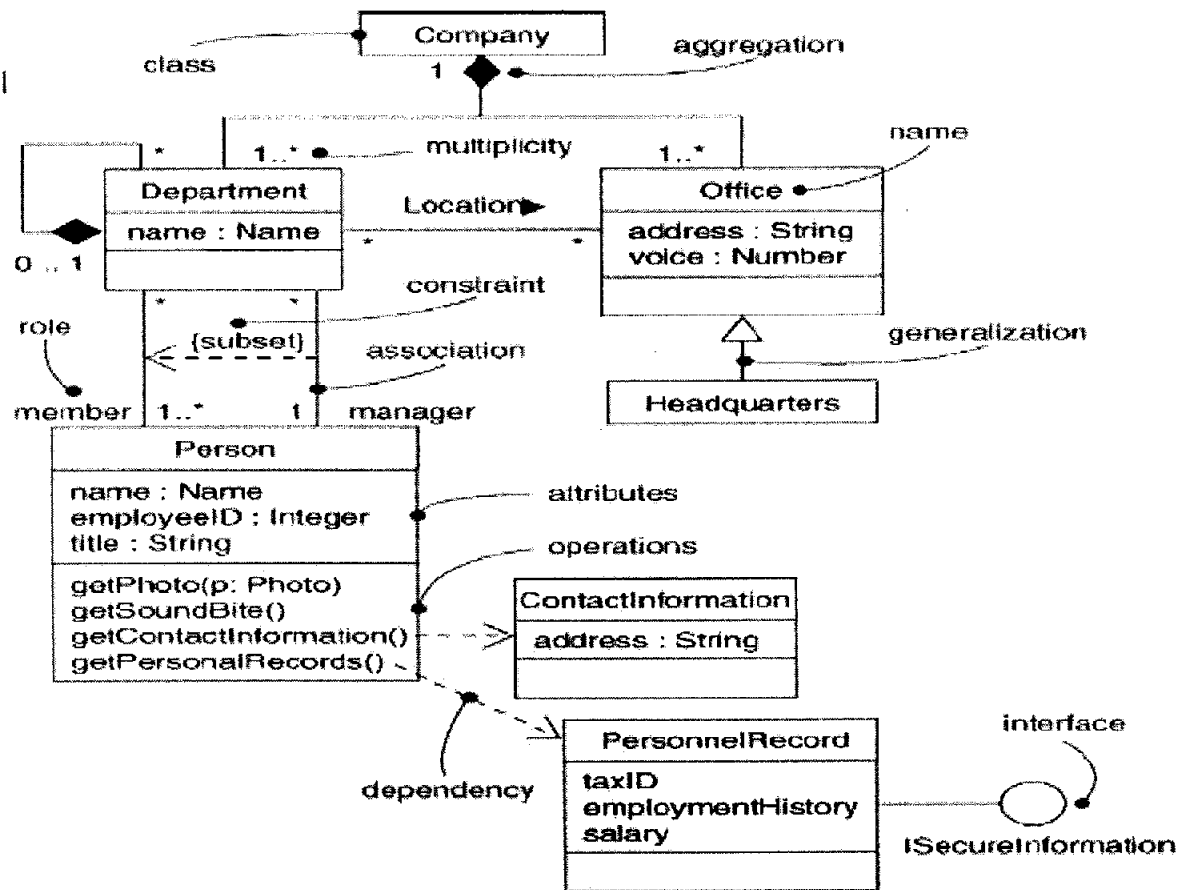


Figura 3.3. Diagrama de Clases

3.4 Diagramas de objetos

En este diagrama se modelan los ejemplares de las clases del diagrama de clases. Muestra a los objetos y sus relaciones, pero en un momento concreto del sistema. Estos diagramas contienen objetos y enlaces. En los diagramas de objetos se incorporan clases -Figura 3.4.

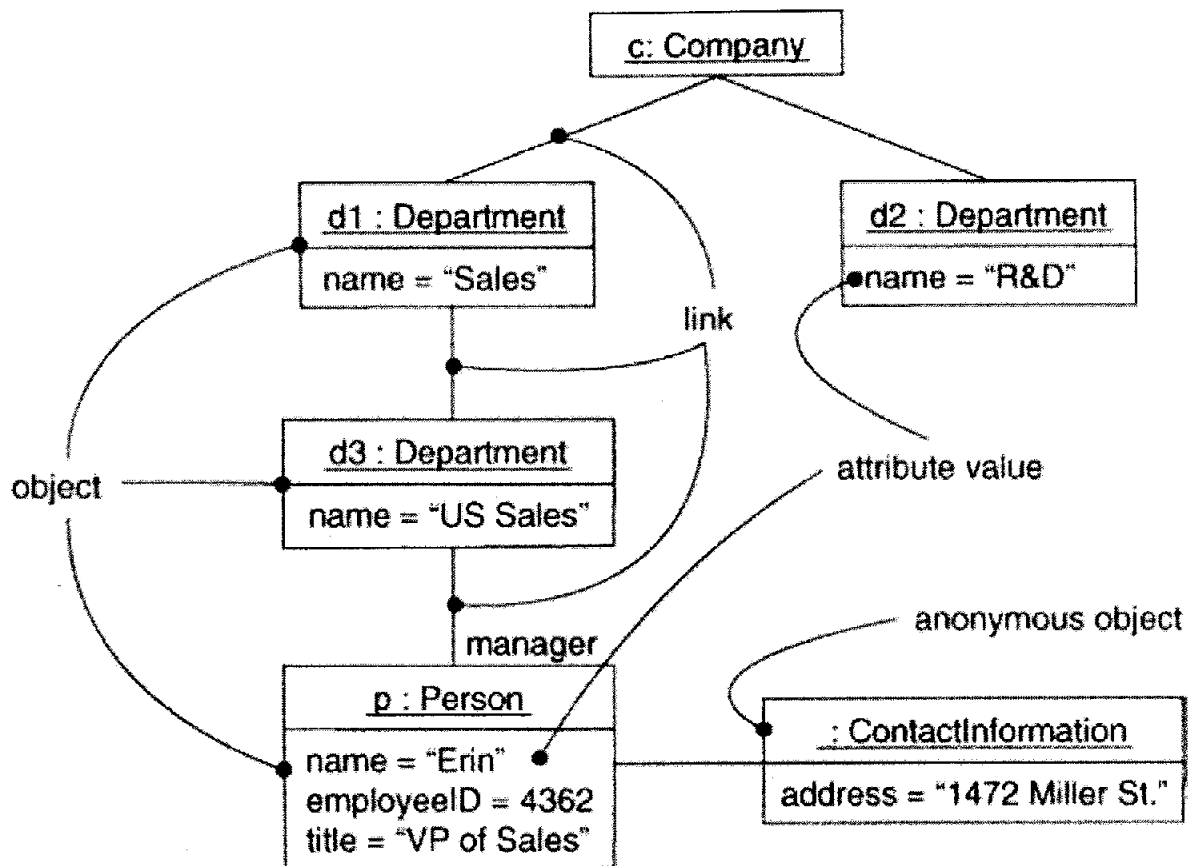


Figura 3.4. Diagrama de Objetos

3.5 Diagrama de componentes

Los diagramas de componentes describen los elementos físicos del sistema y sus relaciones –Figura 3.5. Muestran las opciones de realización incluyendo código fuente, binario y ejecutable

Se utilizan para modelar la vista estática de un sistema. Muestran la organización y las dependencias entre un conjunto de componentes.

3.6 Diagramas de despliegue

Un diagrama de despliegue muestra las relaciones físicas entre los componentes hardware y software en el sistema final, es decir, la configuración de los elementos de procesamiento en tiempo de ejecución y los componentes software (procesos y objetos que se ejecutan en ellos). Estarán formados por instancias de los componentes software

que representan manifestaciones del código en tiempo de ejecución (los componentes que sólo sean utilizados en tiempo de compilación deben mostrarse en el diagrama de componentes).

Un diagrama de despliegue es un grafo de nodos unidos por conexiones de comunicación. Un nodo puede contener instancias de componentes software, objetos, procesos (caso particular de un objeto). En general un nodo será una unidad de computación de algún tipo, desde un sensor a un mainframe. Las instancias de componentes software pueden estar unidas por relaciones de dependencia, posiblemente a interfaces (ya que un componente puede tener más de una interfaz).

En el diagrama -Figura 3.6- de despliegue se indica la situación física de los componentes lógicos desarrollados. Es decir se sitúa el software en el hardware que lo contiene.

Un nodo se representa como un cubo, y es un elemento donde se ejecutan los componentes, representan el despliegue físico de estos componentes.

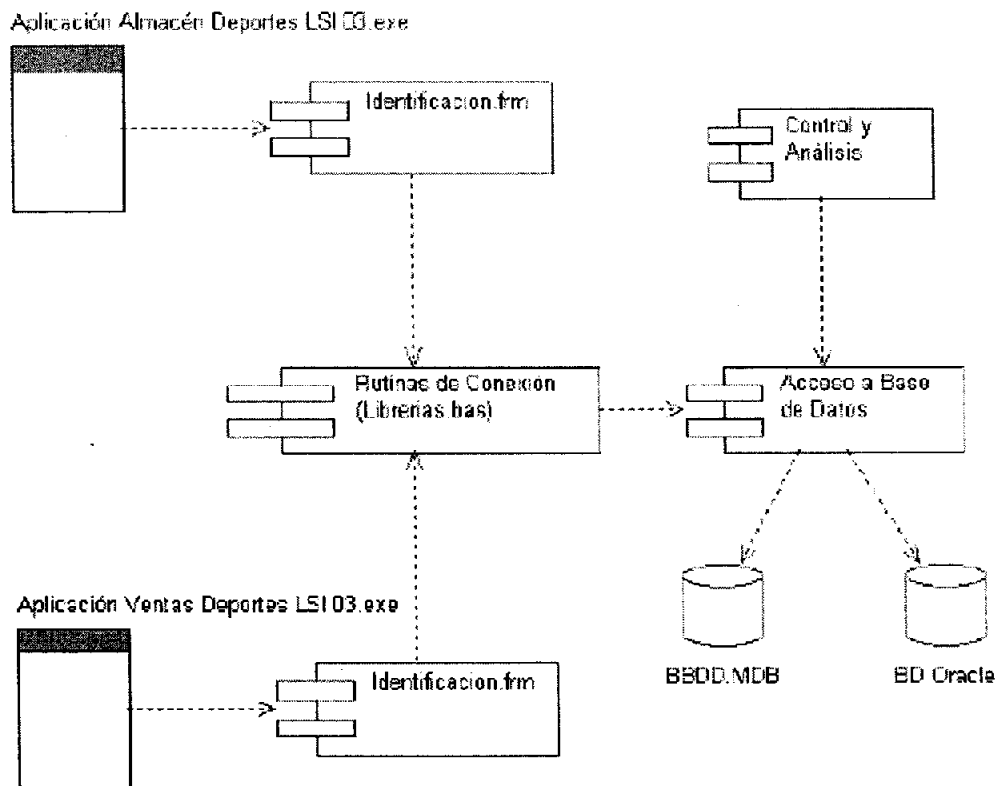


Figura 3.5. Diagrama de componentes

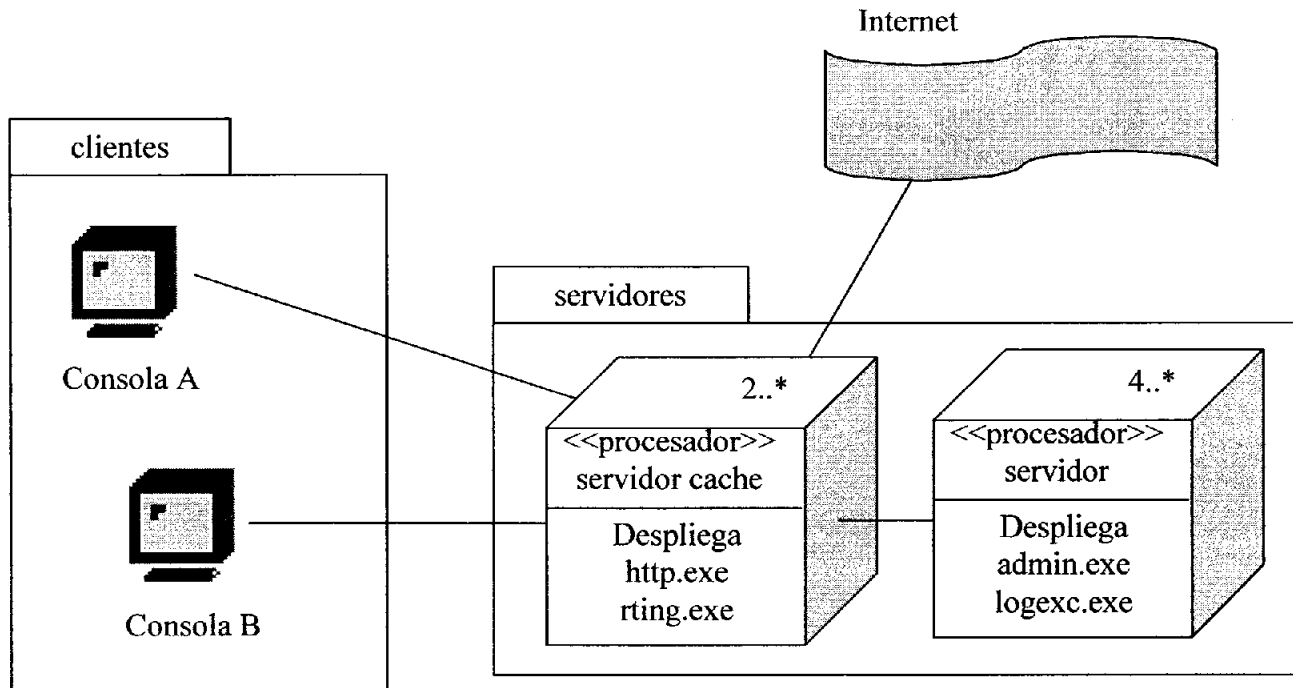


Figura 3.6. Diagrama de despliegue

3.7 Diagramas de secuencia

El diagrama de secuencia forma parte del modelado dinámico del sistema -Figura 3.7. Se modelan las llamadas entre clases desde un punto concreto del sistema. En él se muestra el orden de las llamadas en el sistema. Se utiliza un diagrama para cada llamada a representar. El diagrama se forma con los objetos que forman parte de la secuencia, estos se sitúan en la parte superior de la pantalla, normalmente en la izquierda se sitúa al que inicia la acción- Figura 3.7.

Los diagramas de secuencia, se utilizan para validar los casos de uso. Documentan el diseño desde el punto de vista de los casos de uso, nos ayudan a comprender los cuellos de botella potenciales, para así poder eliminarlos.

3.8 Diagramas de colaboración

Un diagrama de colaboración visualiza las interacciones entre los objetos según las "llamadas" entre ellos -Figura 3.8.

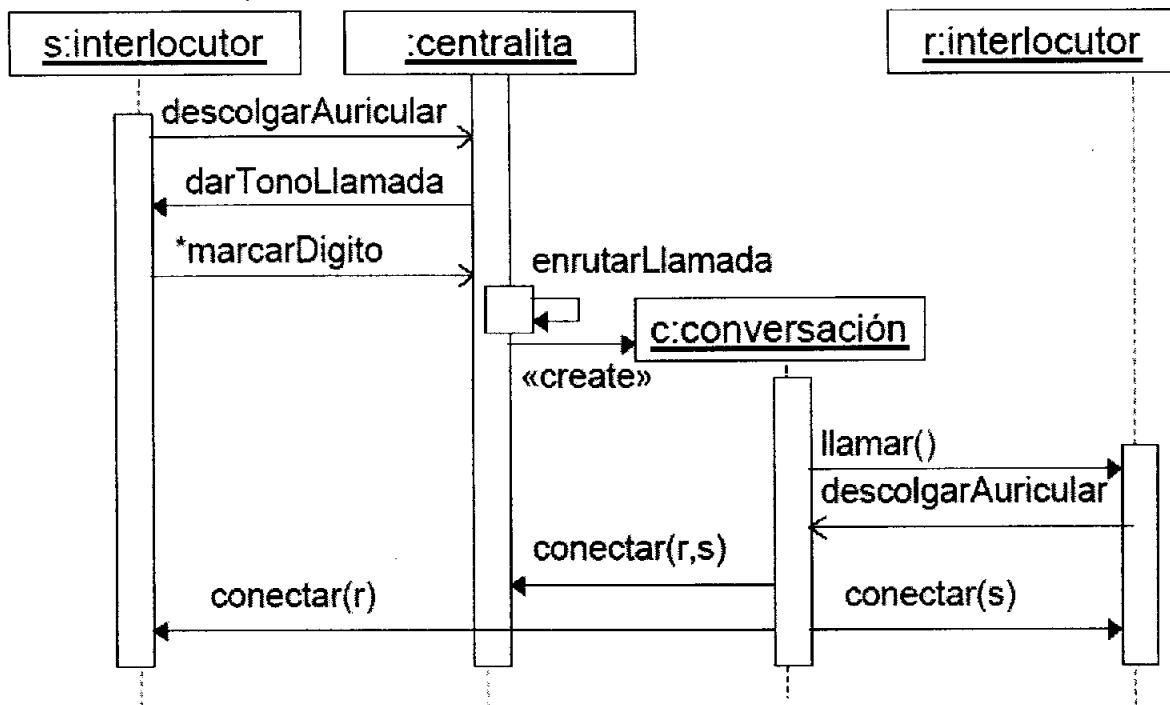


Figura 3.7. Diagrama de secuencia

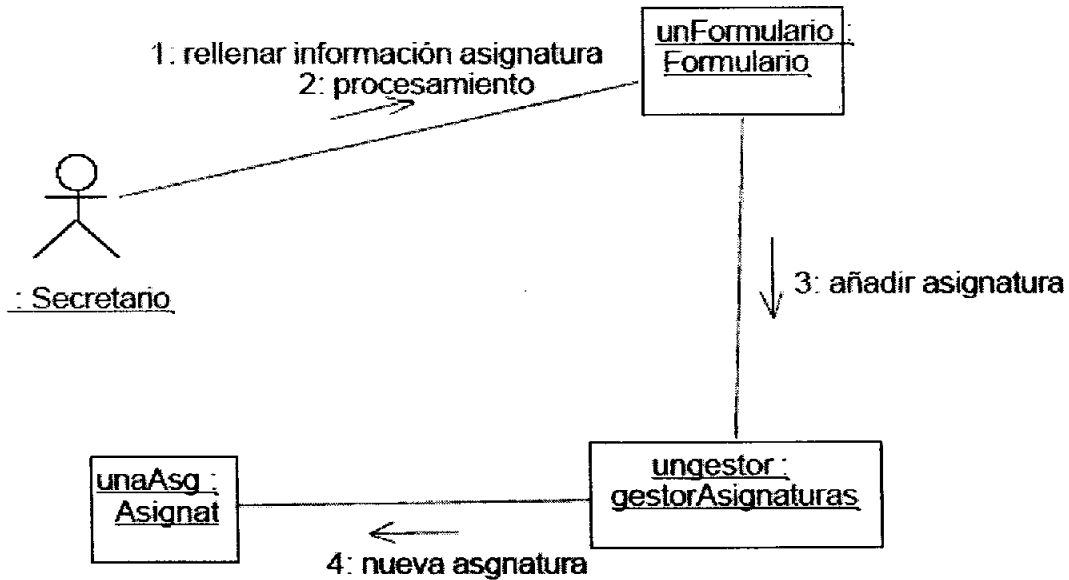


Figura 3.8. Diagrama de colaboración

3.9 Diagramas de paquetes

Sirve para organizar elementos en grupos. Un paquete es puramente conceptual, sólo existe en tiempo de desarrollo.

Un paquete es un mecanismo general de UML para organizar elementos en grupos, se usa para modelar elementos que forman un todo [Booch p. 170], en la representación de la arquitectura representan un componente o funcionalidad del sistema

El diagrama de paquetes se construye mostrando las dependencias de los paquetes. Los paquetes pueden contener clases, interfaces, otros paquetes, código html, etcetera.

Tipos de relaciones entre paquetes: *Dependencia* (línea punteada posiblemente con flecha) Un paquete depende de otro si un cambio en el primero, puede afectar al que depende de él. Un ejemplo de una relación de dependencia entre paquetes es por ejemplo la importación de un paquete por otro.

Asociación (línea continua posiblemente con flecha). Establece una relación entre dos paquetes, un paquete requiere de los servicios de otro.

Generalización (línea continua con flecha transparente que apunta hacia la cosa más general). Un paquete representa aspectos mas generales y otro los especializa, es la relación de herencia.

Realización (línea con flecha transparente que apunta hacia la cosa que va a ser realizada por otra). Es un contrato que promete que un paquete va a ser realizado por otro en la implementación (ej. entre una interfaz y las clases que la implementan).

Con el patrón de capas, se tienen tres paquetes, uno para cada capa:

- Paquete de interfaz de usuario
- Paquete de lógica de la aplicación
- Paquete de almacenamiento.

Las relaciones entre estos tres paquetes será de asociación del paquete de nivel inferior (almacenamiento) al intermedio (lógica de la aplicación) y de éste al superior (interfaz de usuario).

En la figura siguiente se muestra un diagrama con las 3 capas de la arquitectura. La capa de interfaz está compuesta por un paquete, el de la interfaz de usuario que se desglosará posteriormente.

La capa del negocio o de la lógica de la aplicación contiene un paquete Control de préstamos que está asociado por los otros 4 paquetes que corresponden a los principales casos de uso del sistema de *Préstamos personales*.

La capa de datos o almacenamiento contiene el paquete con la base de datos.

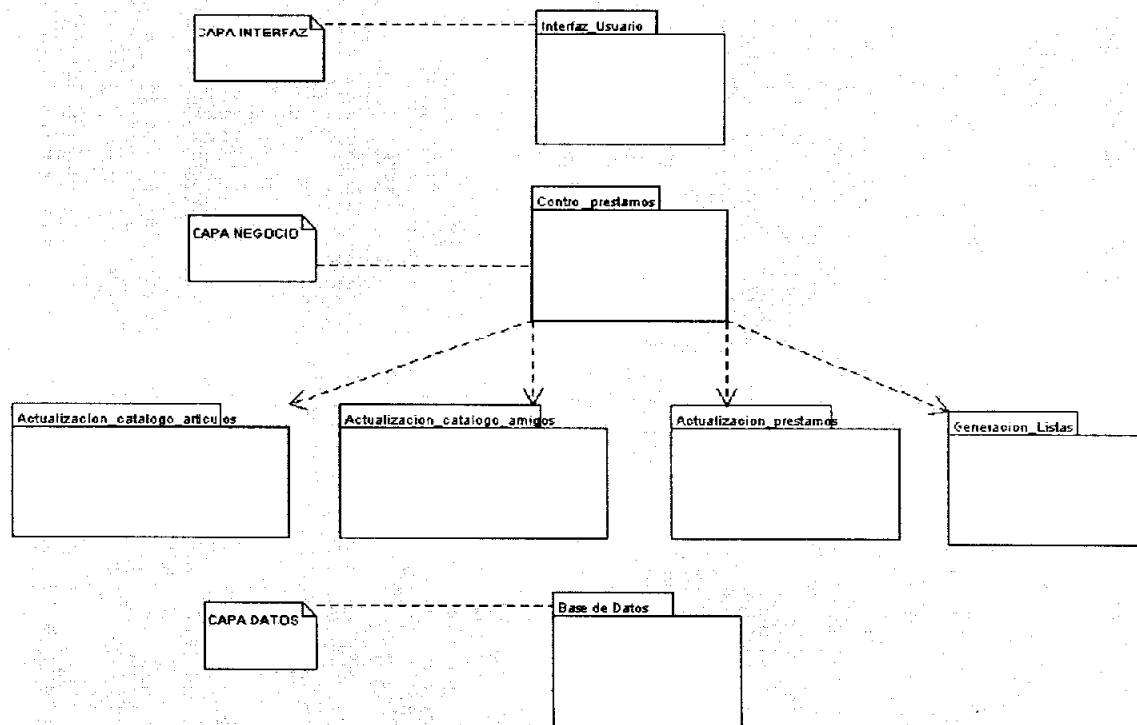


Figura 3.9. Diagrama de paquetes

3.10 Diagrama de distribución

Muchos de las aplicaciones actuales son distribuidos. Un sistema es distribuido si corre en más de una máquina y sus componentes se encuentran distribuidos en ellas. Para mostrar este aspecto de la arquitectura, UML tiene los diagramas de distribución (deployment). Se define que se instalará en cada máquina y cómo se conectarán entre ellas.

Los elementos de estos diagramas son los nodos y sus conexiones. Al construir el diagrama de distribución se asignan los paquetes a los nodos.

Un nodo es un elemento físico que existe y representa un recurso computacional [Booch p.360]. Se representa por un cubo que debe nombrarse. En los nodos se pueden representar los paquetes que deberán correr en esa máquina.

Las conexiones son relaciones que representan las conexiones físicas entre los nodos. Pueden etiquetarse con el protocolo de conexión.

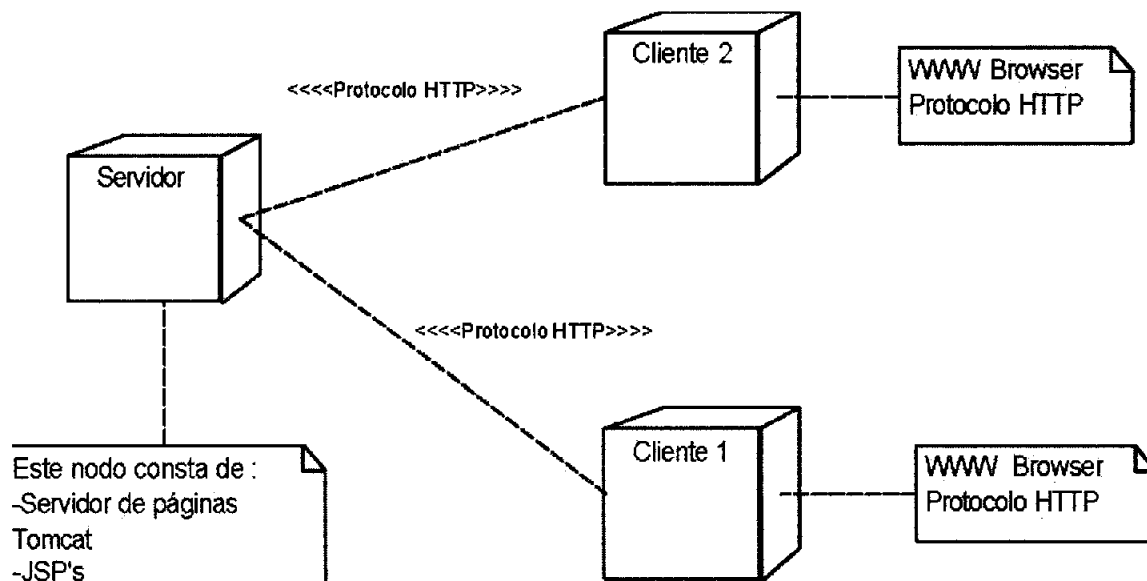


Figura 3.10. Diagrama de distribución

3.11 Diagrama de estados

Los diagramas de estado son una técnica para describir el comportamiento de un sistema. Describen los estados posibles en los que puede entrar un objeto particular y la manera en que cambia el estado del objeto, como resultado de los eventos que llegan a él. En la Figura 3.11 se presenta un diagrama de estados para la clase persona.

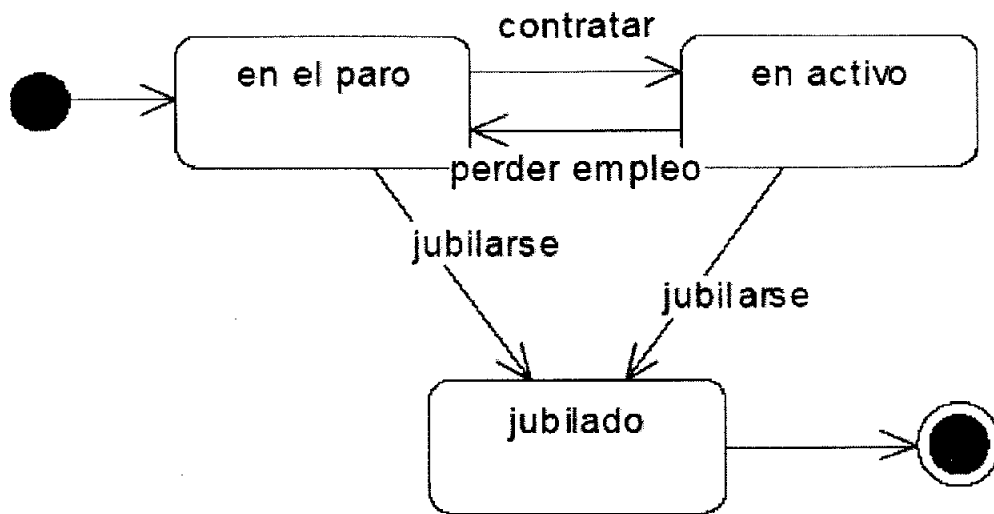


Figura 3.11. Diagrama de estados

II Aplicación

Capítulo 4. Panorama general al sistema de hatos

4.1 Introducción

La producción de leche en México a nivel empresarial es una industria reciente cuya evolución ha sido rápida en los últimos cincuenta años. A partir de la firma del Tratado de Libre Comercio (TLC), en enero de 1994, los niveles de importación de algunos productos lácteos (como la leche en polvo sin grasa) disminuyeron drásticamente del 91% en 1994, al 44% en 2001, previéndose que los granjeros nacionales producirán el 75% de los requerimientos de leche en los próximos años (Dobson & Proctor, 2002). Sin embargo, en otros productos, como el queso, la competencia es fuerte y los empresarios han lanzado estrategias para evitar o disminuir la importación de leche.

Por otro lado, la producción mundial de leche ha aumentado considerablemente, sin guardar paralelo con el incremento en la población de bovinos. En Estados Unidos, por ejemplo, antes de la segunda guerra mundial, se producían anualmente poco más de 45 mil millones de litros, con cerca de 26 millones de vacas, mientras que en la década de los sesentas aumentó a 56,400 millones de litros con una población bovina de 20 millones (Climent y Villanueva, 1997). La situación nacional indica que el número de vacas ha aumentado en tan sólo 5% de 1994 a 2001, sin embargo la producción de leche se incrementó en 23% debido a una mayor productividad por vaca.

Este aumento en productividad se debe a la aparición de innovaciones tecnológicas, como la inseminación artificial, la pasteurización, la refrigeración, la deshidratación y la homogeneización de la leche, la extracción automatizada de leche, la alimentación individualizada y aplicación computarizada para el control de mastitis y el pesaje de leche. El uso de herramientas de manipulación genética así como el desarrollo de nuevos antibióticos y vacunas han permitido una disminución de las tasas de morbilidad¹ y mortalidad de la población. Se considera como uno de los factores más decisivos e importantes en el aumento de la producción de leche, la constante selección de los animales más precoces y productivos a través de técnicas avanzadas como el DHI (Dairy Herd Improvement) y las pruebas de progenie de sementales. Si bien todo lo anterior ha

¹ Cantidad de personas o animales que enferman en un lugar y un periodo de tiempo determinado en relación con el total de la población..

repercutido positivamente en la industria lechera, también ha aumentado de manera notable su complejidad.

Aunado a esto, los granjeros mexicanos han padecido producir a altos costos y ganancias mínimas debido al tope en el precio de la leche, no así de los insumos. Mientras menor sea el margen de ganancia, la intervención del productor en el proceso de planificación es sumamente importante, requiriendo contar con un sistema de evaluación y control continuo. El tener una herramienta que le permita poseer información confiable y oportuna lo apoyará en la toma de decisiones.

4.2 Origen del sistema

Las empresas lecheras tienen uno de los sistemas de producción alimentaria más complejo que haya desarrollado el ser humano. En él convergen un sinnúmero de factores sociales, económicos, técnicos y ambientales, que por su estrecha vinculación, no es posible entender fuera de un concierto de elementos y procesos interdependientes.

Ante los múltiples y complejos problemas que día con día enfrentan los productores de leche en nuestro país, el análisis de elementos aislados resulta insuficiente, por lo que los apoyos de asesoría e investigación deben ampliar su perspectiva, tanto como sea posible.

En los centros educativos de medicina veterinaria y zootecnia, en particular, y de ciencias agropecuarias, en general, es necesaria la formación de recursos humanos que no sólo estén mejor preparados para hacer más competitivo al sector lechero, sino también, para ampliar el horizonte de los problemas que afectan a éste.

La complejidad de los procesos multivariables, como es el de la producción de leche, hacen que su análisis sea sumamente lento y costoso, si no inaccesible, cuando no se cuenta con medios e instrumentos computarizados. Por ello, los modelos de simulación forman parte de las herramientas clave para facilitar el estudio de dichos procesos.

EL Simulador de Hatos² Lecheros con Proyección a 7 Años tiene como primer propósito servir a profesores y alumnos de carreras afines a la producción de alimentos agropecuarios, en la integración de algunos de los aspectos administrativos y operativos

² Hato: Conjunto de animales de ganado mayor o menor.

más relevantes a las explotaciones de bovinos lecheros. No obstante, se espera que también sea de provecho para asesores técnicos así como de productores.

Este simulador tiene como objetivo proyectar la estructura del hato y su comportamiento productivo a lo largo de un período de siete años, dividido en intervalos anuales. Es útil para realizar estudios prospectivos, como es el caso de estudios de factibilidad de proyectos, así como estudios técnicos y económicos a largo plazo. Su uso es particularmente conveniente en la elaboración de planes y programas destinados a la creación de nuevas empresas lecheras; para examinar, a base de simulacros, las perspectivas económicas de una explotación bajo distintos parámetros productivos; y para realizar análisis comparativos de situaciones hipotéticas.

4.3 Supuestos básicos

Como cualquier otra herramienta de sistemas para el análisis de proyectos, el Simulador depende, en cuanto a procesos y resultados, de los datos provistos. Así, pueden darse las siguientes situaciones:

- 1) Los datos son inoperantes.- habrá fallas de procesamiento, comúnmente notorias por la falta o inconsistencia de resultados.
- 2) Los datos sobrestiman o subestiman el comportamiento esperado de las variables.- se puede cometer el error de crear mayores o menores expectativas que afecten la adecuada estimación de la factibilidad del proyecto analizado, sus ventajas y desventajas.
- 3) Los datos son procesables, parten de criterios y condiciones debidamente fundamentados.- (a) Se cuenta con información suficiente, precisa y confiable sobre el comportamiento de la explotación en un período de varios años; (b) la información proviene de explotaciones lecheras operando en situaciones específicas, similares a las del caso en cuestión; (c) la información generada se deriva de bases teóricas sólidas (científicas) ajustadas con sentido práctico, según la experiencia del usuario y sus fuentes de información (coparticipantes del proyecto); o bien, (d) las fuentes de información se combinan.

Para operar de manera correcta el Simulador, el usuario deberá entender y

manipular los supuestos básicos bajo el tercer lineamiento: los datos son procesables, parten de criterios y condiciones debidamente fundamentados.

El Simulador no pretende ser un medio de predicción sobre el comportamiento futuro de una explotación o proyecto lechero, sino una herramienta que permita construir posibles escenarios futuros a partir de una serie de condiciones o supuestos.

Dado que las explotaciones lecheras son sistemas complejos de intervención humana, es difícil, si no imposible, que el cerebro de un individuo conciba y asocie de manera simultánea la gran cantidad de factores involucrados. Incluso los científicos, en estudios relativos a problemas de la ciencia, en general, y del desarrollo lechero, en particular, analizan un reducido número de variables en busca de la comprensión y el control de los fenómenos observados. Así, los estudios de carácter científico tienen su origen en una concepción formal de los problemas e inducen a medidas correctivas de base científica.

Por cuanto toca a los propietarios o administradores de las empresas lecheras - micro, pequeñas, medianas o grandes-, la toma de decisiones, la planificación estratégica y la implementación de acciones a corto, mediano y largo plazo, tienen otro marco de sustentación y análisis: el de la realidad determinada por el ambiente y las condiciones en que se desenvuelven tales empresas. Aquí entran en juego todas las variables relativas a la existencia, creación, modificación, comprensión y operación de situaciones concretas, que incluyen aspectos físicos, políticos, económicos, sociales, biológicos y técnicos, estrechamente interrelacionados. Para la comprensión y análisis de situaciones concretas en contextos reales es preciso integrar sistemas de información multifactoriales y dar cabida al desarrollo una cultura coparticipativa por parte de las personas involucradas en y con el sistema. Son éstas las vías fundamentales para promover en los sistemas de producción, maneras de pensar, actuar y ser, no sólo más eficaces y eficientes, sino sobretodo, más sensatas, justas y coherentes.

En este sentido, el Simulador contribuirá a desarrollar un pensamiento integrador en la concepción de los problemas de los proyectos lecheros (en planeación u operación), así como un mayor interés por este tipo de noción para abordar la realidad de las explotaciones lecheras como situaciones concretas de sistemas complejos, a lo largo de su ciclo de vida.

Las explotaciones lecheras, como todos los sistemas complejos, son extremadamente dinámicos y sensibles: se construyen, descomponen y reconstruyen constantemente. El cambio en la magnitud, calidad o temporalidad de un solo elemento puede ocasionar cambios en el resto de los elementos, como si se tratara de un caleidoscopio. Un de los principales objetivos del Simulador es proyectar, dentro de sus propias limitaciones, una imagen caleidoscópica de la realidad sujeta a constante cambio. Será responsabilidad del usuario ayudarse del Simulador para acrecentar su propia noción sistémica sobre el problema.

Por supuesto que la información obtenida está sujeta a la interpretación del usuario, la cual depende de su experiencia en el área.

Capítulo 5. Análisis del Sistema

5.1 Introducción

El primer paso para determinar las necesidades del usuario es el análisis del sistema, en este proceso el Analista se reúne con el usuario para identificar las metas globales, se analizan sus perspectivas, necesidades y requerimientos, sobre la planificación temporal y presupuestal, líneas de mercado, entre otros puntos que puedan ayudar a la identificación para el desarrollo del proyecto.

El Análisis del Sistema se lleva a cabo teniendo los siguientes objetivos:

- Identificar las necesidades del usuario.
- Evaluar qué conceptos tiene el usuario del sistema para establecer su viabilidad.
- Realizar un análisis técnico y económico.
- Asignar funciones al hardware, software, personal, base de datos, y otros elementos del sistema.
- Establecer las restricciones de presupuestos y planificación temporal.
- Crear una definición del sistema que forme el fundamento de todo el trabajo de Ingeniería.

Para lograr estos objetivos se requiere tener conocimiento, dominio del hardware, del software, de la ingeniería humana (manejo y administración de personal), así como administración de base de datos.

Para comprender las posibilidades y limitaciones de un modelo de simulación es preciso observar sus propiedades desde la concepción estructural y funcional de su sistema; o sea, de la reunión o conjunto de elementos relacionados que lo conforman.

El modelo con que opera el Simulador posee algunas de las características de un sistema cibernético: muestra una estructura circular, cuenta con mecanismos de retroalimentación, plantea una concepción de orden cerrada, por lo que es pobre en elementos contextuales. Asimismo sus procesos, de carácter esencialmente determinista, conllevan la causalidad en la relación de las variables implicadas. Estas propiedades

proporcionan referentes de cierta solidez, en cuanto a la visión futura de los análisis prospectivos a largo plazo, sobre el comportamiento de los proyectos de nueva creación, ampliación o modernización. Tales referentes plantean expectativas en el desempeño productivo y económico de las explotaciones, que es preciso complementar con ejercicios prospectivos operacionales, más flexibles, para la toma de decisiones y la realización de programas de trabajo a corto y mediano plazo.

A través del Simulador se consigue la racionalización de los datos disponibles para obtener información que no se tiene, con rapidez y a bajo costo (en las proyecciones de resultados). Para que el Simulador genere información útil, primero es necesario proporcionarle los datos requeridos; segundo, que estos datos sean consistentes. De hecho, el Simulador conduce, a partir de los datos con que se le alimenta, al desarrollo de un escenario esperado: dinámico, en cuanto al proceso evolutivo que sigue una explotación; y estático, con respecto a los resultados buscados. El usuario puede desarrollar diferentes escenarios que reflejen situaciones alternativas a las que podría llegar, de suerte que opte por orientarse a la que más le convenga.

5.2 Análisis del sistema

Con base a las entrevistas y a la documentación proporcionada, el sistema a realizar tiene como objetivo principal: *la creación de un simulador de hatos lecheros con proyección a 7 años por medio de la computadora*. Para realizarlo se tiene que tomar en cuenta los siguientes puntos.

- Evolución de la viabilidad del simulador.
- Efectuar un análisis técnico y económico.

Determinar las funciones a los usuarios, al software, al hardware, a la base de datos.

5.2.1 Lista de características

De acuerdo a las entrevistas realizadas se estableció que el sistema almacenará la información de cada proyección de forma estructurada. Se toman en cuenta los siguientes supuestos básicos.

I. Movimiento de ganado

- A) Capacidad de la explotación total de vacas en ordeño y secas.
- B) Edad del ganado en sus distintas etapas de desarrollo: terneras (del nacimiento a al destete): determinar el máximo de edad al destete de dos a tres meses; becerras (de la edad máxima de las terneras a los 12 meses); vaquillas (de los 12 meses a la edad mínima para el parto); vaquillas de reemplazo (rango de edad al parto). Determinar el rango de edad al parto entre 24 y 36 meses.
- C) Compra de ganado del primer al séptimo año en las diferentes etapas: número de terneras, número de becerras, número de vaquillas, número de vaquillas de reemplazo, número de vacas de 1er parto, número de vacas de 2 o más partos
- D) Inventario ganadero (población con la que cuenta la explotación al inicio de la proyección): número de ternera, número de becerras, número de vaquillas, número de vaquillas de reemplazo, número de vacas de 1er parto, número de vacas de 2 o más partos.
- E) Intervalo entre partos.
- F) Índices de mortalidad y desecho para las distintas etapas: terneras, becerras, vaquillas, vaquillas de reemplazo, vacas de 1er parto, vacas de 2 o más partos, del año 1 al 7.

II. Engorda de machos.

El sistema permitirá el almacenamiento de la información de la engorda de los machos¹, la cual requiere de la información del Apartado Movimiento de Ganado.

- G) Edad de los machos en las etapas de cría y desarrollo: *terneros* (del nacimiento al destete). Se maneja el rango de edad fijado para las terneras; *añojos* (novillos para el mercado de carne o el autoconsumo). Determinar el rango de edad para la venta, entre 12 y 24 meses.
- H) Porcentaje de terneros conservados.
- I) Índices de mortalidad y desecho para terneros y añojos: *terneros*, *novillos*

¹ Macho: Animal del sexo masculino

III. Producción de Leche.

Producción de leche requiere de la información del Apartado Movimiento de Ganado.

- J) Activación del efecto del intervalo entre partos (I.P.) sobre la producción de leche.
- K) Producción anual por vaca de 1er parto en el hato.
- L) Producción anual por vaca de 2 o más partos en el hato.

IV. Eliminación de ganado

Eliminación de ganado para rastro requiere información del Apartado Engorda de Machos).

- M) Porcentaje de las pérdidas globales por muerte y desecho en las distintas etapas. Es la proporción de ganado eliminado que puede enviarse al rastro: terneras, becerras, vaquillas, vacas y vaquillas de reemplazo, terneros, añojos.

V. Ingresos

Ingresos requiere de los Apartados Producción de Leche y Eliminación de Ganado para rastro.

- N) Precios de venta de: un litro de leche, una ternera de desecho, una becerro de desecho, una vaquilla de desecho, una vaca o vaquilla de reemplazo de desecho, un ternero de desecho, un añojo, una vaquilla de reemplazo.

VI. Costos globales.

- O) Costos anuales por concepto de: alimentación, salarios, compra de ganado, medicinas y vacunas, otros costos, intereses de capital, amortizaciones.

A partir de esta información se estructura un esquema general por medio del cual se construye un diseño de base de datos que permita organizar esta información de una manera eficiente.

5.2.2 Requerimientos de software

Con base en la información proporcionada se establecieron los lineamientos para desarrollar el simulador.

1.- El simulador se desarrolló en Visual Objects 2.0 con archivos de tipos dbf. Algunas de las razones para usar este paquete son las siguientes:

- Es un sistema de desarrollo de 32 bits, que permite crear aplicaciones sofisticadas.
- Es un paquete que combina el acceso a las bases de datos en ambiente usuario-servidor con el uso de un repositorio multitareas, el cual tiene facilidad de uso, flexibilidad, potencia y rapidez en la capacidad de realizar prototipos de ambiente de programación visual.
- Posee una compatibilidad ascendente, que da acceso a todas las características de este sistema, integrándolas fácil y transparentemente en jerarquías de clase
- Está diseñado para Windows 95, 98 y XP lo cual permite hacer "exitosas" las aplicaciones a todos los programadores y elementos de programación con un sistema de desarrollo comprensivo que es fácil de usar, potente y que da el control requerido por los programadores para lo cual se tienen herramientas compatibles.

2.- Para el manejo los archivos de bases de datos se utilizó DBase III Plus.

5.2.3 Requerimiento de interfaz.

La interfaz de usuario está conformada por un diseño que permite navegar por él.

1. El sistema debe facilitar al usuario hacer uso de las bondades del sistema.
 2. El simulador está organizado de tal manera que sean mínimos los errores del usuario.
 3. Los menús que se presentan son organizados de la mejor manera, para llevar a cabo todas las tareas de forma rápida y sencilla.
-

5.2.4 Requerimientos Especiales del Sistema

Estos requerimientos son parte del desarrollo del sistema, pero se les considera especiales ya que por medio de éstos, los usuarios tendrán a la mano la información necesaria para resolver dudas sobre el manejo del simulador.

Manual del usuario.

Módulo de ayuda dentro del sistema.

5.2.5 Diagramas de Casos de Uso

El proceso que efectúa el simulador contempla la selección de los módulos, como se ilustra en la Figura 5.1. El funcionamiento general del simulador Figura 5.2 permite al usuario seleccionar 2 opciones: supuestos básicos y reportes. Al seleccionar "Supuestos básicos", se presentan, en un segundo nivel, 6 opciones, tal como se muestra en la Figura 5.3. Para la generación de reportes, es posible tener resultados sobre 12 aspectos diferentes, así como un informe global Figura 5.4.

PROCESO DE USO DEL SIMULADOR

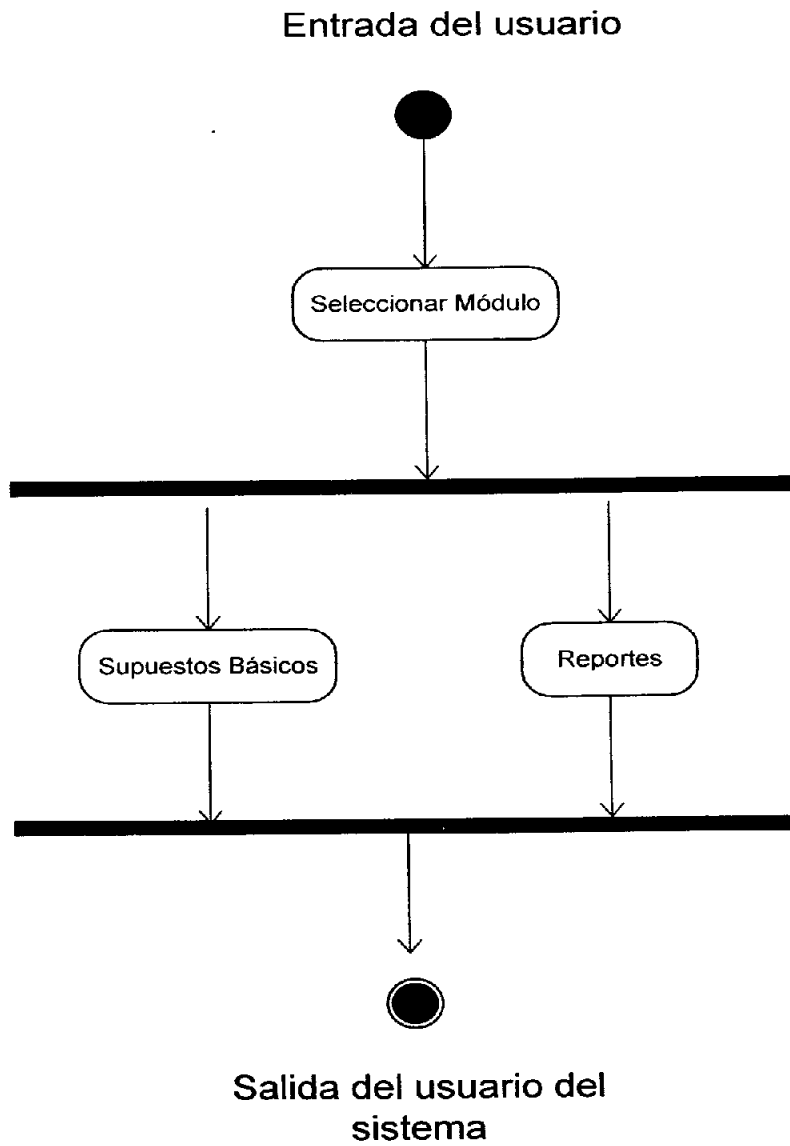


Figura 5.1. Funcionamiento General del Simulador

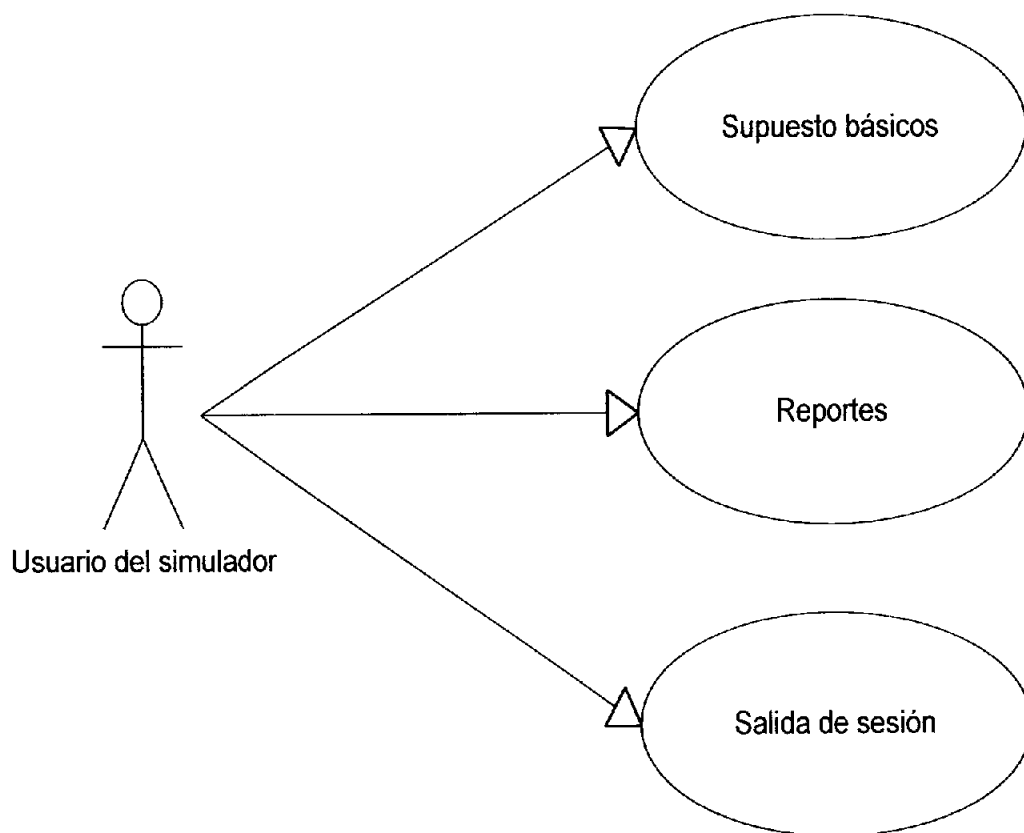


Figura 5.2. Diagrama de casos de uso del Simulador 1er. nivel

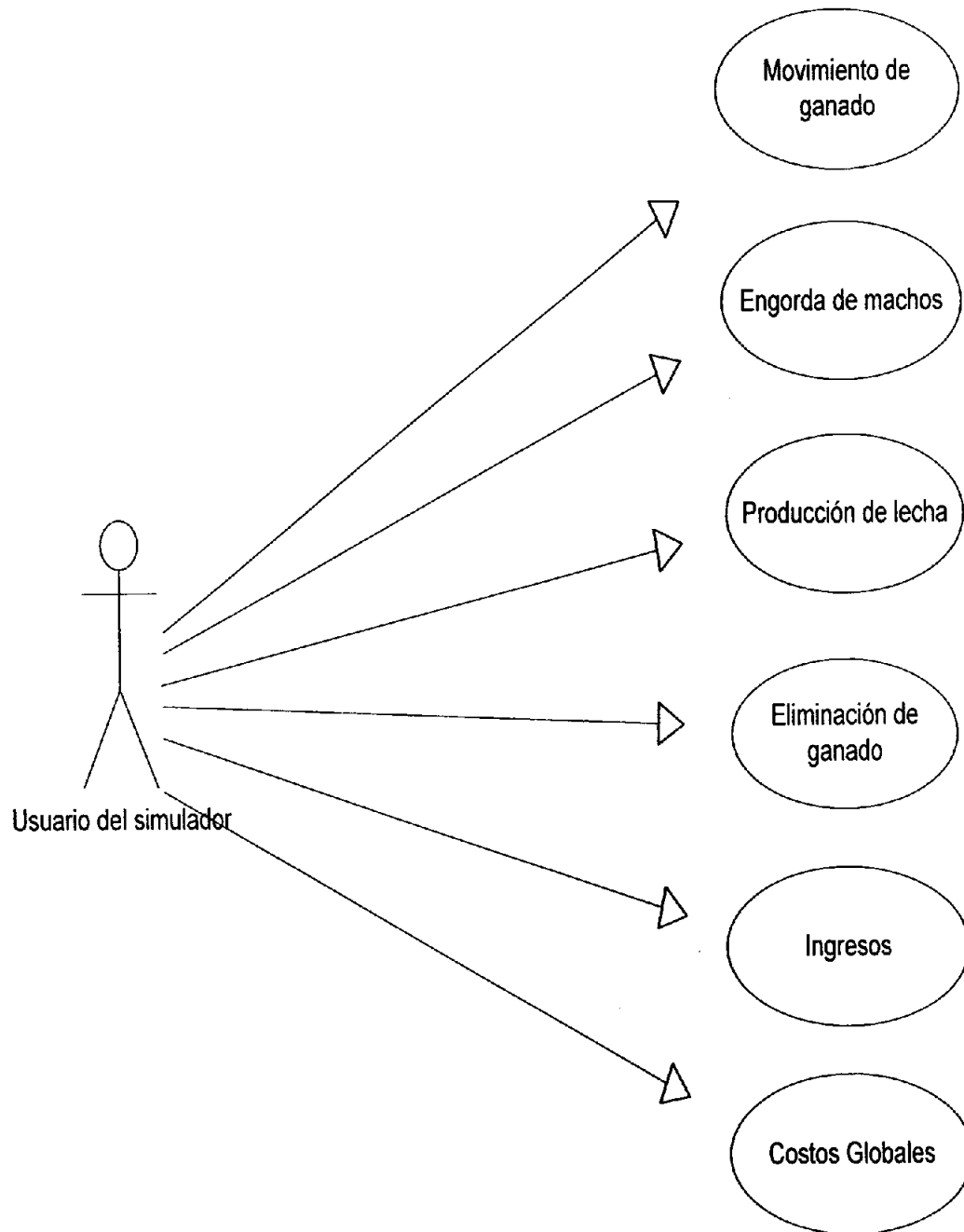


Figura 5.3. Diagrama de Casos del Simulador 2do. Nivel Opción "Supuestos básicos"

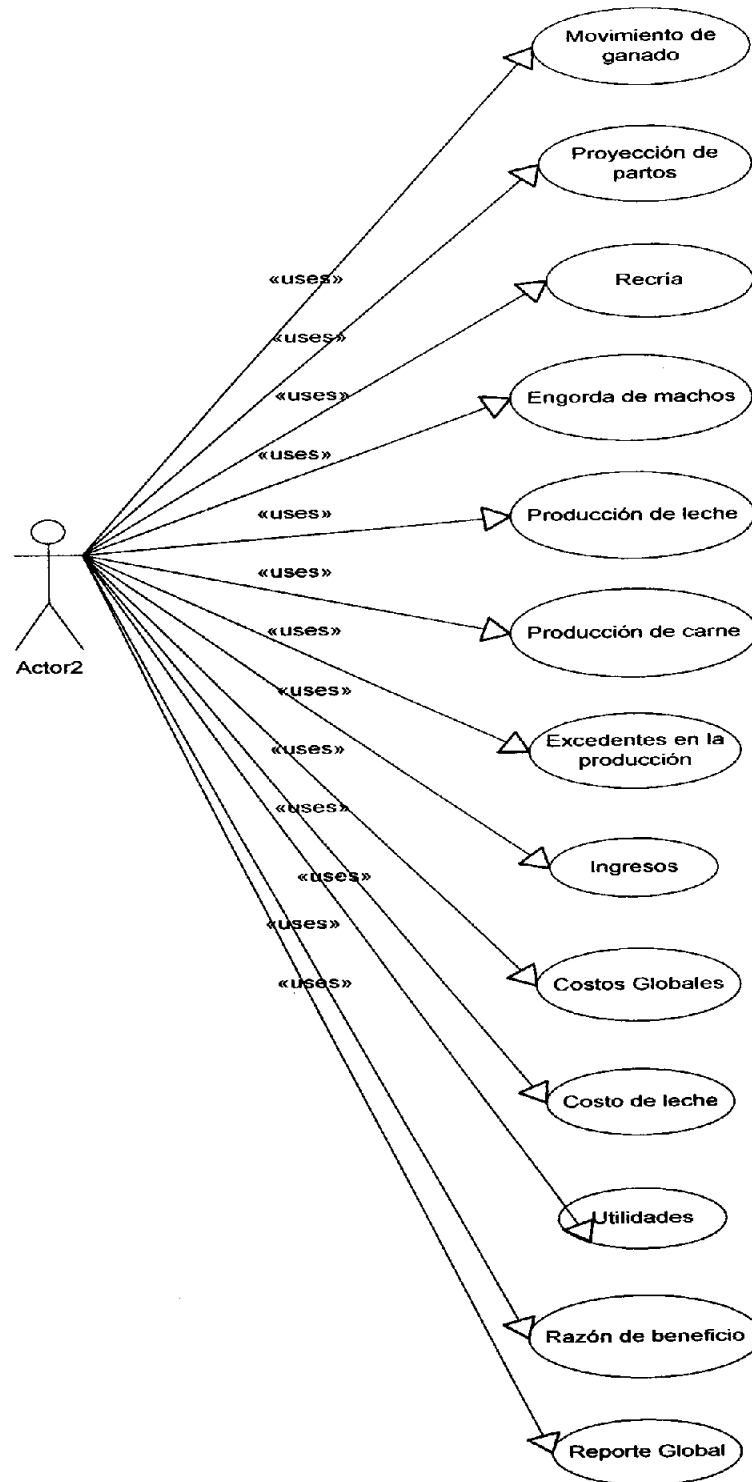


Figura 5.4. Diagrama de Casos del simulador 2do. Nivel Opción "Reportes"

5.2.6 Descripción de Casos de Uso

Para facilitar la comprensión de cada aspecto del sistema, se presenta a continuación una descripción de los casos de uso.

CASO DE USO	Supuestos Básicos
ACTORES	Usuario del simulador
FUNCIÓN	Permitir el manteniendo de los proyectos.
DESCRIPCIÓN	El usuario puede registrar nuevos proyectos, identificando todas sus características. También se pueden modificar algunas de sus características o eliminar un proyecto.
CASO DE USO	Reportes
ACTORES	Usuario del simulador
FUNCIÓN	Imprime el análisis del simulador
DESCRIPCIÓN	El usuario puede imprimir el resultado del análisis que el simulador ha realizado.
CASO DE USO	Movimiento de ganado
ACTORES	Usuario del simulador
DESCRIPCIÓN	El Simulador requiere información concerniente a la estructura del hato: Los apartados correspondientes son seis: <ul style="list-style-type: none"> A) Capacidad de la explotación. B) Edad del ganado en sus distintas etapas de desarrollo. C) Compra de ganado del 1er al 7º año, en las diferentes

etapas.

D) Inventario ganadero.

E) Intervalo entre partos.

F) Índices de mortalidad y desecho para las distintas etapas

CASO DE USO

ACTORES

DESCRIPCIÓN

Engorda de machos

Usuario del simulador

Comprende los siguientes apartados:

G) Edades de los machos en las etapas de cría y desarrollo.

H) Porcentaje de terneros conservados.

I) Índices de mortalidad y desecho para terneros y añojos.

CASO DE USO

ACTORES

DESCRIPCIÓN

Producción de leche

Usuario del simulador

Se abordan los supuestos básicos relativos a la cantidad de leche producida por el hato: intervalo entre partos (I.P.) sobre la producción de leche, debe indicarse el promedio estimado de la producción anual por vaca de 1er parto en el hato, y en seguida, el correspondiente a la producción anual por vaca de 2 o más partos.

CASO DE USO

ACTORES

DESCRIPCIÓN

Eliminación de ganado

Usuario del simulador

Con los datos proporcionados bajo este inciso, el usuario determina, para cada una de las categorías de ganado

establecidas, la proporción de los animales eliminados que espera comercializar, en atención a requerimientos de sanidad, etapa productiva y posibilidades de mercadeo.

CASO DE USO

Ingresos

ACTORES

Usuario del simulador

DESCRIPCIÓN

En este apartado se tiene información de mayor importancia para los ejercicios de simulación prospectiva de proyectos lecheros: los precios de venta de los productos más importantes de los subsistemas de producción de leche, recría de ganado productor y engorda de machos.

CASO DE USO

Costos globales

ACTORES

Usuario del simulador

DESCRIPCIÓN

En este apartado, se refiere a los costos anuales de diferentes bienes y servicios que serán gastados o utilizados en el proceso productivo del proyecto. Por lo tanto, se trata de costos estimados, no de costos reales.

Capítulo 6. Diseño del Sistema

Introducción

El propósito del diseño es crear una arquitectura para la implementación y establecer las políticas comunes que deben utilizarse por parte de elementos dispares del sistema. En esta etapa se examinan las conclusiones que han sido generadas a partir de las observaciones, especificaciones y análisis del problema, que se usan para identificar formas de soporte que son requeridas por la metodología del desarrollo.

6.1 Definición de Subsistemas

Se utilizará el patrón del *Modelo-Vista-Controlador*.

Sus características principales son que el Modelo, las Vistas y los Controladores se tratan como entidades separadas; esto hace que cualquier cambio producido en el Modelo se refleje automáticamente en cada una de las vistas

Modelo es el objeto que representa los datos del programa. Maneja los datos y controla todas sus transformaciones. El Modelo no tiene conocimiento específico de los Controladores o Vistas. Es el propio sistema el que tiene encomendado la responsabilidad de mantener enlaces entre el Modelo y sus Vistas, y notificar a las Vistas cuando cambia el Modelo.

Vista es el objeto que maneja la presentación visual de los datos representados por el Modelo. Genera una representación visual del Modelo. Y muestra los datos del usuario. Interactúa con el modelo a través de una referencia al propio modelo.

Controlador es el objeto que proporciona significado a las órdenes del usuario, actuando sobre los datos representados por el Modelo. Cuando se realiza algún cambio, entra en acción, bien sea por cambios en la información del Modelo o por alteraciones de la Vista. Interactúa con el Modelo a través de una referencia al propio modelo.

Al incorporar el modelo de Arquitectura MVC a un diseño, las piezas de un programa se pueden construir por separado y luego unirlas en tiempo de

ejecución. Si uno de los Componentes, posteriormente, se observa que funciona mal, puede remplazarse sin que las otras piezas se vean afectadas.

6.2 Mecanismos genéricos de diseño

Para las fuentes de manejo de la base de datos, se usarán los mecanismos implementados dentro del sistema manejador de bases de datos *DBase III*. Para el manejo y recuperación de errores, se usarán las rutinas de depuración de Visual Object. En cuanto a la administración de transacciones, se utilizará el manejador *DBase III* y en lo que corresponde a los mecanismos de seguridad, se aplicarán los sistemas de respaldo.

6.3 Posibilidades de reuso

Dado las características y los componentes de hardware y software, el sistema será totalmente portátil y reusable. Se generan bibliotecas de funciones generales, que permitan el desarrollo de otros módulos anexos a este sistema.

6.4 Diagrama de diseño

Se presentan los diagramas de paquetes y capas con más detalle, así como los diagramas de secuencia y diagramas de entidad-relación de las bases de datos usadas por el Simulador.

6.4.1 Diagrama de paquetes y capas

En la Figura 6.1 se muestra el diagrama de la definición de Arquitectura del Simulador: Modelo Vista Controlador

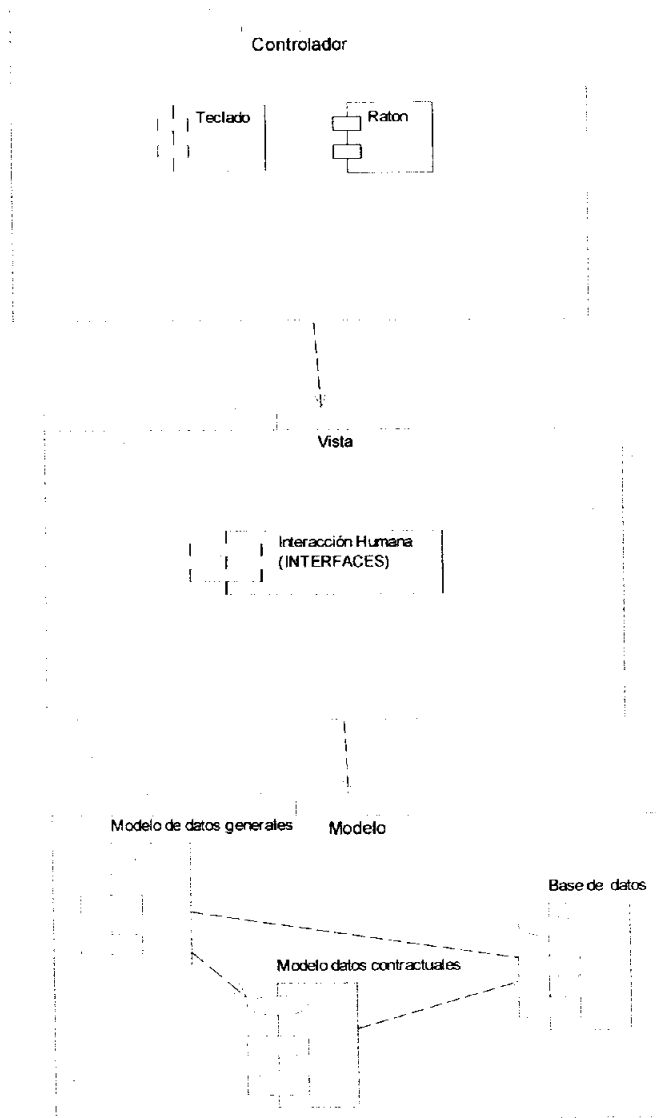


Figura 6. 1 Definición de Arquitectura del Simulador de Hatos Lecheros

En la Figura 6.2 se muestra el diagrama de paquetes y capas detallado. Arquitectura: Modelo-Vista-Controlador.

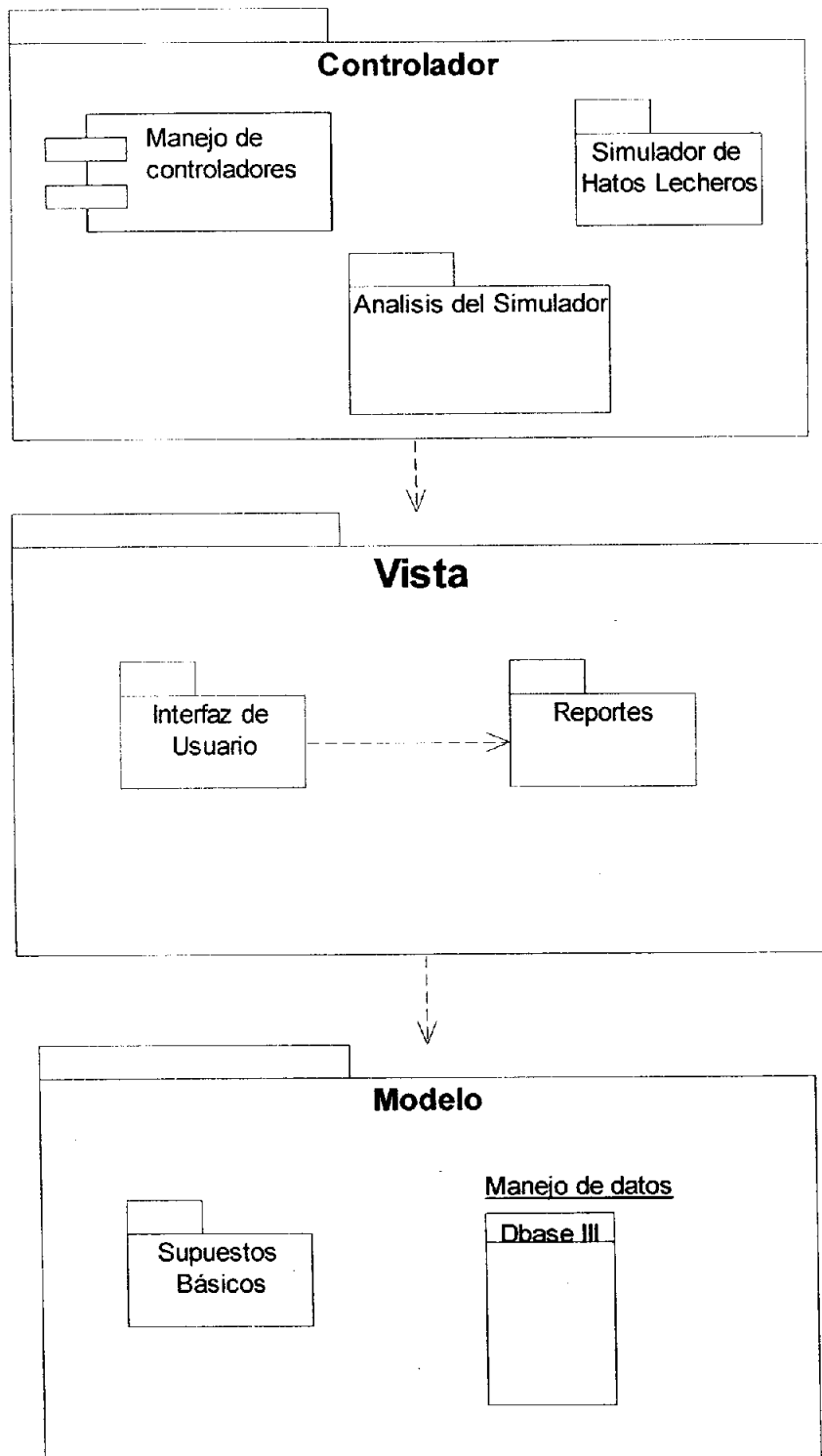


Figura 6. 2 Diagrama de Paquetes

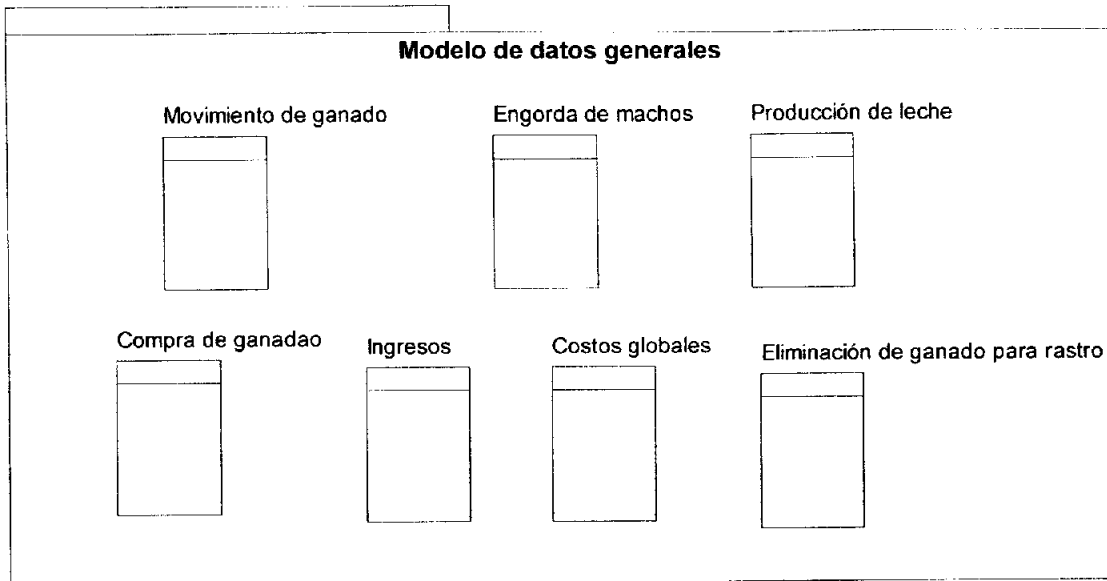


Figura 6. 3 Diagrama de modelo de datos generales

En la Figura 6.3 se describe el diagrama detallado del modelo de datos generales.

En la Figura 6.4 se presenta el diagrama detallado del manejo de datos.

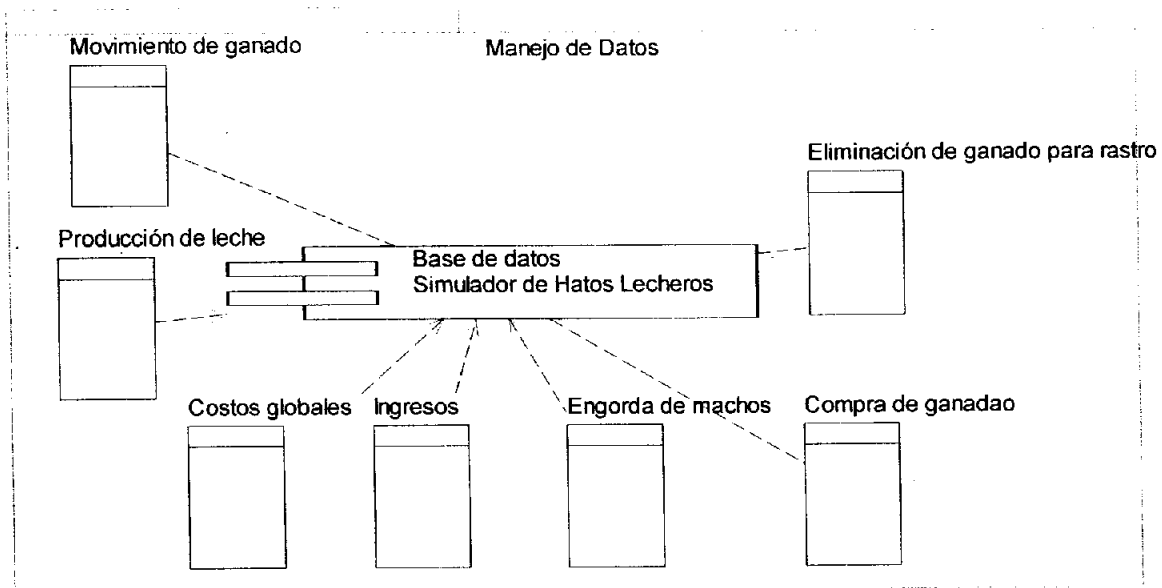


Figura 6. 4 Diagrama detallado del manejo de datos.

6.4.2 Diagramas de Entidad relación

Se muestra en la Figura 6.5 el diagrama entidad-relación que de forma preliminar establecen una estructura para los datos que manejará y almacenará el sistema.

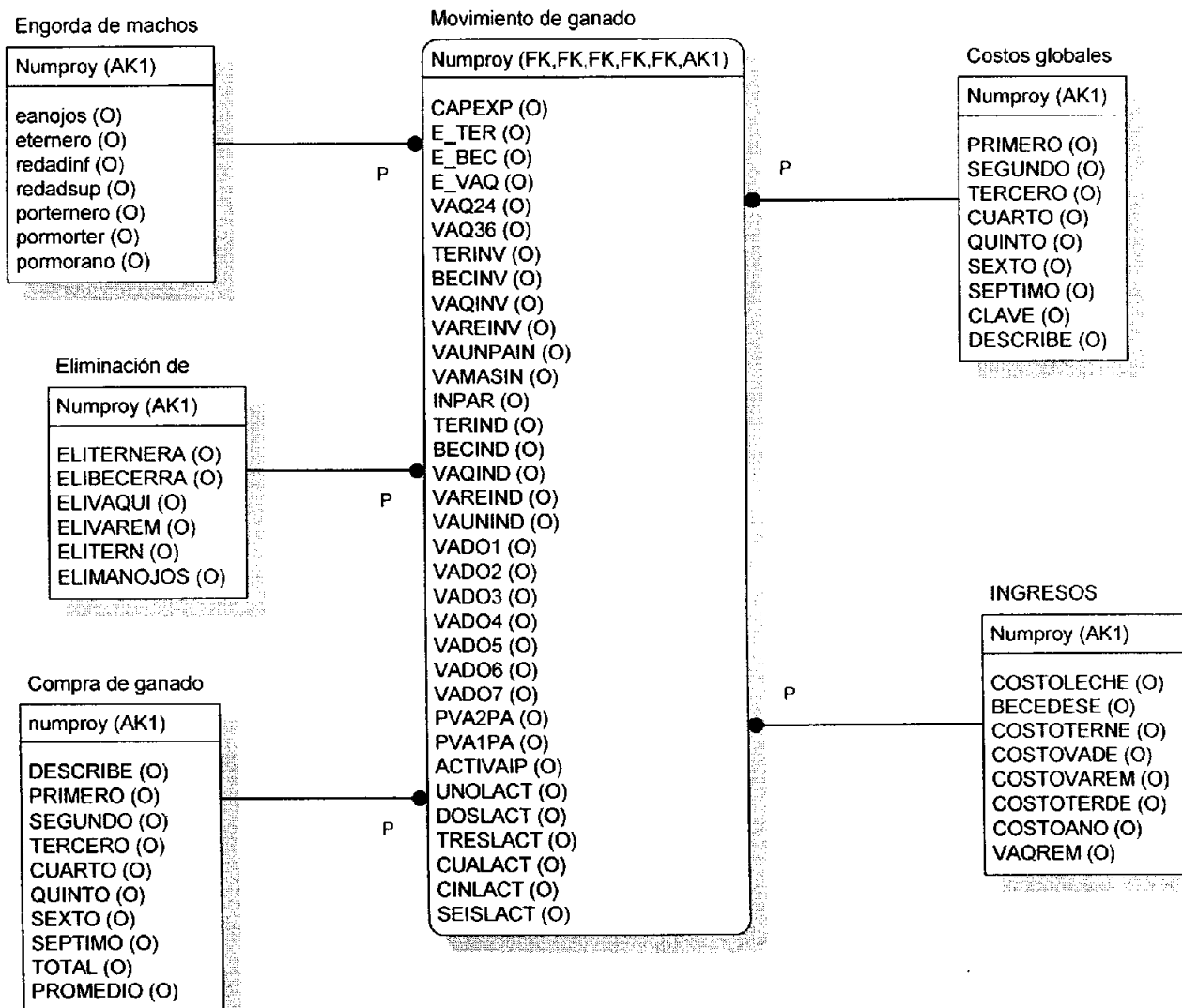


Figura 6. 5 Diagrama entidad-Relación

Capítulo 7. Prototipo del Sistema

El objetivo de este capítulo es presentar el resultado de la codificación del Simulador. Los prototipos son los productos primarios de la conceptualización. La creación de un prototipo es una atractiva idea para sistemas (Ver capítulo 2)

7.1 Prototipo

Se presentan algunas pantallas del prototipo preliminar del sistema de Simulador de Hatos Lecheros. La Figura 7.1 presenta la pantalla principal del Simulador así como la descripción general del mismo



Figura 7. 1. Simulador de Hatos Lecheros.

Supuestos Básicos

En la Figura 7.2 se muestra el menú principal, al cual se tiene acceso de la siguiente manera: dirigir el ratón en la opción Supuestos Básicos.

Mediante el Mousse ("ratón") se desplaza el cursor para seleccionar la opción. El menú principal tiene las siguientes opciones:

1. Movimiento de ganado.
2. Engorda de machos.
3. Producción de leche.
4. Eliminación de ganado para rastro.
5. Ingresos.
6. Costos globales.



Figura 7. 2 Menú Principal

Movimiento de ganado

Una vez que entramos a la primera opción del menú del Simulador como se ve en la Figura 7.3, el primer dato solicitado corresponde a la capacidad de la explotación, que se refiere al número total de vacas en ordeña y secas. Estos animales componen la base del

hato productor y reproductor.

En seguida, se anota la edad de las terneras al destete, dentro del rango de 2 a 3 meses, con un número entero o fraccionario. En la recría, las terneras comprenden el período que va del nacimiento al destete; las becerras, del destete a los 12 meses; y las vaquillas, de los 12 a los 24 meses.

Posteriormente, debe anotarse el rango de edad al parto de las vaquillas de reemplazo; o sea, el margen de edad en que se estima que ocurrirá la mayor parte de los partos de primerizas. El rango que maneja el Simulador es de 24 a 36 meses. El límite inferior no debe ser menor a 24, ni el límite superior mayor a 36.

Movimiento de ganado

A. Capacidad de la Explotación: 100

B. EDADES DEL GANADO

Terneras: 2.5
 Becerras: 2.5 11.9
 Vaquillas: 12 23.9

Vaquillas de reemplazo
 Límite inferior: 24
 Límite superior: 30

B. INVENTARIO GANADERO

Terneras: 100
 Becerras: 5
 Vaquillas: 4

Vaquillas de reemplazo: 10
 Vacas de primer parto: 70
 Vacas de 2 o más partos: 30

E. Intervalo entre partos: 16.0

F. Porcentaje de mortalidad y desecho para las distintas etapas

Terneras: 18.0
 Becerras: 6.0
 Vaquillas: 5.0
 Vaquillas de reemplazo: 2.0
 Vacas de 1er. parto: 2.0

Vacas de 2o. parto: 12.0
 1er. año: 12.0 5o. año: 22
 2o. año: 15.0 6o. año: 22.0
 3er. año: 17.0 7o. año: 22.0
 4o. año: 20.0

C. Compra de ganado del 1er. al 7o. año

Descripción	Primero	Segundo	Tercero	Cuarto	Quinto	Sexto	Séptimo
Terneras	100	30	5	25	30	5	10
Becerras	100	10	30	4	5	5	7
Vaquillas	10	5	30	20	10	8	9
Vaquillas de reemplazo	20	15	10	5	4	3	2
Vacas de primer parto	6	7	6	5	4	3	2
Vacas de 2o. o más partos	50	40	30	20	10	5	5

Cancelar
OK

Figura 7. 3 Supuestos Básicos.

Se inicia también el Apartado (C) compra de ganado del año 1 al 7 en las diferentes etapas, en lo correspondiente a terneras, becerras; vaquillas; vaquillas de reemplazo; vacas de primer parto; y vacas de 2 o más partos.

Después de llenar el Apartado C correspondiente a la compra de ganado programadas para los próximos 7 años, se pasa al Apartado (D) que se refiere al inventario ganadero de los animales de que dispone la explotación al momento de iniciar

la proyección para las diferentes categorías de ganado.

El intervalo entre partos, que aparece en el Apartado E (Figura 7.3), es el siguiente dato a introducir. Tras haber indicado el intervalo entre partos, el Simulador solicita al usuario el porcentaje de mortalidad y desecho para las distintas etapas de desarrollo, según aparece en el Apartado F (Figura 7.3). Este parámetro incluye de manera global las pérdidas de ganado por motivos de muerte y desecho.

Al pasar al Apartado F, el Simulador solicita para cada año de operaciones, el porcentaje de mortalidad y desecho de las vacas de dos o más partos.

Una vez que se introdujeron los datos en los apartados correspondientes se habilita el botón **OK**, presionarlo para guardar los datos.

Engorda de machos

El concepto engorda de machos del menú principal, comprende los siguientes apartados (Figura 7.4):

- G) Edades de los machos en las etapas de cría y desarrollo.
- H) Porcentaje de terneros conservados.
- I) Índices de mortalidad y desecho para terneros y añojos.

Engorda de machos en las etapas de cría y desarrollo

G. - EDADES DE LOS MACHOS EN LAS ETAPAS DE CRÍA

Terneros:

AÑOJOS (Novillos de uno a dos años)

RANGO DE EDAD A LA VENTA ENTRE 12 Y 24 MESES

Limite Inferior: Limite superior:

H. - PORCENTAJE DE TERNEROS CONSERVADOS

L. - % DE MORTALIDAD Y DESECHO

Terneros:

Añojos:

Ok Cancelar

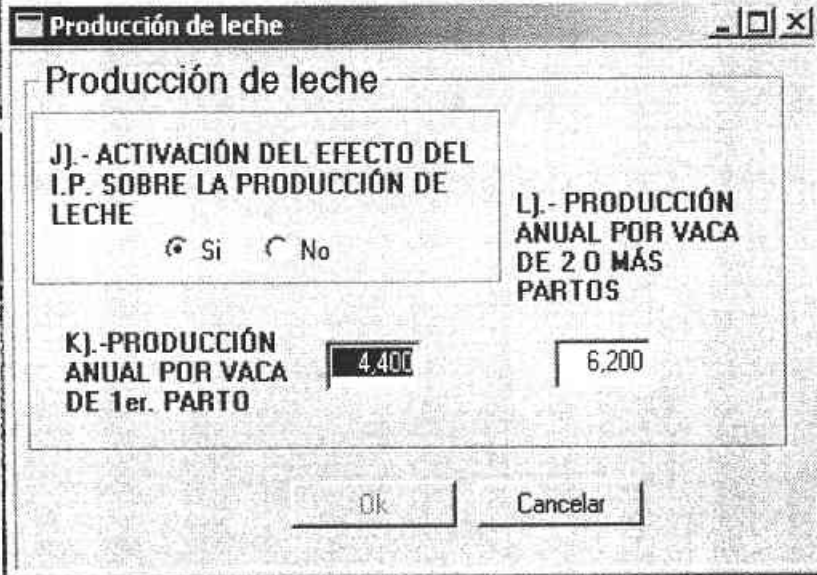
Figura 7. 4 Engorda de Machos

En el primer inciso del Apartado G, es preciso indicar la edad al destete de los terneros, la cual puede ser la misma que para las hembras, o diferente. Cualquier cifra

entera o fraccionaria, de 2 a 3. En seguida, debe anotarse el límite inferior del rango en que salen los añojos a la venta o se da por terminado su período de crianza y desarrollo. El límite inferior no debe ser menor a 12 meses ni mayor a 24. El límite superior puede ser igual o mayor al inferior, pero nunca menor.

Al pasar al Apartado H, debe marcarse el porcentaje de los terneros que se desea conservar para la cría de añojos (novillos de 12 a 24 meses), del total de terneros disponibles (destetados vivos); en tanto que en el Apartado I, el último del cuadro, debe apuntarse el porcentaje de mortalidad y desecho estimado para los terneros, del nacimiento al destete; y para los añojos, dentro del rango fijado.

Una vez que se introdujeron los datos en los apartados correspondientes se habilita el botón **OK**, presionarlo para guardar los datos.



The screenshot shows a window titled "Producción de leche" with a standard Windows-style title bar. The window contains the following elements:

- A header "Producción de leche" inside the window frame.
- A section for "J).- ACTIVACIÓN DEL EFECTO DEL I.P. SOBRE LA PRODUCCIÓN DE LECHE" with two radio buttons: "Si" (selected) and "No".
- A section for "K).- PRODUCCIÓN ANUAL POR VACA DE 1er. PARTO" with a text input field containing the value "4,400".
- A section for "L).- PRODUCCIÓN ANUAL POR VACA DE 2 O MÁS PARTOS" with a text input field containing the value "6,200".
- At the bottom, there are two buttons: "Ok" and "Cancelar".

Figura 7. 5 Producción de Leche

Producción de leche

En la Figura 7.5 se abordan los supuestos básicos relativos a la cantidad de leche producida por el hato, bajo tres Apartados (J, K y L). El primero de ellos consiste en la activación del efecto del intervalo entre partos (I.P.) sobre la producción de leche. Posteriormente, debe indicarse el promedio estimado de la producción anual por vaca de primer parto en el hato, y en seguida, el correspondiente a la producción anual por vaca de dos o más partos.

Una vez que se introdujeron los datos en los apartados correspondientes se habilita el botón **OK**, presionarlo para guardar los datos.

Eliminación de ganado para rastro

La proporción de los animales que puede enviarse al rastro para su sacrificio y comercialización debe definirse en el Apartado M Figura 7.6. Con los datos proporcionados bajo este inciso, el usuario determina, para cada una de las categorías de ganado establecidas, la proporción de los animales eliminados.

Una vez que se introdujeron los datos en los apartados correspondientes se habilita el botón **OK**, presionarlo para guardar los datos.

M). PORCENTAJE DE LAS PÉRDIDAS GLOBALES POR MUERTE Y DESECHO QUE PUEDE ENVIARSE AL RASTRO:	
Terneras:	10
Vaquillas de reemplazo:	90
Becerras:	10
Vacas:	90
Vaquillas:	90
Terneros:	100

OK Cancelar

Figura 7. 6 Eliminación de Ganado para Rastro.

Ingresos

El quinto punto del menú principal, tiene un solo Apartado, el N, en él se proporciona la información de mayor importancia para los ejercicios de simulación prospectiva de proyectos lecheros: los precios de venta de los productos más importantes de los subsistemas de producción de leche, cría de ganado productor y engorda de machos. Entre estos productos, el precio de la leche es el aspecto primordial; sin embargo, en general, el resto de los conceptos es también muy importante. Como se observa en la

Figura 7.7, el precio de la leche está referido a un litro.

En cuanto a los animales eliminados y enviados al rastro, así como de los añojos y vaquillas de reemplazo disponibles para la venta, los precios se cotizan por cabeza

Terneros. Son los animales sacrificados al poco de nacer.

Añojos. Son los novillos para carne que han alcanzado la edad adecuada para su venta, entre 12 y 24 meses.

Una vez que se introdujeron los datos en los apartados correspondientes se habilita el botón **OK**, presionarlo para guardar los datos.

Ingresos			
Un litro de leche:	3.00	Vaquilla de reemplazo:	5,000.00
Ternera de desecho:	500.00	Ternero de desecho:	500.00
Becerra de desecho:	700.00	Añojo:	6,000.00
Vaquilla de desecho:	990.00	Vaquilla de reemplazo:	10,000.00

Ok Cancelar

Figura 7. 7 Ingresos.

Costos globales

El Apartado O, se refiere a los costos anuales de diferentes bienes y servicios que serán gastados o utilizados en el proceso productivo del proyecto. Por lo tanto, se trata de costos estimados, no de costos reales. La Figura 7.8 presenta el Apartado O que comprende los costos anuales por concepto de alimentación, salarios, compra de ganado, medicinas y vacunas, otros costos, intereses de capital y amortizaciones.

Una vez que se introdujeron los datos en los apartados correspondientes se habilita el botón **OK**, presionarlo para guardar los datos.

O). COSTOS GLOBALES

Describe:	Primero:	Segundo:	Tercero:	Cuarto:	Quinto:	Sexto:	Séptimo:	Clave:
Alimentación	500000.00	500000.00	500000.00	00000.00	00000.00	0000.00	500000.00	
Salarios	200000.00	200000.00	200000.00	00000.00	00000.00	0000.00	200000.00	
Compra de ganado	100000.00	100000.60	100000.00	00000.00	00000.00	0000.00	100000.00	
Medicinas y vacunas	60000.00	60000.60	60000.00	60000.00	60000.00	0000.00	60000.00	
Otros costos	50000.00	50000.00	50000.60	50000.00	50000.00	0000.00	50000.00	
Interes de capital	20000.00	20000.60	20000.00	20000.00	20000.00	0000.00	20000.00	
amortizaciones	80000.00	80000.60	80000.00	80000.00	80000.00	0000.00	80000.00	

OK Cancel

Figura 7. 8 Costos Globales.

Opción Reportes del Menú principal

La opción de reportes contiene 13 tópicos principales, mediante los cuales el usuario obtiene información que le permitirá analizar el desarrollo futuro de una explotación lechera, su viabilidad zootécnica, operativa y económica. En el marco de los escenarios alternativos construidos a partir de los supuestos básicos.

El inciso I comprende siete rubros que cubren información sobre el comportamiento de la estructura del hato; del punto II al VII el Simulador proyecta datos relativos al desempeño reproductivo y productivo del hato; y los seis últimos tópicos, del VIII al XII, generan proyecciones sobre el desenvolvimiento económico de la explotación y el XIII presenta de una manera global todo el análisis correspondiente al simulador.

La Figura 7.9 presenta la opción reportes del menú que comprende 19 salidas de información.



Figura 7. 9 Menú de los reportes

I. Movimiento de ganado

Movimiento de ganado se descompone para su análisis en ocho Apartados: (a) compras anuales, (b) ganado no comprado en la etapa —recrea del hato—, (c) población inicial del ganado productor —vacas en ordeño y secas—, (d) estructura del hato, (e) muertes y desechos del ganado productor, (f) población final del ganado productor, (g) tamaño del hato y (h) capacidad disponible.

Compras Anuales									
A.- Compras anuales de ganado del 1er. al 7o. año									
Describe	Primero	Segundo	Tercero	Cuarto	Quinto	Sexto	Séptimo	Total	Promedi
Terneras	100	30	5	25	30	5	10	205.00	29.
Becerras	100	10	30	4	5	6	7	162.00	23.
Vaquillas	10	5	30	20	10	8	9	92.00	13.
Vaquillas de reemplazo	20	15	10	5	4	3	2	59.00	8.
Vacas de primer parto	8	7	6	5	4	3	2	35.00	5.
Vacas de 2o. o más partos	50	40	30	20	10	5	5	160.00	22.

Figura 7. 10 Reporte de Compras Anuales de Ganado del Primer año al Séptimo año

a.- Compras anuales

Bajo este rubro el Simulador presenta de manera ordenada, la misma información que le fuera proporcionada en el Apartado C de la Figura 7.3, del menú principal, sobre las compras anuales de ganado, para las diferentes categorías y en un período de siete años. El Simulador no ofrece aquí ninguna información adicional a la que el usuario ya tiene únicamente la retoma y expone dentro de sus propios formatos.

En la Figura 7.10 se muestra el formato con el que aparecen las compras anuales de ganado: de terneras, becerras, vaquillas, vaquillas de reemplazo, vacas de 1er parto y vacas de dos o más partos, como en todos los cuadros de los reportes, hay dos botones: imprimir, que sirve para imprimir el cuadro y cancelar, que se usa para regresar al menú principal del programa.

b.- Ganado no comprado en la etapa (recría del hato)

En la Figura 7.11, presenta información sobre los animales en las distintas etapas de crianza, disponibles anualmente en la explotación a través de la recría, en la medida en que ésta se lleve a cabo: en cuanto a extensión, por las etapas; y nivel intensivo, por el número de cabezas producidas en cada etapa del ciclo parcial y en todas ellas del ciclo global.

Ganado de recría no comprado									
B.- Ganado de recría no comprado									
Describe:	Primero:	Segundo:	Tercero:	Cuarto:	Quinto:	Sexto:	Séptimo:	Total	Promedio:
Terneras	122	67	82	91	96	98	100	656.00	93.66
Becerras	155	107	81	99	111	99	100	751.00	107.33
Vaquillas	53	231	100	92	103	113	97	789.00	112.65
Vaquillas de reemplazo	12	38	173	155	105	103	110	695.00	99.35
Total	342	444	436	436	415	412	407	2891.00	412.99

Figura 7. 11 Reporte de Ganado no Comprado en la Etapa

c.- Población inicial del ganado productor

Este rubro proporciona información sobre la cantidad de vacas que tiene la explotación al comienzo de cada uno de los siete años que abarca la proyección. La Figura 7.12 se refiere a las vacas de 1er parto, a las vacas de 2 o más partos, y al total de vacas en el hato, respectivamente.

Los montos de población inicial del ganado productor están determinados directamente por: (1) las compras de ganado de reemplazo y/o en producción, (2) la capacidad de la recría y (3) el tamaño y conformación del hato productor al inicio de la proyección; e indirectamente por: (1) el índice de intervalo entre partos y (2) las tasas de mortandad y desecho en las diferentes categorías de ganado, tanto de la recría como del hato productor.

Conforme las vaquillas de reemplazo paren, lo cual ocurre el mismo año en que aparecen calculadas, pasan a formar parte de la población de vacas de primer, si la capacidad de la explotación no ha sido cubierta; de lo contrario, esos animales se tornarían en excedentes para venta.

Las vaquillas de reemplazo pueden provenir de una o más de las siguientes tres vías: (1) la compra de animales a parto, (2) la recría de la propia explotación, y (3) el pie de cría con que arranca la operación de la vaquería o se inicia el ejercicio de planeación.

Por otra parte, la población final de vacas, es decir, el total de vacas con que cuenta la explotación al finalizar cada año, habiendo sufrido pérdidas por mortalidad y desecho, se computa como parte de la población inicial del ganado productor para el año siguiente, según se trate de vacas de primer parto o de dos o más partos.

C.- Población inicial del ganado productor									
Describe	Primero	Segundo	Tercero	Cuarto	Quinto	Sexto	Séptimo	Total	Prom
Vacas de primer parto	78	29	20	19	21	23	22	212.00	
Vacas de 2 o más partos	80	176	221	247	259	265	273	1521.00	21
Total	158	205	241	266	280	288	295	1733.00	24

Figura 7. 12 Reporte Población Inicial del Ganado Productor

d.- Estructura del hato

A través de este apartado se obtiene información sobre la composición del hato, en lo relativo a los dos subsistemas pecuarios más importantes de las explotaciones lecheras: la recría y la producción de leche. Se describen los diferentes conceptos en que fue desagregada la estructura del hato para facilitar tanto su comprensión y análisis como el manejo del Simulador.

En la Figura 7.13 se presenta información de salida de la estructura del hato.

- Terneras acumuladas. Son las crías registradas en la etapa de lactancia (del nacimiento al destete) durante el año de cómputo, que en su momento llegan al destete. Incluyen las terneras disponibles al inicio de la proyección, las compradas, las nacidas en la explotación y, de estas últimas, las del año anterior que no concluyeron la etapa.
- Becerras acumuladas. Son los animales registrados en la etapa de

desarrollo 1 (del destete a los 12 meses de edad) durante el año de cómputo, que se mantienen vivos. Los montos observados responden a las becerras disponibles al inicio de la proyección, las compradas, las nacidas en la explotación y, de éstas, las que continúan en la etapa de un año a otro.

- Vaquillas acumuladas. Son los animales registrados que entran a la etapa de desarrollo 2 (mayores de 12 meses y menores de 24) durante el año de cómputo y alcanzan los 24 meses de edad, ese mismo año o el siguiente. Las estimaciones consideran: las vaquillas disponibles al inicio de la proyección, las compradas, las nacidas en la explotación y, de éstas, las del año anterior que se mantienen en la etapa al inicio del año de cómputo.
- Media anual de vacas de 1er parto. Estimación de la media anual de las vacas vivas de 1er parto.
- Media anual de vacas de dos o más partos. Cálculo de la media anual de las vacas vivas de dos o más partos.
- Media anual del total de vacas. Cómputo de la media anual de todas vacas vivas (uno o más partos).

Estructura del Hato									
D.- Estructura del hato									
Describe:	Primero:	Segundo:	Tercero:	Cuarto:	Quinto:	Sexto:	Séptimo:	Total	Promedio:
Termeras Acumuladas	204	134	105	129	144	127	130	973.00	138.96
Becerras acumuladas	249	312	201	188	195	195	188	1528.00	218.29
Vaquillas acumuladas	62	298	385	318	266	270	263	1863.00	266.16
Vacas de primer parto	83	42	39	44	50	52	52	361.00	51.60
Vacas de 2 o más partos	75	163	202	222	230	236	243	1372.00	195.98
TOTAL de vacas (m.a.):	158	205	241	266	280	288	295	1733.00	247.57

Figura 7. 13 Reporte Estructura del Hato

e.- Muertes y desechos del ganado productor

Aquí se reportan las bajas esperadas anualmente de vacas de primer parto, vacas de dos o más partos y todas las vacas, como se puede ver en la Figura 7.14. Las cifras observadas en los tres renglones del inciso, incluyen pérdidas por muerte y desecho. Evidentemente, estas pérdidas se traducen en la necesidad de reemplazos, que habrá que adquirir por la vía de compras o de la recría propia. Si no se cuenta con un sistema de recría eficiente o las bajas en el hato productor son demasiado altas, las compras de ganado de reposición tendrán que aumentar, pudiendo convertirse en la principal fuente de repoblación, e incluso en la única. Esta interconexión subraya el hecho de que la recría y la producción de leche, o la producción de leche y la recría, en principio, son parte de un continuo pecuario fundamental, no sólo para la modernización de la ganadería lechera de cualquier explotación, región o país, sino también y sobretodo para poder integrar mejores y más coherentes sistemas productivos: pecuarios, agropecuarios, eco-agropecuarios y socio-eco-agropecuarios.

Describe	Primero	Segundo	Tercero	Cuarto	Quinto	Sexto	Séptimo	Total	Promed
Vacas de primer parto	2	1	0	0	0	0	0	4.00	0
Vacas de 2 o más partos	10	26	38	49	57	58	60	298.00	42
Total	11	27	38	50	57	58	60	303.00	43

Figura 7. 14 Reporte Muertes y Desechos del Ganado Productor.

f. Población final del ganado productor

En la dinámica del ciclo de vida reproductivo-productivo de las vacas lecheras, eventualmente se registran bajas, que el Simulador reporta como muertes y desechos del ganado productor como se vio en el Apartado E. Estas bajas van mermando la población

del hato desde el comienzo de cada período analizado, hasta que concluye éste, con la población —final— resultante.

Población final del hato									
F.- Población final del ganado productor									
Describe	Primero	Segundo	Tercero	Cuarto	Quinto	Sexto	Septimo	Total	Promedio
Vacas de primer parto	88	55	57	69	78	81	82	510.00	72.90
Vacas de 2 o más partos	70	150	184	197	202	207	213	1223.00	174.67
Total	158	205	241	266	280	288	295	1733.00	247.57

Figura 7. 15 Reporte Población final del ganado pProductor.

El cuadro que comprende este apartado de la Figura 7.15 muestra la cantidad de vacas con que cuenta la explotación al término de cada año de la proyección, considerando las vacas de primer parto, las vacas de dos o más partos y el total de vacas. La población final de un año se convierte en la población inicial del siguiente.

g. Tamaño del hato

La salida y entrada de animales al hato productor da pie a cambios más o menos frecuentes en la conformación estructural de éste, tanto en lo cualitativo (edad, calidad, estado productivo, condiciones de vigor y salud) como en lo cuantitativo (número de cabezas). El tamaño del hato es un referente a este segundo aspecto: la cantidad de vacas con que cuenta la explotación. Como se aprecia en la Figura 7.16, las cifras se manejan en ocho filas y cuatro columnas. Las siete primeras filas corresponden a los años que abarca la proyección, en tanto que la última, alude al tamaño del hato inicial (H.I.). Los datos obtenidos se ordenan en tres columnas, que describen el tamaño del hato alcanzado en el año de cómputo, o en su caso por el H.I., con respecto a la capacidad instalada de la explotación para el ganado productor. En la primera de estas columnas se indica el máximo —mayor tamaño— registrado; en la segunda, aparece el mínimo —menor tamaño—; y en la tercera, el valor medio —promedio aritmético— de los

dos anteriores. Como puede apreciarse, el tamaño del hato relativo al H.I. es el mismo en las tres columnas, ello obedece a que el H.I. representa la cantidad de vacas de que dispone la explotación al inicio de la proyección (primer día del año 1). Esta cifra es fija, como también lo es la capacidad de la explotación.

Los valores observados siempre serán positivos, salvo que no haya ganado alguno, en cuyo caso serían de cero. Evidentemente, entre más se acerque el tamaño del hato (máximo, mínimo y medio) a la capacidad de la explotación, mayor será el aprovechamiento de los recursos físicos, biológicos y humanos, y en consecuencia, menores los costos de operación y mayor el rendimiento de utilidades. Las cifras superiores a las de la capacidad de la explotación revelan errores, que pueden estar en la cifra del H.I., en los montos de las compras de ganado, en las tasas de mortalidad y desecho del ganado adulto, o en todos ellos.

G.- Tamaño del hato									
Describe:	Primera:	Segundo:	Tercero:	Cuarto:	Quinto:	Sexto:	Séptimo:	Total:	Promedio:
Máximo	158	205	241	266	280	288	295	100.00	262.00
Mínimo	158	205	241	266	280	288	295	100.00	262.00
Medio	158	205	241	266	280	288	295	100.00	262.00

Figura 7. 16 Reporte Tamaño del Hato.

h. Capacidad disponible

En este inciso se despliega la misma información que en el anterior, pero vista desde otra perspectiva. La capacidad disponible muestra en qué medida la capacidad de la explotación está siendo desaprovechada; las vacas que faltan. Como puede apreciarse en la Figura 7.17, los datos se exponen dentro del mismo esquema de filas y columnas del cuadro anterior; sin embargo, en la capacidad disponible los valores se optimizan en su proximidad al cero, y no al número de cabezas con que se define la capacidad de la explotación para el ganado productor, que es el efecto buscado en el tamaño del hato.

H.- Capacidad disponible									
Describe:	Primero:	Segundo:	Tercero:	Cuarto:	Quinto:	Sexto:	Séptimo:	Total	Prome
Máxima	-58	-105	-141	-166	-180	-188	-195	-30.00	-15
Mínima	-58	-105	-141	-166	-180	-188	-195	-30.00	-15
Media	-58	-105	-141	-166	-180	-188	-195	-30.00	-15

Figura 7. 17 Reporte Capacidad Disponible.

Así, en la medida en que las cifras relativas a la capacidad disponible se aproximan al monto de la capacidad de la explotación, se desaprovechan los recursos de operación. Cuando son iguales, la explotación está vacía; cuando llegan a cero, está saturada. Las cifras negativas denotan errores que obedecen a un excesivo tamaño del hato, y pueden deberse a los problemas ya expuestos. El ganado no puede estar donde no cabe, ni debe estar donde no le corresponde.

Para cada uno de los siete años incluidos, se reporta el número de espacios vacíos: el valor máximo (la cifra mayor), el mínimo (la cifra menor) y el medio (el promedio aritmético). Las cifras relativas al H.I. son, como las del tamaño del hato, las mismas. Para reducir la capacidad disponible y elevar el uso eficiente de los recursos puede optarse por:

- Reducir el intervalo entre partos, que supone mejorar los parámetros reproductivos en general, o algunos de ellos en particular, mediante la optimización de las condiciones y procesos de los distintos subsistemas involucrados
- Disminuir las tasas de mortalidad y desecho; es decir, las muertes (provocadas por enfermedades, trastornos o accidentes), y la eliminación de ganado (asociada a problemas de infertilidad o baja producción). Para ello deben eliminarse, o por lo menos aminorarse, las

causas de mortalidad en las diferentes categorías de ganado, y elevar la vida productiva de los animales, lo cual es más complejo, porque usualmente ésto implica un rediseño del Sistema productivo en su conjunto.

- Aumentar las compras, cuando no podemos o no queremos hacer lo anterior.

Más allá de estas soluciones cabría intentar aumentar la rentabilidad de los distintos subsistemas de la explotación elevar el valor agregado de la leche, diversificar la producción (leche, carne, subproductos pecuarios y agrícolas), o hacer más competitivos los productos en el mercado (estrategias de comercialización, promoción, envasado). Conforme la rentabilidad de la explotación aumenta, ésta podrá crecer; en cambio, crecer no siempre conduce a un incremento de la rentabilidad.

Si nada de lo anterior resulta, entonces podría optarse por mantener, reducir o eliminar todo el hato, y por consiguiente el proyecto, dado que las condiciones de la producción o de la economía así lo imponen y no podemos resolverlas.

II. Proyección de partos

El parto marca el evento con que culmina el proceso reproductivo y se desencadena una serie de fenómenos simultáneos de gran importancia: termina la gestación, nace una cría, sobreviene la lactancia y empieza el puerperio. El animal pasa súbitamente de una condición reproductiva, grávida, a otra, ingrávida; y de un estado fisiológico-productivo relativamente pasivo, el período seco, a otro sumamente activo, la lactancia.

Así, al parto concurren, y de él se desprenden, los beneficios fundamentales de las buenas vacas lecheras: abundante leche y nuevas crías. Detrás de esta noción extraída de las bondades del parto, existen muchos otros aspectos de gran relevancia, tanto en lo que se refiere a la productividad de las vacas, como al rendimiento de las explotaciones lecheras, en sus respectivos ciclos de vida.

En la Figura 7.18 el Simulador presenta datos sobre la proyección de partos, en vacas de primer parto, vacas de dos o más partos y todas las vacas, a lo largo de los siete años de cómputo, además del total de partos acumulado en este período y del promedio anual esperado.

Partos									
II.- Proyección de partos									
Describe:	Primero:	Segundo:	Tercero:	Cuarto:	Quinto:	Sexto:	Séptimo:	Total:	Prome
Vacas de pimer parto	41	42	48	55	61	62	62	370.00	5
Vacas de 2 o más partos	56	122	152	166	173	177	182	1029.00	14
Total	97	164	200	221	234	239	244	1399.00	19

Figura 7. 18 Reporte Proyección de Partos

III. Recría

En el Apartado relativo a la recría, el Simulador despliega información sobre los inventarios de ganado en las diferentes etapas del ciclo de crianza. Estas etapas, aunque anteriormente se abordaron, las retomamos aquí con algunas variantes. Como podrá observarse, los rubros que atañen a los animales acumulados en la recría, aparecen también en la estructura del hato, donde se tienen como parte del inventario general de la explotación, aun cuando provienen del proceso de cría y desarrollo. Hechas las aclaraciones pertinentes, en la Figura 7.19 se presenta la relación de conceptos que comprende este Apartado:

- Terneras en etapa de lactancia: del nacimiento al destete de dos a tres meses. Son las crías que entran a la etapa de lactancia a lo largo del año y se mantienen vivas. Incluyen las terneras nacidas en la explotación, las disponibles al inicio de la proyección para el primer año y las compradas.

Recría									
III.- Recría									
Describe:	Primero:	Segundo:	Tercero:	Cuarto:	Quinto:	Sexto:	Séptimo:	Total:	Promedio:
Terneras	204	92	86	111	121	102	108	824	117.67
Terneras Acumuladas	204	134	105	129	144	127	130	973	138.96
Becerras	249	117	109	102	115	104	106	904	129.08
Becerras acumuladas	249	312	201	188	195	195	188	1,528	218.29
Vaqs. (12 ms-24ms)	62	236	128	111	113	120	106	876	125.13
Vaqs. acumuladas	62	298	385	318	266	270	263	1,863	266.16
Vaqs. reep. 24 ms	8	55	231	116	100	105	113	728	103.99
Vaqs. reep. (Interv.)	32	53	182	160	109	106	112	753	107.61
Total	1,070	1,296	1,429	1,235	1,163	1,129	1,126	8,448	1,206.89

Figura 7. 19 Reporte Recría

- Terneras acumuladas. Son las terneras que nacen durante el año de cómputo, más las del año pasado que continúan en la etapa.
- Becerras en etapa de desarrollo 1: del destete a los 12 meses. Son los animales nacidos en el curso del año de cómputo o a finales del año anterior, que llegan a la segunda fase del proceso de crianza y se mantienen vivos. Los montos observados responden a las becerras nacidas en la explotación, las disponibles al inicio de la proyección para el primer año y las compradas.
- Becerras acumuladas. Son las becerras registradas durante el año de cómputo en la etapa de desarrollo 1, y las del año previo, que no han salido aún.
- Vaquillas en etapa de desarrollo 2: mayores de 12 meses y menores de 24. Son los animales nacidos el año anterior al estimado, que entran a la etapa de desarrollo 2 el año de cómputo y continúan vivos. Las

	CONC./AÑO	1	2	3	4	5	6	7	TOTAL
<p>12. Costo/ft (\$)</p> <p>6</p>		<p>=SI(\$B\$80=1,\$M\$35,SI(\$B\$80=2,\$M\$36,SI(\$B\$80=3,\$M\$37,SI(\$B\$80=4,\$M\$38,SI(\$B\$80=5,\$M\$39,SI(\$B\$80=6,\$M\$40,SI(\$B\$80=7,\$M\$41,SI(\$B\$80>=8,\$M\$42)))))))/C\$52</p>	<p>=SI(\$B\$80=1,\$N\$35,SI(\$B\$80=2,\$N\$36,SI(\$B\$80=3,\$N\$37,SI(\$B\$80=4,\$N\$38,SI(\$B\$80=5,\$N\$39,SI(\$B\$80=6,\$N\$40,SI(\$B\$80=7,\$N\$41,SI(\$B\$80>=8,\$N\$42)))))))/D\$52</p>	<p>=SI(\$B\$80=1,\$O\$35,SI(\$B\$80=2,\$O\$36,SI(\$B\$80=3,\$O\$37,SI(\$B\$80=4,\$O\$38,SI(\$B\$80=5,\$O\$39,SI(\$B\$80=7,\$O\$41,SI(\$B\$80>=8,\$O\$42)))))))/E\$52</p>	<p>=SI(\$B\$80=1,\$P\$35,SI(\$B\$80=2,\$P\$36,SI(\$B\$80=3,\$P\$37,SI(\$B\$80=4,\$P\$38,SI(\$B\$80=5,\$P\$39,SI(\$B\$80=6,\$P\$40,SI(\$B\$80=7,\$P\$41,SI(\$B\$80>=8,\$P\$42)))))))/F\$52</p>	<p>=SI(\$B\$80=1,\$Q\$35,SI(\$B\$80=2,\$Q\$36,SI(\$B\$80=3,\$Q\$37,SI(\$B\$80=4,\$Q\$38,SI(\$B\$80=5,\$Q\$39,SI(\$B\$80=6,\$Q\$40,SI(\$B\$80=7,\$Q\$41,SI(\$B\$80>=8,\$Q\$42)))))))/G\$52</p>	<p>=SI(\$B\$80=1,\$R\$35,SI(\$B\$80=2,\$R\$36,SI(\$B\$80=3,\$R\$37,SI(\$B\$80=4,\$R\$38,SI(\$B\$80=5,\$R\$39,SI(\$B\$80=6,\$R\$40,SI(\$B\$80=7,\$R\$41,SI(\$B\$80>=8,\$R\$42)))))))/H\$52</p>	<p>=SI(\$B\$80=1,\$S\$35,SI(\$B\$80=2,\$S\$36,SI(\$B\$80=3,\$S\$37,SI(\$B\$80=4,\$S\$38,SI(\$B\$80=5,\$S\$39,SI(\$B\$80=6,\$S\$40,SI(\$B\$80=7,\$S\$41,SI(\$B\$80>=8,\$S\$42)))))))/I\$52</p>	<p>=SI(\$B\$80=1,\$T\$35,SI(\$B\$80=2,\$T\$36,SI(\$B\$80=3,\$T\$37,SI(\$B\$80=4,\$T\$38,SI(\$B\$80=5,\$T\$39,SI(\$B\$80=6,\$T\$40,SI(\$B\$80=7,\$T\$41,SI(\$B\$80>=8,\$T\$42)))))))/J\$52</p>

Apéndice B

Glosario

1. **ACOPLAMIENTO.** Medida del intercambio de información entre módulos de un sistema de software durante la fase de diseño.
2. **ADT («Abstract Data Type»).** Tipo abstracto de datos. Concepto empleado en programación y base teórica de los métodos de desarrollo orientados a objetos.
3. **ANIMACIÓN.** Técnica de validación de un sistema de software por el que se visualiza la evolución dinámica del sistema mediante la ejecución de un modelo del mismo.
4. **ARQUITECTURA SOFTWARE.** Descripción de los módulos de un sistema de software y su relación.
5. **CAIE («Computer Aided Innovation Engineering »).** Conjunto de herramientas de software para el soporte del proceso de innovación.
6. **CASE («Computer Aided Software Engineering»).** Conjunto de herramientas de software integradas para apoyar el desarrollo de un sistema de software.
7. **CALIDAD DE UN PROCESO SOFTWARE.** Grado en el que el proceso realiza la función para la que se ha definido.
8. **CALIDAD DE UN PRODUCTO SOFTWARE.** Grado en el que satisface las expectativas planteadas por los usuarios.
9. **CASO DE PRUEBA.** Definición de una prueba concreta que debe superar un sistema de software.
10. **COHESIÓN.** Medida de la relación existente entre las funciones que se incluyen dentro de un módulo.
11. **CMM («Capability Maturity Model»).** Modelo de madurez de procesos de desarrollo de software propuesto en EEUU por el Instituto de Ingeniería de Software.
12. **ENTORNO DE PROGRAMACIÓN.** Sistema CASE que soporta la fase de implementación; por extensión, cualquier sistema CASE.
13. **ESA («European Space Agency»).** Agencia Espacial Europea.
14. **IEEE («Institute of Electrical and Electronics Engineering»).** Instituto de Ingenieros Eléctricos y Electrónicos. Organización profesional de EEUU.

-
- 15. INGENIERÍA DE SISTEMAS DE SOFTWARE.** Aplicación de la ingeniería de sistemas al desarrollo de un sistema de software. Conjunto de técnicas de desarrollo y procedimientos de gestión necesarios para el desarrollo y mantenimiento de un sistema de software para obtener un sistema de calidad optimizando los recursos disponibles.
- 16. INTEGRACIÓN.** Relación existente entre las herramientas de un sistema CASE. Se definen cuatro niveles de integración: visual, de datos, de control y de proceso.
- 17. ISO («International Standard Organization»).** Organización Internacional de Normas.
- 18. LOTOS («Language for Temporal Ordering of Specifications»).** Norma internacional propuesta por la ISO para la especificación formal de sistemas de comunicaciones.
- 19. MÉTRICAS DE DESARROLLO SOFTWARE.** Factores susceptibles de ser medidos cuantitativamente en un sistema de software con el fin de evaluar algunos parámetros de calidad del mismo.
- 20. MODELO DE CICLO DE VIDA.** Secuencia de fases ordenadas en el tiempo y las relaciones entre ellas que sirven de marco de referencia para el desarrollo de un sistema de software.
- 21. MODELO DE SÍNTESIS AUTOMÁTICA.** Modelo de ciclo de vida en el que mediante el empleo de métodos formales es posible generar automáticamente el sistema de software a partir de la especificación del mismo.
- 22. MODELO EN CASCADA.** Modelo de ciclo de vida convencional en el que el desarrollo se realiza en una serie de fases en las que, en cada una de ellas, se parte de los resultados alcanzados en la anterior
- 23. MODELO EN ESPIRAL.** Modelo de ciclo de vida orientado al control de riesgos propuesto por Boehm en el que el desarrollo se realiza en varios ciclos. Considerado como meta-modelo al permitir el uso de cualquiera de los modelos de ciclo de vida.
- 24. PROCESO.** Conjunto de actividades orientadas a un fin concreto dentro del desarrollo o gestión de un sistema de software.
- 25. PROTOTIPADO.** Técnica de desarrollo de software en el que se genera un sistema incompleto con el fin de ayudar a completar la especificación de requisitos o la arquitectura del sistema.
-

- 26. PROTOTIPADO INCREMENTAL.** Técnica de desarrollo en la que se realizan diversos prototipos que van acercándose a la funcionalidad del sistema final.
- 27. PROTOTIPO.** Sistema de software construido de forma rápida cuya misión es ayudar a fijar la funcionalidad deseada del sistema final. Utilizado como base para las técnicas de prototipado.
- 28. PROTOTIPO HETEROGÉNEO.** Prototipo constituido por componentes a diferentes niveles de abstracción que cooperan para ofrecer la funcionalidad deseada por el usuario.
- 29. PUNTOS DE FUNCIÓN.** Métrica empleada para conocer la complejidad de un sistema basada en el número de unidades funcionales que posee un módulo concreto de cinco tipos predefinidos.
- 30. REPOSITORIO.** Componente de un sistema CASE en el que se controla el almacenamiento y acceso de la información relativa a un sistema de software en desarrollo.
- 31. REQUISITO FUNCIONAL.** Relación precisa entre entradas y salidas que debe satisfacer un sistema de software.
- 32. REQUISITO NO FUNCIONAL.** Atributo de calidad que debe satisfacer un sistema de software.
- 33. REQUISITOS.** Funciones o limitaciones que debe satisfacer un sistema de software. Se denominan requisitos de usuario cuando son definidos por éstos y de sistemas cuando surgen de la funcionalidad que debe tener un sistema para cumplir con los requisitos de usuario.
- 34. SA/RT («Structured Analysis for Real Time»).** Análisis Estructurado para Tiempo Real. Extensión de las técnicas de desarrollo estructurado.
- 35. SDL («Specification and Design Language»).** Norma internacional promovida por el UIT-T (Unión Internacional de Telecomunicaciones) para la descripción de sistemas de comunicación.
- 36. SISTEMA DE SOFTWARE.** Sistema en el que la funcionalidad ofrecida al usuario se consigue mediante el desarrollo de uno o varios programas ejecutables.
- 37. SISTEMA DE TIEMPO REAL (STR).** Es aquél sistema de software que debe completar sus actividades en plazos de tiempo predeterminados. Como consecuencia,
-

su ejecución debe satisfacer restricciones temporales cuyo incumplimiento supone el funcionamiento incorrecto del sistema.

38. TECNOLOGÍA DE SOFTWARE. Conjunto de elementos que puede utilizar un desarrollador de un sistema de software durante las diferentes fases del modelo de ciclo de vida elegido. Está constituida por un conjunto de notaciones, formas de razonar, herramientas, método de desarrollo y directrices de aplicación industrial.

39. TRANSFERENCIA DE TECNOLOGÍA. Conjunto de procedimientos necesarios para que una organización adopte una tecnología desde un proveedor de la misma.

40. VALIDACIÓN. Procedimientos de gestión requeridos para asegurar que un sistema de software satisface los requisitos de calidad impuestos.

41. VERIFICACIÓN. Procedimiento matemático para asegurar la corrección de un algoritmo

BIBLIOGRAFÍA:

- [1] Ludwing Von Bertalanfy, *Teoría General de los sistemas*, Fondo de Cultura Económica, Duodécima reimpresión 2000
- [2] Gran Enciclopedia Ecisa; Matemáticas, Informática; Ediciones Culturales Internacionales. Pág. 205; Editorial Norma, S.A., 2001
- [3] Roberto Rodríguez González; Tesis de Licenciatura de la computación.
- [4] Yourdon E. ; *Análisis estructurado Moderno*; Prentice Hall; 1993

- [5] Gonzalo León Serrano; *Ingeniería de Sistemas de Software*; Edit. Isdefe; 1996
- [6] Lewis G. 1994. "What is Software Engineering?" *DataPro* (4015). Feb 1994. pp. 1-10
- [7] Cota A. 1994 "Ingeniería de Software". *Soluciones Avanzadas*. Julio de 1994. pp. 5-13.
- [8] Martin Fowler, Kendall Scout; *UML gota agota*, Addison Wesley, 1999
- [9] Jacobson, I. 1998. "Applying UML in The Unified Process" Presentación. Rational Software. Presentación disponible en <http://www.rational.com/uml> como UMLconf.zip
- [10] Ibargüengoitia G., Oktaba H. *Ingeniería de software para principiantes*. Facultad de Ciencias UNAM 2005.

- [11] Leif Oswaldo Alducin Gil; "Aplicación de TSPi para un sitio WEB en CONACULTA"; Tesis de Licenciatura en Ciencias de la Computación; 2004.

- [12] Alejandro Espinoza Vega; "Administración de sistemas de información con enfoque en análisis de requerimientos"; Tesis de Actuario; 2002

- [13] Sergio Ramos Chávez; "Desarrollo de un sitio WEB siguiendo la metodología TSPi."; Tesis de Licenciatura en Ciencias de la Computación; 2002

- [14] Barros V. Oscar; *Desarrollo orientado a objetos, Sistemas de Información para la Reingeniería*; Editorial Universitaria, Colección Nuestra Técnica; 1996
- [15] Grady Booch; *Análisis y diseño orientado a objetos con aplicaciones*; Segunda edición; Addison- Wesley / Diaz de Santos; 1996
- [16] Gerez Víctor, Mier Mauricio; *Desarrollo y Administración de Programas de Computadoras (Software)*; Instituto de Investigaciones Eléctricas; CECSA; 1984.
- [17] James Martín; James J. Odell; *Análisis y Diseño Orientado a Objetos*; Prentice Hall; 1992
- [18] Oscar Johansen Bertoglio; *Introducción a la Teoría General de Sistemas*; Pág. 53; Editorial LIMUSA; 2001

estimaciones consideran: las vaquillas nacidas en la explotación, las disponibles al inicio de la proyección para el primer año y las compradas.

- Vaquillas acumuladas. Son los animales que ingresan el año de cómputo a la etapa de desarrollo 2, y alcanzan los 24 meses de edad ese mismo año o el siguiente; además de los que entraron en la etapa el año anterior y todavía permanecen.
- Vaquillas de 24 meses en etapa de gestación. Son todas las vaquillas que alcanzan los 24 meses de edad durante el año de cómputo, por las distintas vías de incorporación a la recría, y consiguen su primer parto en el curso de los próximos 12 meses —entre los 24 y 36 meses de edad—, según los supuestos básicos fijados.
- Vaquillas de reemplazo en etapa de gestación a parto. Son todas las vaquillas que entran al hato productor el año de cómputo.

IV. Engorda de machos

En cuanto a la engorda de machos (Figura 20), el apartado que nos ocupa, el Simulador despliega información sobre los terneros que se conservan en la explotación, principalmente con el propósito de comercializar su carne. Estos animales pueden resultar de progenitores puros o ser híbridos de diferentes razas lecheras o de la combinación de razas lecheras y de carne, lo cual tendrá efecto sobre uno o más aspectos de interés para el productor, como por ejemplo: ganancias de peso, calidad de la carne, rendimiento de la canal, precio del producto y edad a la venta.

En el primer renglón del Figura 7.20 se reporta información sobre la cantidad de terneros vivos de que se puede disponer anualmente para crianza de becerros; y en el siguiente renglón, se indica la cantidad de terneros que se destinan a la engorda de añejos. En el primer caso, se han computado las bajas por muerte y desecho en el transcurso de la lactancia -del nacimiento al destete-; y en el segundo, durante el período que va del destete a la venta. La edad al destete no afecta el número de terneros vivos disponibles anualmente para rastro o engorda, pero sí la disponibilidad de animales destetados para incorporarse a la crianza de novillos.

IV.- Engorda de machos									
Describe:	Primero:	Segundo:	Tercero:	Cuarto:	Quinto:	Sexto:	Séptimo:	Total:	Promed
Terneros-Ms. y Ds	44	74	90	99	105	108	110	630.00	89
Añojos[engorda]	0	4	7	9	10	11	11	52.00	7
Total	44	78	97	108	115	118	121	682.00	97

Figura 7. 20 Reporte Engorda de Machos

V. Producción de leche

En la Figura 7.21 se muestra que el Simulador distingue la producción de leche correspondiente a vacas de primer parto en el primer renglón, vacas de dos o más partos segundo renglón y total de vacas ultimo renglón.

Los factores que afectan la producción y composición de la leche son muchos, muy variados y, algunos de ellos, por demás complejos.

V.- Producción de leche									
Describe	Primero:	Segundo:	Tercero:	Cuarto:	Quinto:	Sexto:	Séptimo:	Total:	Promedio:
Vacas de 1er parto/año	349,019	177,492	162,826	185,558	209,253	218,054	220,191	522,392	217,484.64
Vacas de 2 o más ps/año	446,658	967,516	1,202,007	1,318,465	1,368,231	1,403,348	1,441,913	148,137	164,019.63
Total	795,676	1,145,008	1,364,833	1,504,023	1,577,484	1,621,401	1,662,104	670,530	381,504.00

Figura 7. 21 Reporte Producción de Leche

VI. Producción de carne

El Simulador calcula y expone datos sobre la producción de carne, en la Figura 7.22 se muestra información en lo referente a dos conceptos:

(a) La eliminación de ganado para rastro: terneras, becerras, vaquillas, vaquillas de reemplazo, vacas.

(b) Engorda de machos: terneros, añojos.

En cuanto a la eliminación de ganado para rastro, conviene recordar que los datos se verán afectados, entre otros factores, por las tasas de mortalidad y desecho fijadas para las distintas categorías de hembras, y los porcentajes de recuperación correspondientes. Asimismo, por lo que toca a la engorda de machos, los montos fluctuarán en gran medida, dependiendo de la proporción de terneros conservados para la engorda, el porcentaje de terneros eliminados por muerte y desecho, la tasa de recuperación para rastro del total de animales eliminados y el índice de mortalidad y desecho de los añojos.

VI- Producción de carne									
Describe	Primero	Segundo	Tercero	Cuarto	Quinto	Sexto	Séptimo	Total	Promedio
Terneras	4	2	2	2	3	2	2	18.00	2.58
Becerras	2	1	1	1	1	1	1	6.00	0.82
Vaquillas	11	5	5	5	5	5	5	41.00	5.81
Vaquillas de reemplazo	1	2	7	6	4	4	4	27.00	3.84
Vacas	10	24	34	45	52	53	54	272.00	38.90

Figura 7. 22 Reporte Producción de Carne

VII. Excedentes en la producción de vaquillas de reemplazo

Con mayor o menor frecuencia, e incluso de manera eventual, las bajas en un hato son cosa segura. Como se mencionó en el punto anterior, de una cuarta parte a dos

quintas partes de las vacas de un hato deben sustituirse con animales de la recria, o en su defecto, con vaquillas o vacas adquiridas en otras explotaciones. De no ser así, el rebaño en pocos años desaparecería. De ahí que la primera función de la recria sea la repoblación del hato productor con animales de calidad igual o superior a la de los relevados. Hecho esto, los excedentes de la producción de reemplazos pueden comercializarse, generando ingresos adicionales para la empresa. Para satisfacer la primer función, y más aún para desarrollar la segunda, no basta con tener una recria muy eficiente, en que se mantengan bajos índices de mortalidad y desecho, se consigan las metas de las distintas etapas de desarrollo y se optimicen los costos de operación; además, es primordial que las vacas tengan una prolongada vida productiva, de modo que se reduzcan las necesidades de reemplazos, por una parte, y aumente el número de crías por vaca, por la otra. En este plan se consigue la convergencia de un doble círculo virtuoso, donde la vida que posee y genera el hato productor, numérica y temporalmente supera el advenimiento inexorable de las bajas.

En esencia, los factores que contribuyen a elevar la disponibilidad de ganado de reemplazo, para atender las necesidades de repoblación y comercialización de una empresa lechera, son los siguientes:

- Animales bien alimentados y sanos.
 - Condiciones ambientales y de manejo confortables.
 - Parámetros reproductivos y productivos adecuados.
 - Minimización de causas de muerte y desecho.
 - Un sistema de registro y control efectivo.
 - Un enfoque de producción armónico (integración de aspectos ambientales, biológicos, etológicos, sociales, económicos y tecnológicos).
-

VII- Excedentes en la Producción de vaquillas de reemplazo									
Describe	Primero	Segundo	Tercero	Cuarto	Quinto	Sexto	Séptimo	Total	Promedio
Para venta	21	26	144	110	52	47	51	451.00	64.39

Figura 7. 23 Reporte Excedente en la Producción de Vaquillas de Reemplazo

Por lo que toca al Simulador y dadas sus limitaciones, los factores arriba citados han de considerarse de manera indirecta. Siendo las explotaciones lecheras sistemas complejos, como sistemas de sistemas de acciones —y/o eventos—, prácticamente nada que forme parte de su realidad o la de su entorno, está disociado de ellas. De una forma u otra, los diferentes factores han de repercutir en todos y cada uno de los supuestos básicos con que opera el Simulador; por lo cual, no deben verse de manera aislada. Todos los factores referidos, y por consiguiente los supuestos básicos del Simulador, han de reflejarse, directa o indirectamente, en el tamaño del hato, la capacidad disponible y el excedente de reemplazos para venta.

VIII. Ingresos

A partir de los precios de venta proporcionados al Simulador, éste calcula y reporta los ingreso anuales de la operación de la explotación (ingresos brutos), con respecto a los siguientes conceptos de ventas: leche, terneras de desecho, becerras de desecho, vaquillas de desecho, vacas y vaquillas de reemplazo de desecho, terneros de desecho o engorda, añojos y vaquillas de reemplazo para producción, como se muestra en la Figura 7.24, donde aparece el ingreso total anual.

VIII- Ingresos									
Describe:	Primero:	Segundo:	Tercero:	Cuarto:	Quinto:	Sexto:	Séptimo:	Total:	Prom
Leche	2387029	3,435,023	4,094,499	4,512,070	4,732,453	864,203	4,986,311	011,590	4.1
Terneras	2236	1,007	943	1,220	1,323	1,120	1,190	9,041	
Becerras	1111	522	489	458	515	467	475	4,037	
Vaquillas	8954	4,208	3,940	3,688	4,149	3,759	3,831	32,529	
Vacas y Vqs R.)	50220	121,468	170,981	223,721	258,146	264,844	272,040	361,420	1
Terneros	22058	37,279	45,403	50,300	53,247	54,372	55,620	318,280	
Añojos	0	12,304	33,098	46,120	53,384	57,759	60,030	262,695	
Venta de reemplazos	207227	256,918	1,444,533	1,097,962	516,480	469,503	514,959	507,581	6
Total	2678836	3,868,730	5,793,886	5,935,538	5,619,696	716,028	5,894,458	507,173	5.0

Figura 7. 24 Reporte Ingresos.

IX. Costos globales

Tras activar el rubro de costos globales del menú de salidas, aparecen los costos anuales de producción, Figura 7.25. Los costos globales resultan de la suma de los costos de producción considerados.

IX.- Costos Globales									
Describe:	Primero:	Segundo:	Tercero:	Cuarto:	Quinto:	Sexto:	Séptimo:	Total:	Prom
Total de costos	1010000	1010002	1010001	1010000	1010000	010000	1010000	0003.00	1010000

Figura 7. 25 Reporte Costos globales.

X. Costos de "X" por litro

En la Figura 7.26 se expone la información referente al costo de producción de un litro de leche: alimentación, salarios, compra de ganado, medicinas y vacunas, Otros costos, Intereses de capital, amortizaciones, total

X- Costo del litro de leche por concepto de 'X'									
Describe:	Primero:	Segundo:	Tercero:	Cuarto:	Quinto:	Sexto:	Séptimo:	Total:	Promedio:
Alimentación	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Salarios	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.15
Compra de ganado	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.08
Medicinas y vacunas	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.05
Otros costos	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.04
Interes de capital	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02
Amortizaciones	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.06
Total	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.78

Figura 7. 26 Reporte Costo del Litro de Leche por Concepto "X"

Los conceptos de costo, como se explicó en su oportunidad, pueden ser sustituidos o modificados de acuerdo a las necesidades del usuario. Cada concepto seleccionado requiere de la alimentación adecuada del Simulador desde un principio, considerando los Apartados involucrados (A-F, J-L, O y P).

XI. Utilidades

Las utilidades constituyen el excedente de los ingresos sobre los costos. Cuando ocurre al revés, los costos superan a los ingresos, hablamos de pérdidas. Como ingresos netos de operación, las utilidades resultan de la diferencia entre los ingresos totales anuales (ingresos brutos de operación) y los costos de operación anuales. De tal manera, a los conceptos de ingreso —las entradas— se restan los conceptos de gasto. En la Figura 7.27 se muestran las utilidades calculadas por el Simulador, en el marco de los datos aportados por el usuario y de las consideraciones propias.

XI.- Utilidades									
Describe:	Primero:	Segundo:	Tercero:	Cuarto:	Quinto:	Sexto:	Séptimo:	Total:	Promedio:
Utilidades	1668836	2858728	4783885	4925538	4609696	1706028	4884458	7170.00	4062452.8

cancelar Imprimir

Figura 7. 27 Reporte Utilidades.

XII. Razón beneficio: Costo

Por definición, la razón es la relación entre el número de elementos de un grupo (a) y el número de elementos de otro (b); así, la razón beneficio-costos se obtiene dividiendo el monto de los beneficios entre el monto de los costos.

El método de análisis beneficio-costos permite estimar las tasas de retorno de un proyecto y observar la eficiencia de su rentabilidad con respecto a la de otros proyectos; o bien, comparar la eficiencia de una situación dada (sin proyecto) frente a una de cambio (con proyecto). El análisis beneficio-costos proporciona información muy valiosa sobre la viabilidad de un proyecto, con respecto a sus propias condiciones y circunstancias, como también a otros proyectos.

Una razón beneficio-costos de 1 significa que los beneficios del proyecto equivalen a sus costos, considerando el costo de oportunidad del capital. Si tenemos que comparar varios proyectos lecheros, en planeación u operación, podemos jerarquizarlos por su eficiencia económica, mediante análisis de beneficio-costos. El de la mayor razón será el más eficiente, y el de la menor, el menos. Asimismo, en una situación de recursos limitados, en que, por ejemplo, un ganadero tiene que decidir entre invertir en una ampliación de la sala de ordeño o invertir en mejoras a las instalaciones de los parideros, la comparación de la eficiencia rentable de ambos proyectos puede ser muy útil. Frente a estos problemas, la jerarquía de proyectos alternativos apunta a la maximización del

rendimiento de cada peso invertido.

Aunado a sus notables ventajas, el análisis beneficio-costos también presenta limitaciones:

Primero, la distribución de los beneficios y los costos no se refleja en la razón de unos a otros. Por ejemplo, dos proyectos lecheros pueden tener un rendimiento económico similar, pero en uno son pocas las personas beneficiadas, y en otro, muchas. Dado el carácter del estimador —cuantitativo, unidimensional, de proporción simple—, se puede caer en un esquema reduccionista de apreciación y decisiones, donde únicamente los aspectos económicos determinen el uso eficiente de los recursos.

Segundo, cuando se trata de proyectos mutuamente excluyentes, la razón beneficio-costos puede engañarnos. Un proyecto puede tener una elevada razón beneficio-costos comparado con otro, pero su margen utilidades ser mucho más bajo; por ello, en la selección de proyectos mutuamente excluyentes, cuando se busca la maximización de utilidades, suele preferirse el criterio del valor presente neto al del beneficio-costos.

Tercero, en el afán de apreciar a través de una escala numérica los diferentes factores en pro (beneficios) y en contra (costos) de un proyecto, a las variables cualitativas se les asignan valores cuantitativos que irremediablemente las distorsionan, sesgan o, por lo menos, desnaturalizan.

Cuarto, comúnmente en las explotaciones lecheras una iniciativa de cambio (proyecto) tiene que competir con otras, en tiempo y por recursos, a fin de probar mayores niveles de eficiencia, productividad o utilidades; de tal suerte, los proyectos que incorporan aspectos de desarrollo social —como mejores ingresos para los trabajadores, apoyo a la demanda de vivienda, o programas de capacitación y educación— se encuentran en franca desventaja.

La Figura 7.28 reporta la razón beneficio-costos calculada por el Simulador a partir de las instrucciones y parámetros manejados por el usuario en los supuestos básicos. Si bien el cómputo de la razón beneficio-costos supone una operación muy simple, prácticamente todos los conceptos involucrados en el Simulador repercuten en la magnitud y sentido (positivo o negativo) de los datos. A manera de balanza, el estimador sopesa los factores de ingreso, tenidos por beneficios, y los factores de egreso, reconocidos como costos, con lo que se integra en una sola observación la participación

compleja de unos y otros; pero además, esta información se proyecta en períodos regulares de tiempo, de modo que es posible analizar la interrelación beneficio-costos en un año particular, como a través de varios años.

Razon beneficio: Costo									
XII. Razón beneficio: costo									
Describe:	Primero:	Segundo:	Tercero:	Cuarto:	Quinto:	Sexto:	Séptimo:	Total:	Promedio:
Beneficio/costos	2	3	5	5	5	5	5	0.00	5.02

Figura 7. 28 Reporte Razón Beneficio: Costo

7.2 Evaluación del simulador

El plan de pruebas que se siguió para el Simulador de Hatos Lecheros se aplicó en el Rancho San José ubicado en el Kilómetro 32.5 de la carretera México-Texcoco, Santiago Cuatlalpan, Edo. de México. Este rancho cuenta con vacas Holstein, se dedica a la producción de leche, con una industria heterogénea de ordeña estacional, ordinaria, que se realiza de manera mecánica, con dos ordeñas diarias, lo que permite producir 12000 l/día de leche. Cuenta con un número total de vacas de 532, con una distribución que se muestra en la tabla 7.1.

Funcionamiento general del Simulador.

La simulación que se realizó consistió en dar de alta una serie de supuestos básicos, con el fin de evaluar la rentabilidad de la empresa, ante un incremento del número de cabezas, con costos de producción e ingresos por la venta de leche.

Este plan de pruebas nos permitirá conocer el comportamiento del Simulador en situaciones reales, así como podremos dar cuenta de los problemas en su funcionamiento. Se evaluarán los siguientes aspectos:

- Validación de la información.
- Comportamiento en los cálculos y transacciones.
- Calidad y tiempo.

No. de vacas (total)	532	No. de vaquillas	403
No. de vacas para ordeña	462	No. de becerras vírgenes	218
No. de vacas paridas	122	No. de becerras inseminadas	55
No. de vacas inseminadas	156	No. de becerras abiertas	6
No. de vacas abiertas	45	No. de becerras preñadas	124
No. de vacas preñadas	139	No. de becerras de 1-3 meses	60
No. de vacas secas	70	No. de becerras de 4-6 meses	35
No. de vacas en lactancia 1	194	No. de becerras de 7-9 meses	34
No. de vacas en lactancia 2	158	No. de becerras de 10-12 meses	49
No. de vacas en lactancia 3	76	No. de becerras de 13-15 meses	39
No. de vacas en lactancia 4	55	No. de becerras de 16-18 meses	58
No. de vacas en lactancia 5	25	No. de becerras de 19-21 meses	60
No. de vacas en lactancia 6	8	No. de becerras de 22-24 meses	40
No. de vacas en lactancia 7	12	No. de becerras de 25-27 meses	13
No. de vacas en lactancia 8	3	No. de becerras > 27 meses	15
No. de vacas en lactancia 9	1	No. de machos	67

Tabla 7.1. Datos de Número total de cabezas que integran el rancho San José

Los supuestos básicos incluyen una población inicial de ganado productor de primer parto igual a 301, dando un promedio anual de 43. La población inicial de ganado productor de vacas de 2 o más partos es de 3,423 (489 el promedio anual).

La estructura del hato se muestra en la tabla 7.2. En el apéndice A se muestra el análisis realizado por el simulador, así como también los cálculos que se efectúan en el análisis.

Concepto	Total	Promedio anual
Terneras	2295	1147
Becerras	2987	1494
Vaquillas	3723	1862
Vacas de primer parto	625	312
Vacas de 2 o más partos	2099	1550
Vacas de 1 o más partos	3724	1862
No. final de vacas de 1 o más partos	3724	532

Tabla 7.2. Estructura del hato

En base en los puntos anteriores de evaluación del simulador, se detectaron los siguientes problemas:

La simulación que se realizó para evaluar la productividad en un lapso de 7 años, tomando en cuenta un costo del litro de leche de \$3.50, arrojó un ingreso total de \$23,843,339.00, con costos de producción que alcanzan un total de \$53,024,658.00, dejando utilidades negativas por -\$29,181,319.00. Estos resultados revelan que el aumento del número de cabezas de 562 a 3,724, en 7 años, no es factible, ya que los costos totales son mayores que las utilidades, teniendo un déficit muy grande en las ganancias. El precio de la leche, así como los costos de producción no permiten que haya un crecimiento como el calculado dentro del hato.

Con esto, es posible concluir que al manipular los datos iniciales, se puede establecer el tamaño del hato, para hacerlo productivo y rentable.

Conclusiones

La evaluación de la producción de leche es complicada dado el número grande de variables que se involucra. Es por ello que el simulador que se presenta en este trabajo modela las condiciones de este sistema complejo, con el objeto de dar información a los productores que les permita hacer más rentable el proceso.

El uso de la Ingeniería de Software hizo posible optimizar el desarrollo del sistema permitiendo organizar y manejar los datos de una manera comprensible, tanto para el programador como para el usuario. En este aspecto, las herramientas empleadas permitieron disminuir el tiempo de diseño, desarrollo, instrumentación y la fase de pruebas así como la de detectar posibles fallas en el funcionamiento del sistema. Así mismo, es importante señalar que gracias a esta metodología, el mantenimiento y modificaciones que requiere el sistema, se realiza de una manera fácil.

El simulador hace una proyección de la productividad de un hato en un lapso de 7 años, tiempo considerado necesario para el reemplazo del ganado. El sistema se ha desarrollado con una tecnología de software orientado a objetos, manejando archivos dbf, el diseño se ha hecho siguiendo los modelos de UML, que muestra diferentes diagramas, y hace más comprensible la tarea de programar lo que el usuario solicita.

Por medio del simulador, se ha concluido que los datos presentados sean procesables y partan de criterios y condiciones debidamente fundamentados, ya que se cuenta con información suficiente, precisa y confiable sobre el comportamiento de la explotación en un período de varios años; además, la información proviene de explotaciones lecheras operando en situaciones específicas, similares a las del caso en cuestión.

El proyecto se ha llevado a cabo, no sólo con datos reales, sino hipotéticos, derivados de información teórica, sólida (científica), ajustada con sentido práctico, según el buen saber y entender del usuario y sus fuentes de información.

El simulador ha sido probado exitosamente en un rancho productor de leche que se encuentra en el Edo. de México, haciendo una proyección a 7 años. Si bien arrojó resultados negativos, ya que se encontró que los ingresos eran mucho menores que los egresos al aumentar el tamaño del hato, la información obtenida es muy útil para el productor, pues le permite planear el crecimiento de su empresa.

Este tipo de aplicaciones muestra que es factible realizar modelación práctica que realmente sea funcional.

III Apéndices

Apéndice A

Resultados del Simulador de Hatos Lecheros a 7 años para el rancho.

Movimiento de Ganado

Cap. Hato	500
I.P. (ms)	16
Días Abiertos	

1	=SI(F1C19<12,1,042,SI(F1C19<12,1,036,SI(F1C19<13,1,036,SI(F1C19<14,0,968,1))))
---	--

Compra de Ganado

Compra de Ganado				
Año	Ternereras	Becerras	Vaquillas	Vqs. Reem.
1	100	100	10	20
2	30	10	5	15
3	5	30	30	10
4	25	4	20	5
5	30	5	10	4
6	5	6	8	3
7	10	7	9	2
Hato Año1	100	5	4	10

Producción del año anterior	No. de Vaqs. (cbz./año)	No. de Vacas (cbz./año)
830000	80	120
Produc/Año	$= (1 - F28C19/100) * (F26C18) / ((1 - F28C19/100) * F26C19 + F26C20)$	$= (F26C18) / ((1 - F28C19/100) * (F26C19) + (F26C20))$
Diferencia %	25	

Costos de producción

Costos de Producción (\$)								
Conc./Año	1	2	3	4	5	6	7	Total
Alimentación	500000	500000	500000	500000	500000	500000	500000	=SUMA(FC13:FC19)
Salarios	200000	200000	200000	200000	200000	200000	200000	=SUMA(FC13:FC19)
Compra de ganado	100000	100000	100000	100000	100000	100000	100000	=SUMA(FC13:FC19)
Medicinas y vacs.	60000	60000	60000	60000	60000	60000	60000	=SUMA(FC13:FC19)
Otros costos	50000	50000	50000	50000	50000	50000	50000	=SUMA(FC13:FC19)
Gastos financieros	20000	20000	20000	20000	20000	20000	20000	=SUMA(FC13:FC19)
Depreciación	80000	80000	80000	80000	80000	80000	80000	=SUMA(FC13:FC19)
Total	=SUMA(F35C:F41C)	=SUMA(F35C:F41C)	=SUMA(F35C:F41C)	=SUMA(F35C:F41C)	=SUMA(F35C:F41C)	=SUMA(F35C:F41C)	=SUMA(F35C:F41C)	=SUMA(FC13:FC19)

Eliminación de Ganado para Rastro

Rastro: Ms+Ds	
Terneras	10
Becerras	10
Vaquillas	90
Vaqs. de Reem.	90
Vacas	90
Terneros	100

Ingresos

Precio de Venta (N\$)	
Leche	3
Terneras	500
Becerras	700
Vaquillas	800
Vacas y Vqs. R.	5000
Terneros	500
Añojos	6000
Reemplazos	10000

Recría		
Terneritas (0-3 ms)	0	2.5
Becerras (2 ms-Vq.)	=F16C17+0.1	=SUMA(F18C16-0.1)
Vaqs (12-vq. reemp.)	12	=SUMA(F19C16-0.1)
Vqs. R. (24-36 ms)	24	30
Añojos (%terneritas)	10	=(FC16)/100
Añojos (12-24 ms)	24	24

Producción de leche

Producción de leche lts.	
Vs 1er parto/Año	4400
Vs 2 o más ps/Año	6200

Muertes y desechos

Muertes y Desechos	
Vacas de 2 o + ps	%
año 1	12
año 2	15
año 3	17
año 4	20
año 5	22
año 6	22
año 7	22
Terneritas	18
Becerras	6
Vaquillas	5
Vqs. R.	2
Terneritos	10
Añojos	6

Vacas 1er parto

2

Tamaño del hato

Capacidad disponible

Vacas 1er p.	Vacas 2 o + ps.	Máximo	Mínimo	Medio	Máxima	Minima	Media
8	50	=F16C3	=F32C3	=(F16C3+F32C3)/2	=F3C19-FC21	=F3C19-FC20	=F3C19-FC22
7	40	=F16C4	=F32C4	=(F16C4+F32C4)/2	=F3C19-FC21	=F3C19-FC20	=F3C19-FC22
6	30	=F16C5	=F32C5	=(F16C5+F32C5)/2	=F3C19-FC21	=F3C19-FC20	=F3C19-FC22
5	20	=F16C6	=F32C6	=(F16C6+F32C6)/2	=F3C19-FC21	=F3C19-FC20	=F3C19-FC22
4	10	=F16C7	=F32C7	=(F16C7+F32C7)/2	=F3C19-FC21	=F3C19-FC20	=F3C19-FC22
3	5	=F16C8	=F32C8	=(F16C8+F32C8)/2	=F3C19-FC21	=F3C19-FC20	=F3C19-FC22
2	5	=F16C9	=F32C9	=(F16C9+F32C9)/2	=F3C19-FC21	=F3C19-FC20	=F3C19-FC22
70	30	=SUMA(FC18:FC19)	=SUMA(FC18:FC19)	=SUMA(FC18:FC19)	=F3C19-FC21	=F3C19-FC20	=F3C19-FC22

Cálculos para la Proyección del Simulador de Hatos Lecheros a 7 años

CONC./AÑO		1	2	3	4	5	6	7	TOTAL
1. Movimiento de ganado	Terneras	=(N\$5)	=(N\$6)	=(N\$7)	=(N\$8)	=(N\$9)	=(N\$10)	=(N\$11)	=SUMA(\$C2:\$I2)
	Becerras	=(O\$5)	=(O\$6)	=(O\$7)	=(O\$8)	=(O\$9)	=(O\$10)	=(O\$11)	=SUMA(\$C3:\$I3)
	Vaquillas	=(P\$5)	=(P\$6)	=(P\$7)	=(P\$8)	=(P\$9)	=(P\$10)	=(P\$11)	=SUMA(\$C4:\$I4)
	Vaquillas de reemplazo	=(Q\$5)	=(Q\$6)	=(Q\$7)	=(Q\$8)	=(Q\$9)	=(Q\$10)	=(Q\$11)	=SUMA(\$C5:\$I5)
	Vacas de 1er parto	=(R\$5)	=(R\$6)	=(R\$7)	=(R\$8)	=(R\$9)	=(R\$10)	=(R\$11)	=SUMA(\$C6:\$I6)
	Vacas de 2 o más partos	=(S\$5)	=(S\$6)	=(S\$7)	=(S\$8)	=(S\$9)	=(S\$10)	=(S\$11)	=SUMA(\$C7:\$I7)
1.2. Ganado de recría no comprado	Terneras	=C\$38-C\$2*(1-\$L\$12/100)	=D\$38-D\$2*(1-\$L\$12/100)	=E\$38-E\$2*(1-\$L\$12/100)	=F\$38-F\$2*(1-\$L\$12/100)	=G\$38-G\$2*(1-\$L\$12/100)	=H\$38-H\$2*(1-\$L\$12/100)	=I\$38-I\$2*(1-\$L\$12/100)	=SUMA(\$C9:\$I9)
	Becerras	=C\$40-C\$3*(1-\$L\$13/100)	=D\$40-D\$3*(1-\$L\$13/100)	=E\$40-E\$3*(1-\$L\$13/100)	=F\$40-F\$3*(1-\$L\$13/100)	=G\$40-G\$3*(1-\$L\$13/100)	=H\$40-H\$3*(1-\$L\$13/100)	=I\$40-I\$3*(1-\$L\$13/100)	=SUMA(\$C10:\$I10)
	Vaquillas	=C\$42-C\$4*(1-\$L\$14/100)	=D\$42-D\$4*(1-\$L\$14/100)	=E\$42-E\$4*(1-\$L\$14/100)	=F\$42-F\$4*(1-\$L\$14/100)	=G\$42-G\$4*(1-\$L\$14/100)	=H\$42-H\$4*(1-\$L\$14/100)	=I\$42-I\$4*(1-\$L\$14/100)	=SUMA(\$C11:\$I11)
	Vaquillas de reemp.	=C\$45-C\$5*(1-\$L\$15/100)	=D\$45-D\$5*(1-\$L\$15/100)	=E\$45-E\$5*(1-\$L\$15/100)	=F\$45-F\$5*(1-\$L\$15/100)	=G\$45-G\$5*(1-\$L\$15/100)	=H\$45-H\$5*(1-\$L\$15/100)	=I\$45-I\$5*(1-\$L\$15/100)	=SUMA(\$C12:\$I12)
1.3. Población inicial del hato productor	Vacas de 1er parto	=R\$12+C\$6+\$24:\$24	=D\$6+\$24:\$24+C\$30*(1-12/\$S\$1)	=E\$6+\$24:\$24+D\$30*(1-12/\$S\$1)	=F\$6+\$24:\$24+E\$30*(1-12/\$S\$1)	=G\$6+\$24:\$24+F\$30*(1-12/\$S\$1)	=H\$6+\$24:\$24+G\$30*(1-12/\$S\$1)	=I\$6+\$24:\$24+H\$30*(1-12/\$S\$1)	=SUMA(\$C14:\$I14)
	Vacas de 2 o más partos	=S\$12+\$7:\$7	=C\$31+\$7:\$7+(C\$30)*(1-12/\$S\$1)	=D\$31+\$7:\$7+(D\$30)*(1-12/\$S\$1)	=E\$31+\$7:\$7+(E\$30)*(1-12/\$S\$1)	=F\$31+\$7:\$7+(F\$30)*(1-12/\$S\$1)	=G\$31+\$7:\$7+(G\$30)*(1-12/\$S\$1)	=H\$31+\$7:\$7+(H\$30)*(1-12/\$S\$1)	=SUMA(\$C15:\$I15)
	Total	=\$14:\$14+\$15:\$15	=\$14:\$14+\$15:\$15	=\$14:\$14+\$15:\$15	=\$14:\$14+\$15:\$15	=\$14:\$14+\$15:\$15	=\$14:\$14+\$15:\$15	=\$14:\$14+\$15:\$15	=SUMA(\$C16:\$I16)

CONC./AÑO		1	2	3	4	5	6	7	TOTAL
1.4. Estructura del hato	Teneras acumuladas	=C\$39	=D\$39	=E\$39	=F\$39	=G\$39	=H\$39	=I\$39	=SUMA(\$C18:\$I18)
	Becerras acumuladas	=C\$41	=D\$41	=E\$41	=F\$41	=G\$41	=H\$41	=I\$41	=SUMA(\$C19:\$I19)
	Vaquillas acumuladas	=C\$43	=D\$43	=E\$43	=F\$43	=G\$43	=H\$43	=I\$43	=SUMA(\$C20:\$I20)
	Vacas de 1er parto (m.a.)	=(C\$14+\$30:\$30)/2	=(D\$14+\$30:\$30)/2	=(E\$14+\$30:\$30)/2	=(F\$14+\$30:\$30)/2	=(G\$14+\$30:\$30)/2	=(H\$14+\$30:\$30)/2	=(I\$14+\$30:\$30)/2	=SUMA(\$C21:\$I21)
	Vacas de 2 o más ps. (m.a.)	=(C\$15+\$31:\$31)/2	=(D\$15+\$31:\$31)/2	=(E\$15+\$31:\$31)/2	=(F\$15+\$31:\$31)/2	=(G\$15+\$31:\$31)/2	=(H\$15+\$31:\$31)/2	=(I\$15+\$31:\$31)/2	=SUMA(\$C22:\$I22)
	TOTAL de vacas (m.a.)	=21:\$21+\$22:\$22	=21:\$21+\$22:\$22	=21:\$21+\$22:\$22	=21:\$21+\$22:\$22	=21:\$21+\$22:\$22	=21:\$21+\$22:\$22	=21:\$21+\$22:\$22	=SUMA(\$C23:\$I23)
	Ajuste (vqs-vacas 1er p.-PI)	=SI(\$R\$12+C\$6+C\$15>=\$S\$3,0,SI(\$S\$3-\$R\$12	=SI(D\$6+C\$30*(1-12/\$S\$1)+D	=SI(E\$6+D\$30*(1-12/\$S\$1)+E\$15	=SI(F\$6+E\$30*(1-12/\$S\$1)+F	=SI(G\$6+F\$30*(1-12/\$S\$1)+G\$1	=SI(H\$6+G\$30*(1-12/\$S\$1)+H	=SI(I\$6+H\$30*(1-12/\$S\$1)+I	=SUMA(\$C24:\$I24)
Ajuste (vqs-vacas 1er p.-PF)	=SI(\$45:\$45-\$24:\$24<=\$28:\$28,\$45:\$45-\$24:\$24,\$28:\$2	=SI(\$45:\$45-\$24:\$24<=\$28:\$28,\$45:\$45-\$24:\$24,\$28:\$2	=SI(\$45:\$45-\$24:\$24<=\$28:\$28,\$45:\$45-\$24:\$24,\$28:\$2	=SI(\$45:\$45-\$24:\$24<=\$28:\$28,\$45:\$45-\$24:\$24,\$28:\$2	=SI(\$45:\$45-\$24:\$24<=\$28:\$28,\$45:\$45-\$24:\$24,\$28:\$2	=SI(\$45:\$45-\$24:\$24<=\$28:\$28,\$45:\$45-\$24:\$24,\$28:\$2	=SI(\$45:\$45-\$24:\$24<=\$28:\$28,\$45:\$45-\$24:\$24,\$28:\$2	=SUMA(\$C25:\$I25)	
1.5. Muertes y desechos del hato productor	Vacas de 1er parto	=14:\$14*\$L\$18/100	=14:\$14*\$L\$18/100	=14:\$14*\$L\$18/100	=14:\$14*\$L\$18/100	=14:\$14*\$L\$18/100	=14:\$14*\$L\$18/100	=14:\$14*\$L\$18/100	=SUMA(\$C26:\$I26)
	Vacas de 2 o más partos	=C\$15*\$L\$5/100	=D\$15*\$L\$6/100	=E\$15*\$L\$7/100	=F\$15*\$L\$8/100	=G\$15*\$L\$9/100	=H\$15*\$L\$10/100	=I\$15*\$L\$11/100	=SUMA(\$C27:\$I27)
	Total	=SUMA(C\$26:C\$27)	=SUMA(D\$26:D\$27)	=SUMA(E\$26:E\$27)	=SUMA(F\$26:F\$27)	=SUMA(G\$26:G\$27)	=SUMA(H\$26:H\$27)	=SUMA(I\$26:I\$27)	=SUMA(\$C28:\$I28)
1.6. Población final del hato productor	Vacas de 1er parto	=14:\$14-\$26:\$26+\$25:\$25	=14:\$14-\$26:\$26+\$25:\$25	=14:\$14-\$26:\$26+\$25:\$25	=14:\$14-\$26:\$26+\$25:\$25	=14:\$14-\$26:\$26+\$25:\$25	=14:\$14-\$26:\$26+\$25:\$25	=14:\$14-\$26:\$26+\$25:\$25	=SUMA(\$C30:\$I30)
	Vacas de 2 o más partos	=C\$15-C\$27	=D\$15-D\$27	=E\$15-E\$27	=F\$15-F\$27	=G\$15-G\$27	=H\$15-H\$27	=I\$15-I\$27	=SUMA(\$C31:\$I31)
	Total	=SUMA(C\$30:C\$31)	=SUMA(D\$30:D\$31)	=SUMA(E\$30:E\$31)	=SUMA(F\$30:F\$31)	=SUMA(G\$30:G\$31)	=SUMA(H\$30:H\$31)	=SUMA(I\$30:I\$31)	=SUMA(\$C32:\$I32)

CONC./AÑO		1	2	3	4	5	6	7	TOTAL	
2. Partos	Vacas de 1er parto	$=(\$Q\$12+\$5:\$5) * (1 - \frac{\$L\$15}{\$100}) + \$24 + \$25:\25	$=\$5:\$5 * (1 - \frac{\$L\$15}{\$100}) + \$24 + \$24 + \$25:\$25$	$=\$5:\$5 * (1 - \frac{\$L\$15}{\$100}) + \$24 + \$24 + \$25:\$25$	$=\$5:\$5 * (1 - \frac{\$L\$15}{\$100}) + \$24 + \$24 + \$25:\$25$	$=\$5:\$5 * (1 - \frac{\$L\$15}{\$100}) + \$24 + \$24 + \$25:\$25$	$=\$5:\$5 * (1 - \frac{\$L\$15}{\$100}) + \$24 + \$24 + \$25:\$25$	$=\$5:\$5 * (1 - \frac{\$L\$15}{\$100}) + \$24 + \$24 + \$25:\$25$	=SUMA(\$C34:\$I34)	
	Vacas de 2 o más partos	$=(\$C\$15+\$C\$31) / 2 * (12 / \$S\$1)$	$=(\$D\$15+\$D\$31) / 2 * (12 / \$S\$1)$	$=(\$E\$15+\$E\$31) / 2 * (12 / \$S\$1)$	$=(\$F\$15+\$F\$31) / 2 * (12 / \$S\$1)$	$=(\$G\$15+\$G\$31) / 2 * (12 / \$S\$1)$	$=(\$H\$15+\$H\$31) / 2 * (12 / \$S\$1)$	$=(\$I\$15+\$I\$31) / 2 * (12 / \$S\$1)$	=SUMA(\$C35:\$I35)	
	Total	=SUMA(\$C34:\$C35)	=SUMA(\$D34:\$D35)	=SUMA(\$E34:\$E35)	=SUMA(\$F34:\$F35)	=SUMA(\$G34:\$G35)	=SUMA(\$H34:\$H35)	=SUMA(\$I34:\$I35)	=SUMA(\$C36:\$I36)	
3. Recría	Terneras	$=(\$C\$36/2) + \$C\$2 + \$N\$12 * (1 - \frac{\$L\$12}{\$100})$	$=(\$D\$36/2) + \$D\$2 * (1 - \frac{\$L\$12}{\$100})$	$=(\$E\$36/2) + \$E\$2 * (1 - \frac{\$L\$12}{\$100})$	$=(\$F\$36/2) + \$F\$2 * (1 - \frac{\$L\$12}{\$100})$	$=(\$G\$36/2) + \$G\$2 * (1 - \frac{\$L\$12}{\$100})$	$=(\$H\$36/2) + \$H\$2 * (1 - \frac{\$L\$12}{\$100})$	$=(\$I\$36/2) + \$I\$2 * (1 - \frac{\$L\$12}{\$100})$	=SUMA(\$C38:\$I38)	
	Terneras acumuladas	=C\$38	=D\$38+\$C\$38*\$O\$23	=E\$38+\$D\$38*\$O\$23	=F\$38+\$E\$38*\$O\$23	=G\$38+\$F\$38*\$O\$23	=H\$38+\$G\$38*\$O\$23	=I\$38+\$H\$38*\$O\$23	=SUMA(\$C39:\$I39)	
	Becerras	$=(\$C\$38 * \$P\$23 + \$C\$3 + \$O\$12) * (1 - \frac{\$L\$12}{\$100})$	$=(\$D\$38 * \$P\$23 + \$D\$3 + \$O\$12) * (1 - \frac{\$L\$12}{\$100})$	$=(\$E\$38 * \$P\$23 + \$E\$3 + \$O\$12) * (1 - \frac{\$L\$12}{\$100})$	$=(\$F\$38 * \$P\$23 + \$F\$3 + \$O\$12) * (1 - \frac{\$L\$12}{\$100})$	$=(\$G\$38 * \$P\$23 + \$G\$3 + \$O\$12) * (1 - \frac{\$L\$12}{\$100})$	$=(\$H\$38 * \$P\$23 + \$H\$3 + \$O\$12) * (1 - \frac{\$L\$12}{\$100})$	$=(\$I\$38 * \$P\$23 + \$I\$3 + \$O\$12) * (1 - \frac{\$L\$12}{\$100})$	$=(\$J\$38 * \$P\$23 + \$J\$3 + \$O\$12) * (1 - \frac{\$L\$12}{\$100})$	=SUMA(\$C40:\$I40)
	Becerras acumuladas	=C\$40	=D\$40+\$C\$40*\$P\$23	=E\$40+\$D\$40*\$P\$23	=F\$40+\$E\$40*\$P\$23	=G\$40+\$F\$40*\$P\$23	=H\$40+\$G\$40*\$P\$23	=I\$40+\$H\$40*\$P\$23	=SUMA(\$C41:\$I41)	
	Vaqs. (12ms-24 ms.)	$=(\$C\$4 + \$P\$12) * (1 - \frac{\$L\$14}{\$100}) + (\$O\$12 + \$C\$3/2) * (1 - \frac{\$L\$12}{\$100})$	$=(\$D\$38 + \$C\$3/2 + \$D\$3/2) * (1 - \frac{\$L\$13}{\$100}) + (\$O\$12 + \$D\$3/2) * (1 - \frac{\$L\$12}{\$100})$	$=(\$E\$38 + \$D\$3/2 + \$E\$3/2) * (1 - \frac{\$L\$13}{\$100}) + (\$O\$12 + \$E\$3/2) * (1 - \frac{\$L\$12}{\$100})$	$=(\$F\$38 + \$E\$3/2 + \$F\$3/2) * (1 - \frac{\$L\$13}{\$100}) + (\$O\$12 + \$F\$3/2) * (1 - \frac{\$L\$12}{\$100})$	$=(\$G\$38 + \$F\$3/2 + \$G\$3/2) * (1 - \frac{\$L\$13}{\$100}) + (\$O\$12 + \$G\$3/2) * (1 - \frac{\$L\$12}{\$100})$	$=(\$H\$38 + \$G\$3/2 + \$H\$3/2) * (1 - \frac{\$L\$13}{\$100}) + (\$O\$12 + \$H\$3/2) * (1 - \frac{\$L\$12}{\$100})$	$=(\$I\$38 + \$H\$3/2 + \$I\$3/2) * (1 - \frac{\$L\$13}{\$100}) + (\$O\$12 + \$I\$3/2) * (1 - \frac{\$L\$12}{\$100})$	=SUMA(\$C42:\$I42)	
	Vqs. acumuladas	=C\$42	=D\$42+\$C\$42	=E\$42+\$D\$42+\$C\$42*\$Q\$23	=F\$42+\$E\$42+\$D\$42*\$Q\$23	=G\$42+\$F\$42+\$E\$42*\$Q\$23	=H\$42+\$G\$42+\$F\$42*\$Q\$23	=I\$42+\$H\$42+\$G\$42*\$Q\$23	=SUMA(\$C43:\$I43)	
	Vaqs. Reep 24 ms.	$=(\$P\$12 + \$C\$4/2) * (1 - \frac{\$L\$14}{\$100}) * (1 - \frac{\$L\$15}{\$100})$	$=(\$D\$11 + \$D\$4/2 + \$D\$4/2) * (1 - \frac{\$L\$14}{\$100}) * (1 - \frac{\$L\$15}{\$100})$	$=(\$E\$11 + \$E\$4/2 + \$E\$4/2) * (1 - \frac{\$L\$14}{\$100}) * (1 - \frac{\$L\$15}{\$100})$	$=(\$F\$11 + \$F\$4/2 + \$F\$4/2) * (1 - \frac{\$L\$14}{\$100}) * (1 - \frac{\$L\$15}{\$100})$	$=(\$G\$11 + \$G\$4/2 + \$G\$4/2) * (1 - \frac{\$L\$14}{\$100}) * (1 - \frac{\$L\$15}{\$100})$	$=(\$H\$11 + \$H\$4/2 + \$H\$4/2) * (1 - \frac{\$L\$14}{\$100}) * (1 - \frac{\$L\$15}{\$100})$	$=(\$I\$11 + \$I\$4/2 + \$I\$4/2) * (1 - \frac{\$L\$14}{\$100}) * (1 - \frac{\$L\$15}{\$100})$	=SUMA(\$C44:\$I44)	

	CONC./AÑO	1	2	3	4	5	6	7	TOTAL
	Vqs. Reep. (Interv.)	$=(\$Q\$12+C\$5) * (1 - \$L\$15/100) + \$P\$12 * (1 -$	$=\$5. \$5 * (1 - \$L\$15/100) + (D\$44 * \$P\$25) + (E\$44 * \$Q\$23)$	$=\$5. \$5 * (1 - \$L\$15/100) + (E\$44 * \$P\$25) + (D\$44 * \$Q\$23)$	$=\$5. \$5 * (1 - \$L\$15/100) + (F\$44 * \$P\$25) + (E\$44 * \$Q\$23)$	$=\$5. \$5 * (1 - \$L\$15/100) + (G\$44 * \$P\$25) + (F\$44 * \$Q\$23)$	$=\$5. \$5 * (1 - \$L\$15/100) + (H\$44 * \$P\$25) + (G\$44 * \$Q\$23)$	$=\$5. \$5 * (1 - \$L\$15/100) + (I\$44 * \$P\$25) + (H\$44 * \$Q\$23)$	=SUMA(\$C45:\$I45)
4. Engorda	Terneros-Ms. y Ds.	$=(C\$36/2) * (1 - \$L\$16/100)$	$=(D\$36/2) * (1 - \$L\$16/100)$	$=(E\$36/2) * (1 - \$L\$16/100)$	$=(F\$36/2) * (1 - \$L\$16/100)$	$=(G\$36/2) * (1 - \$L\$16/100)$	$=(H\$36/2) * (1 - \$L\$16/100)$	$=(I\$36/2) * (1 - \$L\$16/100)$	=SUMA(\$C47:\$I47)
	Añojos	0	=C\$47*\$Q\$20	=D\$47*\$Q\$20	=E\$47*\$Q\$20	=F\$47*\$Q\$20	=G\$47*\$Q\$20	=H\$47*\$Q\$20	=SUMA(\$C48:\$I48)

5. Leche (Litros)	Vacas de 1er parto/año	$=SI(\$T\$1=0, (C\$14+C\$30)/2 * \$N\$16, (C\$14+C\$30)/2 * \$N\$16 * \$U\$1)$	$=SI(\$T\$1=0, (D\$14+D\$30)/2 * \$N\$16, (D\$14+D\$30)/2 * \$N\$16 * \$U\$1)$	$=SI(\$T\$1=0, (E\$14+E\$30)/2 * \$N\$16, (E\$14+E\$30)/2 * \$N\$16 * \$U\$1)$	$=SI(\$T\$1=0, (F\$14+F\$30)/2 * \$N\$16, (F\$14+F\$30)/2 * \$N\$16 * \$U\$1)$	$=SI(\$T\$1=0, (G\$14+G\$30)/2 * \$N\$16, (G\$14+G\$30)/2 * \$N\$16 * \$U\$1)$	$=SI(\$T\$1=0, (H\$14+H\$30)/2 * \$N\$16, (H\$14+H\$30)/2 * \$N\$16 * \$U\$1)$	$=SI(\$T\$1=0, (I\$14+I\$30)/2 * \$N\$16, (I\$14+I\$30)/2 * \$N\$16 * \$U\$1)$	=SUMA(\$C50:\$I50)
	Vacas de 2 o más ps/año	$=SI(\$T\$1=0, (C\$15+C\$31)/2 * \$N\$17, (C\$15+C\$31)/2 * \$N\$17 * \$U\$1)$	$=SI(\$T\$1=0, (D\$15+D\$31)/2 * \$N\$17, (D\$15+D\$31)/2 * \$N\$17 * \$U\$1)$	$=SI(\$T\$1=0, (E\$15+E\$31)/2 * \$N\$17, (E\$15+E\$31)/2 * \$N\$17 * \$U\$1)$	$=SI(\$T\$1=0, (F\$15+F\$31)/2 * \$N\$17, (F\$15+F\$31)/2 * \$N\$17 * \$U\$1)$	$=SI(\$T\$1=0, (G\$15+G\$31)/2 * \$N\$17, (G\$15+G\$31)/2 * \$N\$17 * \$U\$1)$	$=SI(\$T\$1=0, (H\$15+H\$31)/2 * \$N\$17, (H\$15+H\$31)/2 * \$N\$17 * \$U\$1)$	$=SI(\$T\$1=0, (I\$15+I\$31)/2 * \$N\$17, (I\$15+I\$31)/2 * \$N\$17 * \$U\$1)$	=SUMA(\$C51:\$I51)
	Total	=SUMA(C\$50:C\$51)	=SUMA(D\$50:D\$51)	=SUMA(E\$50:E\$51)	=SUMA(F\$50:F\$51)	=SUMA(G\$50:G\$51)	=SUMA(H\$50:H\$51)	=SUMA(I\$50:I\$51)	=SUMA(\$C52:\$I52)

6. Carne (Cabezas)	Terneras	$=(C\$2 + \$N\$12 + C\$36/2) * \$L\$12/100 * \$L\$24/100$	$=(D\$2 + D\$36/2) * \$L\$12/100 * \$L\$24/100$	$=(E\$2 + E\$36/2) * \$L\$12/100 * \$L\$24/100$	$=(F\$2 + F\$36/2) * \$L\$12/100 * \$L\$24/100$	$=(G\$2 + G\$36/2) * \$L\$12/100 * \$L\$24/100$	$=(H\$2 + H\$36/2) * \$L\$12/100 * \$L\$24/100$	$=(I\$2 + I\$36/2) * \$L\$12/100 * \$L\$24/100$	=SUMA(\$C54:\$I54)
	Becerras	$=(C\$38 * \$P\$23 + C\$3 + \$O\$12) * \$L\$13/100 * \$L\$25/100$	$=(C\$38 * \$O\$23 + D\$38 * \$P\$23 + D\$3) * \$L\$13/100 * \$L\$25/100$	$=(D\$38 * \$O\$23 + E\$38 * \$P\$23 + E\$3) * \$L\$13/100 * \$L\$25/100$	$=(E\$38 * \$O\$23 + F\$38 * \$P\$23 + F\$3) * \$L\$13/100 * \$L\$25/100$	$=(F\$38 * \$O\$23 + G\$38 * \$P\$23 + G\$3) * \$L\$13/100 * \$L\$25/100$	$=(G\$38 * \$O\$23 + H\$38 * \$P\$23 + H\$3) * \$L\$13/100 * \$L\$25/100$	$=(H\$38 * \$O\$23 + I\$38 * \$P\$23 + I\$3) * \$L\$13/100 * \$L\$25/100$	=SUMA(\$C55:\$I55)
	Vaquillas	$=C\$40 * \$L\$14/100 * \$L\$26/100$	$=D\$40 * \$L\$14/100 * \$L\$26/100$	$=E\$40 * \$L\$14/100 * \$L\$26/100$	$=F\$40 * \$L\$14/100 * \$L\$26/100$	$=G\$40 * \$L\$14/100 * \$L\$26/100$	$=H\$40 * \$L\$14/100 * \$L\$26/100$	$=I\$40 * \$L\$14/100 * \$L\$26/100$	=SUMA(\$C56:\$I56)

CONC./AÑO		1	2	3	4	5	6	7	TOTAL
Vaquillas reemplazo	de	$=($45:$45+$45:$45*(1-$L$15/100))*$L$15/100*$L$27/100$	$=($45:$45+$45:$45*(1-$L$15/100))*$L$15/100*$$	$=($45:$45+$45:$45*(1-$L$15/100))*$L$15/100*$L$27/100$	$=($45:$45+$45:$45*(1-$L$15/100))*$L$15/100$	$=($45:$45+$45:$45*(1-$L$15/100))*$L$15/100*$L$27/100$	$=($45:$45+$45:$45*(1-$L$15/100))*$L$15/100$	$=($45:$45+$45:$45*(1-$L$15/100))*$L$15/100$	=SUMA(\$C57:\$I57)
	Vacas	$=C$28*L28/100$	$=D$28*L28/100$	$=E$28*L28/100$	$=F$28*L28/100$	$=G$28*L28/100$	$=H$28*L28/100$	$=I$28*L28/100$	=J\$28*\$L\$28/100
	Terneros+Ds.-Añs.	$=(C$36/2)*(1-L16/100)$	$=(D$36/2)*(1-L16/100)$	$=(E$36/2)*(1-L16/100)$	$=(F$36/2)*(1-L16/100)$	$=(G$36/2)*(1-L16/100)$	$=(H$36/2)*(1-L16/100)$	$=(I$36/2)*(1-L16/100)$	=SUMA(\$C59:\$I59)
	Añojos	0	$=((C$47)*Q20)$	$=((D$47)*Q20)$	$=((E$47)*Q20)$	$=((F$47)*Q20)$	$=((G$47)*Q20)$	$=((H$47)*Q20)$	=SUMA(\$C60:\$I60)
7. Reemplazos	para venta.	$= $45:$45-$24:$24-$25:$25$	$= $45:$45-$24:$24-$25:$25$	$= $45:$45-$24:$24-$25:$25$	$= $45:$45-$24:$24-$25:$25$	$= $45:$45-$24:$24-$25:$25$	$= $45:$45-$24:$24-$25:$25$	$= $45:$45-$24:$24-$25:$25$	=SUMA(\$C62:\$I62)

CONC. AÑO		1	2	3	4	5	6	7	TOTAL
8. Ingresos (Pesos)	Leche	=C\$52*\$N\$24	=D\$52*\$N\$24 4	=E\$52*\$N\$24	=F\$52*\$N\$24 24	=G\$52*\$N\$24	=H\$52*\$N\$24 24	=I\$52*\$N\$24 24	=SUMA(\$C64:\$I64)
	Terneras	=C\$54*\$N\$25	=D\$54*\$N\$25 5	=E\$54*\$N\$25	=F\$54*\$N\$25 25	=G\$54*\$N\$25	=H\$54*\$N\$25 25	=I\$54*\$N\$25 25	=SUMA(\$C65:\$I65)
	Becerras	=C\$55:\$55*\$N\$26	=D\$55:\$55*\$N\$26 6	=E\$55:\$55*\$N\$26 6	=F\$55:\$55*\$N\$26 N\$26	=G\$55:\$55*\$N\$26 26	=H\$55:\$55*\$N\$26 N\$26	=I\$55:\$55*\$N\$26 N\$26	=SUMA(\$C66:\$I66)
	Vaquillas	=C\$56:\$56*\$N\$27	=D\$56:\$56*\$N\$27 7	=E\$56:\$56*\$N\$27 7	=F\$56:\$56*\$N\$27 N\$27	=G\$56:\$56*\$N\$27 27	=H\$56:\$56*\$N\$27 N\$27	=I\$56:\$56*\$N\$27 N\$27	=SUMA(\$C67:\$I67)
	Vacas y Vqs. R.	=C\$58*\$N\$28	=D\$58*\$N\$28 8	=E\$58*\$N\$28	=F\$58*\$N\$28 28	=G\$58*\$N\$28	=H\$58*\$N\$28 28	=I\$58*\$N\$28 28	=SUMA(\$C68:\$I68)
	Terneros	=C\$59*\$N\$29	=D\$59*\$N\$29 9	=E\$59*\$N\$29	=F\$59*\$N\$29 29	=G\$59*\$N\$29	=H\$59*\$N\$29 G68 N\$29	=I\$59*\$N\$29 29	=SUMA(\$C69:\$I69)
	Añojos	=C\$60*\$N\$30	=D\$60*\$N\$30 0	=E\$60*\$N\$30	=F\$60*\$N\$30 30	=G\$60*\$N\$30	=H\$60*\$N\$30 30	=I\$60*\$N\$30 30	=SUMA(\$C70:\$I70)
	Venta de Reemps.	=C\$62*\$N\$31		=E\$62*\$N\$31	=F\$62*\$N\$31 31	=G\$62*\$N\$31	=H\$62*\$N\$31 31	=I\$62*\$N\$31 31	=SUMA(\$C71:\$I71)
	TOTAL	=SUMA(C\$64:C\$71)	=SUMA(D\$64:D\$71) 4438743	=SUMA(E\$64:E\$71)	=SUMA(F\$64:F\$71)	=SUMA(G\$64:G\$71)	=SUMA(H\$64:H\$71)	=SUMA(I\$64:I\$71)	=SUMA(\$C72:\$I72)
9. Costos globales (\$)	=M\$42	=N\$42	=O\$42	=P\$42	=Q\$42	=R\$42	=S\$42	=T\$42	
10. Utilidades (\$)	=C\$72-C\$74	=D\$72-D\$74	=E\$72-E\$74	=F\$72-F\$74	=G\$72-G\$74	=H\$72-H\$74	=I\$72-I\$74	=J\$72-J\$74	
11. Beneficio/Costo	=C\$72/C\$74	=D\$72/D\$74	=E\$72/E\$74	=F\$72/F\$74 4	=G\$72/G\$74	=H\$72/H\$74 4	=I\$72/I\$74	=J\$72/J\$74	