

64



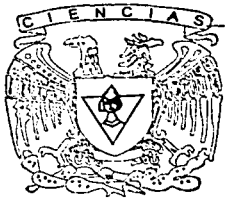
# UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTAD DE CIENCIAS

MODELO DE OPERACION DEL SEGURO DE VIDA  
APLICANDO UML PARA MODELADO DE DATOS

T E S I S  
QUE PARA OBTENER EL TITULO DE:  
ACTUARIO  
PRESENTA

JUAN JOSE LOPEZ GONZALEZ



FACULTAD DE CIENCIAS  
UNAM

Director de Tesis:

M. en C. María Guadalupe Elena Ibargüengoitia



FACULTAD DE CIENCIAS  
SECCION ESCOLAR

**TESIS CON  
FALTA DE ORIGEN**



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.



ESTADOS UNIDOS MEXICANOS  
SECRETARÍA DE EDUCACIÓN PÚBLICA

**M. EN C. ELENA DE OTEYZA DE OTEYZA**

Jefa de la División de Estudios Profesionales de la  
Facultad de Ciencias  
Presente

Comunico a usted que hemos revisado el trabajo escrito:

"Modelo de Operación del Seguro de Vida Aplicando UML para Modelado de Datos"

realizado por **JUAN JOSÉ LÓPEZ GONZÁLEZ**.

con número de cuenta **06703355 3**, quién cubrió los créditos de la carrera de **Actuario**.

Dicho trabajo cuenta con nuestro voto aprobatorio.

**Atentamente**

Director de Tesis  
Propietario

M. en C. Guadalupe E. Ibarra Eugoltin González

*Guadalupe E. Ibarra Eugoltin González*

Propietario

Mat. Adrián Girard Ibañeta

*Adrián Girard Ibañeta*

Propietario

Act. María Aurora Valdez Michel

*María Aurora Valdez Michel*

Suplente

Act. Martín Castillo Garduño

*Martín Castillo Garduño*

Suplente

Act. Humberto Santillana Loyo

*Humberto Santillana Loyo*

Consejo Departamental de Matemáticas

*[Firma]*

M. en C. José Antonio Flores



FACULTAD DE CIENCIAS  
CONSEJO DEPARTAMENTAL  
DE  
MATEMÁTICAS

**Agradecimiento**

---

***A MIS PADRES Y HERMANOS***

**AGRADEZCO SU AMOR  
AGRADEZCO SU GRAN APOYO  
AGRADEZCO SU EJEMPLO EN LA VIDA  
A MAGUITO CON TODO MI AMOR**

***A MI ESPOSA CARMEN SUÁREZ***

**LE AGRADEZCO SU AMISTAD  
LE AGRADEZCO SU CARIÑO  
LE AGRADEZCO SU AMOR  
LE AGRADEZCO SU APOYO  
LE AGRADEZCO LOS HIJOS  
QUE DIOS NOS DIO**

***A MI HIJO JONATHAN LÓPEZ SUÁREZ***

**POR SU TENACIDAD PARA  
HACER LO QUE SE PROPONE**

***A MI HIJO IVAN I. LÓPEZ SUÁREZ***

**POR SU GRAN DETERMINACIÓN  
PARA LOGRAR SUS PROPÓSITOS**

**A MIS AMIGOS**

**POR SU APOYO Y SU AMISTAD**

***A LUPITA IBARGÜENGOITIA***

**LE AGRADEZCO EL VALIOSO  
TIEMPO QUE DEDICO PARA LA  
REALIZACIÓN DE ESTE  
PROYECTO**

***AGRADEZCO EL APOYO DE:***

**ADRIÁN GIRARD  
AURORA VALDEZ  
MARINA CASTILLO  
HUMBERTO SANTILLANA**

# Índice

Introducción.....	i
Capítulo 1	
1. Modelado de Datos .....	01
1.1. Definición .....	01
1.2. Un Modelo de Datos Relacional .....	01
1.3. Diagramas de Entidad Relación .....	02
1.4. Multiplicidad .....	03
1.5. OO (Análisis y Diseño Orientado a Objetos) .....	03
1.6. Conceptos de Modelado .....	04
1.6.1. Clase .....	04
1.6.2. Objetos .....	05
1.6.3. Relaciones .....	06
1.6.4. Asociación .....	07
1.6.5. Agregación .....	08
1.6.6. Herencia .....	08
1.6.7. Generalización .....	09
Capítulo 2	
2. UML (Lenguaje Unificado de Modelado) .....	10
2.1. Antecedentes .....	10
2.2. Definición .....	11
2.3. Conceptos y Terminología del UML .....	12
2.5. Componentes .....	13
2.6. Diagramas de UML .....	21
2.6.1. Diagrama de Clase .....	21
2.6.2. Diagrama de Objetos .....	22
2.6.3. Diagrama de Casos de Uso .....	22
2.6.4. Diagrama de Secuencia y Colaboración .....	23
2.6.5. Diagrama de Estado .....	24
2.6.6. Diagrama de Actividad .....	25
2.6.7. Diagrama de Componentes .....	25
2.6.8. Diagrama de Instalación .....	26
2.6. Arquitectura .....	26
2.7. Herramientas de Apoyo al Modelado .....	27
2.7.1. Evolución de las Herramientas .....	27
2.7.2. Herramientas CASE .....	28
2.7.3. Herramientas PSEE .....	30

## Índice

---

### Capítulo 3

3. Modelo de Operaciones del Negocio del Seguro de Vida (Nota Técnica) .....	34
3.1. Definición de la Nota Técnica.....	34
3.2. Sistema Integral para la administración del modelo de seguros .....	34
3.2.1. Descripción de la Información.....	38

### Capítulo 4

4. Modelo de Datos UML del Negocio del Seguro de Vida.....	49
4.1. Determinar los Actores .....	49
4.2. Determinar Diagramas los Casos de Uso .....	54
4.3. Determinar Diagramas de Actividades .....	63
4.4. Determinar Diagramas de Clases o Estados.....	69

Conclusiones.....	76
-------------------	----

Anexo A.....	78
--------------	----

Bibliografía.....	89
-------------------	----

## Introducción

Durante las últimas 3 décadas, nacen muchas técnicas de diagramación, como los Diagramas de Descomposición, Diagramas de Dependencia, Diagramas de Flujo de datos, Cartas Estructuradas, Diagramas HIPO, Diagramas de Entidad Relación, Diagramas de Datos Estructurados, etc., también surgen las Metodologías de Modelado, Análisis, Diseño y Programación, Orientados a Objetos.

En el año de 1994, surge la necesidad de crear un solo lenguaje de modelado visual y consistente, en el que se pudieran expresar los resultados de las numerosas metodologías de Orientación a Objetos existentes hasta la fecha. En noviembre de 1997, surge la versión estandarizada del Lenguaje Unificado de Modelado.

Por otro lado las compañías de seguros en general, utilizan un sistema integral, para mantener y desarrollar los procesos de operación del negocio, y pueden apoyarse en estos sistemas prácticamente desde que surge la idea o la necesidad de crear y diseñar los nuevos planes de seguros.

El área técnica se encarga del diseño del plan, y al definirlo en el sistema integral establece las condiciones del proceso del negocio, para las diferentes áreas o departamentos que intervienen en la operación.

Entonces las áreas de expedición, facturación y cobranzas, agentes, servicio a asegurados, contabilidad, y operación y sistemas, deben apegarse a las políticas y procedimientos definidos por el técnico.

Con base en el Lenguaje Unificado de Modelado (UML), se modela el desarrollo de la operación del negocio de una compañía de seguros en el ramo de vida, mostrando sus diferentes etapas, desde la creación del modelo conceptual, de la autorización del plan por las autoridades (CNSF, Comisión Nacional de Seguros y Fianzas), de la definición y administración en el sistema integral, así como de la responsabilidad y operación de cada área.

El desarrollo de modelos de la operación del negocio en una compañía de seguros de vida, nos proporciona las siguientes facilidades:

- Desarrollo de productos adecuados a las necesidades de la empresa
- Conocer las responsabilidades del flujo de la operación
- Entender la estructura del proceso del negocio

- Hacer ingeniería (directa e inversa)
- Elaborar procesos más eficientes (modelando estructuras de objetos)
- Efectuar procesos de simulación, para determinar opciones óptimas de negocio
- Establecer una metodología para visualizar y documentar las estructuras de la compañía

Un modelo es la simplificación de la realidad, creado para comprender mejor el sistema que se está creando. Podemos mencionar nueve modelos que en conjunto cubren las decisiones importantes implicadas en la visualización, especificación, construcción y documentación de un sistema con gran cantidad de software:

- Modelo del negocio                      Establece una abstracción de la organización.
- Modelo del dominio                      Establece el contexto del sistema.
- Modelo de casos de uso                Establece los requisitos funcionales del sistema
- Modelo de análisis                      Establece un diseño de las ideas
- Modelo de diseño                        Establece el vocabulario del problema y su solución.
  
- Modelo del proceso                      Establece los mecanismos de concurrencias y sincronización del sistema.
  
- Modelo de instalación                Establece la topología del hardware sobre la cual se ejecutará el sistema.
  
- Modelo de implementación           Establece las partes que se utilizarán para ensamblar y hacer disponible el sistema físico.
  
- Modelo de pruebas                      Establece las formas de validar y verificar el sistema.

Cuando se ve un sistema de software desde cualquier perspectiva mediante UML (Lenguaje Unificado de Modelado), se usan los diagramas para organizar los elementos de interés. UML define nueve tipos de diagramas, que se pueden mezclar y conectarse para componer cada vista. Por ejemplo, los aspectos **estáticos** de la vista de implementación de un sistema pueden visualizarse con diagramas de componentes; los aspectos **dinámicos** de la misma vista de implementación pueden visualizarse con diagramas de interacción. Análogamente,



los aspectos **estáticos** de la base de datos de un sistema podrían visualizarse con diagramas de clases; sus aspectos **dinámicos** podrían verse con diagramas de colaboración.

El objetivo del presente trabajo es desarrollar modelos de la operación del seguro de vida entre las distintas áreas que conforman una compañía de seguros, utilizando el Lenguaje Unificado de Modelado (UML).

El capítulo uno muestra básicamente el proceso de modelado de datos, con base en el proceso y desarrollo de bases de datos relacionales, con base en las herramientas que se desarrollaron al respecto, como son los diagramas de entidad relación y su estructura y características. También se describe el Análisis y Diseño Orientado a Objetos, así como los conceptos del modelado y sus componentes.

El segundo capítulo proporciona una descripción detallada de los componentes del Lenguaje Unificado de Modelado (UML), iniciando desde sus antecedentes históricos, su definición, conceptos y terminología. También se describen detalladamente las 9 clases de diagramas que lo componen. Encontraremos la descripción de las herramientas de apoyo al modelado y su evolución a través del tiempo, pasando por las herramientas CASE (Ingeniería de Software Asistida por Computadora), y los PSEE's (Entornos de Ingeniería de Software Centrados en el Proceso).

El tercer capítulo, describe las funciones del proceso del negocio del seguro de vida con base en un sistema integral automatizado, para la administración de la cartera correspondiente. Se toma en cuenta que el nuevo plan de seguros es modelado a través de paquetes o sistemas de simulación, en los que se puede ir "jugando" literalmente con los parámetros que componen el sistema y que intervienen en la definición de las "Notas Técnicas", interviniendo en éstas factores financieros, factores de mortalidad, tasas de interés, etc.

En el capítulo cuatro se presenta el modelado de la operación del seguro de vida, utilizando diagramas de UML para determinar los actores que intervienen en el proceso del negocio, los diagramas de casos de uso para determinar las secuencias y acciones de las distintas áreas; los diagramas de actividad, que nos permiten modelar los aspectos dinámicos del sistema que existen entre las diferentes áreas, y los diagramas de clases que modelan un conjunto de clases, interfases y colaboraciones, así como sus relaciones.

## Capítulo 1

### 1. Modelado de Datos

#### 1.1. Definición

Iniciaremos presentando los conceptos básicos de los sistemas de administración de datos y posteriormente presentaremos las diferencias entre los sistemas convencionales de organización de datos y de los sistemas de bases de datos, haciendo énfasis en los beneficios relativos de ambos tipos de sistemas. Y finalmente se presentarán las técnicas básicas de organización de datos y se compara cada tipo de organización.

#### 1.2 Un Modelo de Datos Relacional

Dada una colección de conjuntos  $D_1, D_2, \dots, D_n$  (no necesariamente distintos)  $R$  es una relación en estos  $n$  conjuntos si existe un conjunto ordenado de  $n$ -duplas,  $(d_1, d_2, d_3, \dots, d_n)$ , tales que  $d_1$  pertenece a  $D_1$ ,  $d_2$  pertenece a  $D_2, \dots$ ,  $d_n$  pertenece a  $D_n$ . Los conjuntos  $D_1, D_2, \dots, D_n$  están en el dominio de  $R$ . El valor  $n$  es el grado de  $R$ . La figura 1.1, ilustra una relación llamada PARTE, de grado 5, definida en el dominio de  $P\#$  (número de parte),  $PNOMBRE$  (nombre de parte),  $COLOR$  (color de la parte),  $PESO$  (peso de la parte), y  $CIUDAD$  ( lugar en donde la parte está almacenada). El dominio  $COLOR$ , por ejemplo, es el conjunto de todos los colores válidos de las partes; nótese que puede haber colores incluidos en este dominio que actualmente no aparecen en la relación de PARTE en este momento en particular.

Como se ilustra en la figura 1.1, es conveniente representar una relación como una tabla.

PARTE

P#	NOMBRE DE PARTE	COLOR	PESO	CIUDAD
P1	Nuez	Rojo	12 gr.	México, D.F.
P2	Tornillo	Verde	17 gr.	Puebla, Pue.
P3	Desarmador	Azul	17 gr.	Veracruz, Ver.
P4	Desarmador	Rojo	14 gr.	Monterrey, NL
P5	A. levas	Azul	12 gr.	Puebla, Pue.
P6	Diente	Rojo	19 gr.	México, D.F.

Figura 1.1 La relación PARTE

Cada renglón de la tabla representa una n-dupla de la relación. El número de duplas en una relación es llamado la cardinalidad de la relación; por ejemplo la cardinalidad de la tabla de la figura 1.1, es seis.

### 1.3 Diagramas de Entidad Relación

Los diagramas de Entidad Relación asignan atributos a las entidades y definen relaciones entre las entidades dando por resultado un modelo de base de datos o un esquema de cualquier sistema que se entiende de manera fácil.

Un diagrama de Entidad Relación (ER), es el modelo que se usa para representar las entidades en el modelo relacional y sus relaciones, tiene tres elementos:

- Entidades
- Relaciones
- Atributos

Una entidad es una persona, lugar, objeto o concepto acerca del cual se almacenan los datos. Las entidades típicas de un fabricante son los empleados, proveedores, cuentas por cobrar y cuentas por pagar, equipo e inventarios.

Las entidades son similares a nombres y los atributos corresponden a los adjetivos.

Las relaciones son similares a los verbos. Sólo las entidades tienen atributos.

#### Construcción de diagramas de Entidad Relación

El modelo ER es un modelo conceptual, independiente de máquinas o restricciones físicas.

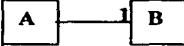
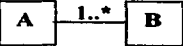
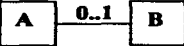
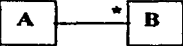
Los rectángulos representan entidades y los rombos y las líneas de conexión muestran las relaciones, y los atributos se representan por círculos.

## 1.4 Multiplicidad

### Multiplicidad

Una vez que se identifican las entidades, la relación entre dos entidades cualesquiera se asigna como (ver figura 1.2):

1. de una a una (1:1)
2. de una a muchas (1:M) ó (1:\*)
3. de muchas a muchas (M:N) ó (\*:\*)

	Una A siempre se asocia con una B	Una A siempre se asocia con una o más B	Una A siempre se asocia con ninguna o con una B	Una A siempre se asocia con ninguna, con una o con más B
UML				

\* puede ser unidireccional

Figura 1.2 Notaciones de Cardinalidad

## 1.5. OO (Análisis y Diseño Orientado a Objetos)

El diseño y modelado Orientado a Objetos, es una nueva forma de pensamiento sobre problemas usando modelos organizados alrededor del mundo real. La construcción fundamental es un objeto, que combina tanto la estructura como el comportamiento de una sola entidad. Los modelos Orientados a Objetos, son usados para el entendimiento de problemas, en la comunicación con los expertos de aplicaciones, modelado de empresas, preparación de documentación, y diseño de programas de bases de datos. Representando una metodología de desarrollo de software Orientada a Objetos. La Técnica de Modelado de Objetos (OMT), que se extiende desde el diseño hasta la implementación. Primero se construye un modelo del análisis con los aspectos abstractos esenciales del dominio de la aplicación sin observar la eventual implementación. Este modelo contiene objetos fundados en el dominio de la aplicación, incluyendo una descripción de las propiedades del objeto y

su comportamiento. Entonces las decisiones del diseño se efectúan y los detalles son agregados al modelo para describir y optimizar la implementación. Los objetos del dominio de la aplicación del marco de trabajo del modelo del diseño, se ha implementado en términos de los objetos del dominio de la computadora. Finalmente el modelo de diseño es implementado en un lenguaje de programación, base de datos o hardware.

Se describe una notación gráfica para expresar modelos Orientados a Objetos. El dominio de la aplicación y los objetos del dominio del computador pueden ser modelados, diseñados e implementados usando los mismos conceptos y notación Orientada a Objetos. Notación sin costuras es usada para el análisis el diseño y la implementación. Así la información agregada en una etapa de desarrollo no se pierde en la nueva etapa.

### **1.6. Conceptos de Modelado.**

El desarrollo Orientado a Objetos, es un proceso conceptual independiente de un lenguaje de programación hasta sus etapas finales. El desarrollo Orientado a Objetos, es fundamentalmente una nueva forma de pensamiento, y no una técnica de programación. Su gran beneficio viene a la ayuda de los especificadores, desarrolladores, junto con los conceptos abstractos de clases, y la comunicación de unos con otros. Esto puede servir como medio para la especificación, el análisis, la documentación y la creación de interfases así como para la programación. Aun como una herramienta de programación , puede tener varios objetivos, incluyendo los lenguajes de programación convencionales y las bases de datos, o lenguajes Orientados a Objetos.

#### **1.6.1. Clase**

Un objeto clase, describe a un grupo de objetos con propiedades similares (atributos), y comportamiento común (operaciones), relaciones comunes a otros objetos y semántica común. Persona, compañía, animal, proceso y ventana, todos son clases de objetos. Cada persona tiene una edad, un IQ, y puede trabajar en un empleo. Cada proceso tiene un propietario, una propiedad, y una lista de recursos necesarios.

Objetos y clase de objetos, frecuentemente aparecen como sustantivos de la descripción del problema. La abreviación de clase, frecuentemente es usada en lugar de clase de objeto. Los objetos en una clase tienen los mismos patrones de atributos y el mismo

comportamiento, la mayoría de los objetos manejan su individualidad de las diferencias en los valores de sus atributos y sus relaciones con otros objetos. Sin embargo, objetos con valores de atributos y relaciones idénticas, son posibles.

Los objetos en una clase comparten el propósito de una semántica común, además de los atributos y el comportamiento comunes.

Así aunque un granero, y un caballo, ambos tienen costo, edad, pueden pertenecer a clases distintas. Si el granero y el caballo fueran relacionados como productos financieros, podrían pertenecer a la misma clase. Si el desarrollador, tomara en consideración que una persona pinta un granero y alimenta un caballo, serán modelados como clases diferentes. La interpretación de la semántica, depende del propósito de cada aplicación.

Cada objeto "conoce" sus clases, la mayoría de los lenguajes de programación orientados a objetos pueden determinar una clase de objeto al momento del proceso. Una clase de objeto, es una propiedad implícita de un objeto.

Si los objetos son el foco del modelado de objetos, ¿por qué molestarse con las clases?, la noción de abstracción es el corazón del tema. Mediante la agrupación de objetos en clases, la abstracción le da al modelado el poder y la habilidad para generalizar desde unas pocas cosas específicas a muchas cosas similares. Por ejemplo, todas las elipses comparten el mismo procedimiento para dibujarlas, para el cálculo de sus áreas, o para la prueba de intersección con una línea; los polígonos tendrán una serie de procedimientos por separado. Aun casos especiales, como los círculos y los cuadrados, pueden usar procedimientos generales aunque más eficientes.

### **1.6.2. Objetos**

El propósito del modelado de objetos, es describir objetos. Por ejemplo, Juan López, Compañía Simple, Súper Perro, número de proceso 19, y la ventana de alta, son objetos. Un objeto es simple y a veces toma sentido en el contexto de una aplicación.

Es posible definir objetos como un concepto, o como una abstracción o cosa. Los objetos sirven para dos propósitos: promueven el entendimiento del mundo real y proveen bases prácticas para la implementación.

Todos los objetos tienen identidad y son distinguibles. Dos manzanas con el mismo color, forma y textura siguen siendo manzanas individuales; una persona puede comer una y

después la otra. Similarmente, gemelos idénticos son dos personas distintas, aunque parezcan la misma persona. El termino identidad, significa que los objetos son distintos por su existencia inherente y no por las propiedades que los describen y que ellos puedan tener. Algunas veces, objeto significa una sola cosa, otras veces se refiere a un grupo de cosas similares. Usualmente el contexto resuelve cualquier ambigüedad. Cuando queremos ser precisos y referirnos a una cosa exactamente, usamos la frase instancia del objeto. Usaremos la frase clase de objeto para referirnos a un grupo de cosas similares.

### 1.6.3. Relaciones

Las relaciones son conexiones semánticas materializadas entre los elementos de un modelo. En las asociaciones entre clases y objetos, para un objeto que envía un mensaje a otro, se deben asociar con cada uno de los otros en alguna manera. Podemos imaginar que alguna asociación existe entre las siguientes parejas de objetos:

<b>Objeto</b>	<b>Asociación</b>	<b>Objeto</b>
Motor	¿?	Pistón
Computadora de vuelo	¿?	Motor
Censor linear	¿?	Censor
Nave	¿?	tablón
Elevador	¿?	Botones de mando
Banco	¿?	Cuenta bancaria

El proceso de análisis identifica las llaves de objetos en el sistema, también como se van a relacionar unos con otros. Todas las asociaciones pueden ser (opcionalmente) nombradas cuando se clarifique la asociación. Las relaciones anteriores se pueden redefinir:

<b>Objeto</b>	<b>Asociación</b>	<b>Objeto</b>
Motor	<i>tiene un</i>	Pistón
Computadora de vuelo	<i>controla un</i>	Motor
Censor linear	<i>es un tipo de</i>	Censor
Nave	<i>contiene algún</i>	Tablón
Elevador	<i>es llamado por</i>	Botones de llamado

Banco

*almacena dinero en una*

Cuenta bancaria

Existen relaciones entre clases. Existen cinco tipos elementales de relaciones entre objetos: *asociación, agregación, composición, generalización y refinamiento.*

*Asociaciones*, son relaciones que se manifiestan por sí mismas en tiempo de proceso, para permitir el intercambio de mensajes entre objetos.

*Agregación*, es usada cuando un objeto lógico o físicamente contiene a otro.

*Composición*, es una forma fuerte de agregación en la cual el propietario es responsable explícitamente por la creación y destrucción de las partes del objeto.

*Refinamiento*, son relaciones que soportan elaboraciones genéricas o preestablecidas de especificaciones de clases incompletas.

#### **1.6.4. Asociación**

Cuando un objeto utiliza los servicios de otro, pero no es el propietario, el objeto tiene una asociación. Las asociaciones son apropiadas cuando se cumple alguna de las siguientes reglas:

- Un objeto usa los servicios de otro, pero no está agregado a él.
- El ciclo de vida de las clases de uso no son la responsabilidad del usuario de las clases; esto es, no es responsable por la creación y la destrucción de los objetos usados.
- La asociación entre objetos está más suelta que una de agregación.
- La asociación puede estar caracterizada como cliente servidor.
- Los objetos usados son compartidos y usados igualmente por muchos otros.



### 1.6.5. Agregación

Una relación de agregación se aplica cuando un objeto físicamente o conceptualmente contiene a otro. Una clase larga es referida como el propietario o el todo, y tiene forma de diamante al final de la agregación. La clase más pequeña es el propietario, la parte o la clase componente. El propietario es típicamente responsable por la creación y destrucción de la clase apropiada. Se proveen componentes de agregados a ser compartidos entre propietarios. Cuando un objeto comparte la propiedad, alguna regla ideal para especificar quien tiene la responsabilidad para su creación y destrucción.

En programación Windows, una ventana es una especie de objeto que contiene un *área cliente*. El *área cliente* no puede estar por sí misma, sin estar contenida por una ventana. El *área cliente* existe durante la creación de la ventana y es destruida cuando la ventana se destruye.

La composición es una forma fuerte de agregación. Los componentes normalmente son mostrados por la inclusión actual de los componentes de clase dentro de la composición. Alternativamente, una asociación de agregación puede ser usada, pero mostrada con un diamante relleno. Los componentes compuestos, no pueden ser compartidos (esto es, porque sólo tienen un propietario) y el compuesto es requerido para crear y destruir sus componentes. Un uso común de compuestos es un proyecto activo, esto es, objetos que son las raíces de los hilos. Estos objetos activos crean los hilos en los cuales operar. Los compuestos reciben mensajes y eventos de los RTOS (Real-Time Operation System) y otros hilos, y los despachan a los componentes apropiados dentro de sus propios hilos.

### 1.6.6. Herencia

Es el reparto de atributos y operaciones entre clases basadas en una relación jerárquica. Una clase puede estar definida ampliamente, y redefinida dentro, y redefinida en, y así sucesivamente formando una subclase. Cada subclase incorpora, o hereda, todas las propiedades de la superclase y agrega sus (propias) propiedades. Las propiedades de la superclase no necesitan estar repetidas en cada subclase. Por ejemplo, pantalla que se enrolla o analiza (scrolling) y la ventana fija, son subclases de ventana. Ambas subclases heredan las propiedades de ventana, como una región visible en una pantalla. Ventana enrollable o de análisis, agrega una barra de movimiento y una compensación. La habilidad

de obtener un factor de propiedades comunes y heredar las propiedades de la superclase pueden reducir de forma importante la repetición dentro de los diseños y programas, siendo una de las principales ventajas de un sistema Orientado a Objetos.

#### **1.6.7. Generalización**

Cuando una clase es una especialización de otra, la relación es llamada *generalización o herencia*. Lo que significa que el hijo o clase descendiente tiene todas las características definidas por el padre, aunque esta puede especializarse. El hijo además puede extender la clase de su padre agregando comportamientos y atributos adicionales. El fundamento de generalización es que es “es-una-clase-de” relación entre clases. Un mamífero es-una-clase-de animal y un censor infrarrojo es una-clase-de- censor.

Tipos de jerarquías son creadas desde las clases y sus relaciones de herencias. Tales jerarquías desde la base de algunas clases de marcos de trabajo, tales como MFC (Microsoft Foundation Classes) y Borland's OWL (Object Windows Library).

## Capítulo 2

### 2. UML (Lenguaje Unificado de Modelado)

#### 2.1. Antecedentes

A fines de la década de los años ochenta y principios de los noventa, surgen lenguajes de modelado orientados a objetos, iniciando la experimentación de alternativas para el análisis y diseño.

El número de métodos orientados a objetos, se incremento de unos diez a más de cincuenta entre los años de 1989 a 1994, por lo que encontrar un método adecuado era muy problemático.

Entre los métodos más notables, surgieron los de BOOCH, OOSE ( Ingeniería de Software Orientado a Objetos) de Jacobson y el OMT (Técnica de Modelado de Objetos) de Rumbaugh, así como Fusion de Coad-Yourdon.

El método BOOCH, fue un método expresivo en las etapas de diseño, y las fases de construcción del proyecto.

OOSE es un método que proporciona el concepto de los casos de uso, como una forma para el manejo de la captura de requerimientos, análisis y diseño de alto nivel.

OMT fue más usado para el análisis y la información más intensiva del sistema.

A mediados de los años noventas, Grady Booch (Rational Software Corporation), Ivar Jacobson (Objector), y James Rumbaugh (General Electric), se reunieron con la finalidad de reunir las ideas expresadas por los demás métodos, siendo reconocidos a nivel mundial como los líderes del método orientado a objetos, decididos a desarrollar un lenguaje unificado de modelado en forma conjunta.

Al iniciar la unificación del lenguaje de modelado UML, establecieron tres objetivos:

1. Para el modelado de sistemas, desde lo conceptual, al artefacto ejecutable, utilizando técnicas orientadas a objetos.
2. Para dirigir los problemas desde escala inherentemente compleja, hasta sistemas de misión crítica.

3. Para crear un lenguaje de modelado, utilizable tanto por máquinas como por humanos.

Visualizando que un lenguaje para su uso en el análisis y diseño orientado a objetos, es diferente a uno solo para la programación.

El desarrollo del nuevo lenguaje UML (Unified Modeling Language), se inició oficialmente en octubre de 1994, cuando Rumbaugh, se unió con Booch y Rational.

Algunos socios contribuyeron para definir la versión 1.0 de UML, como Digital Equipment Corporation, Hewlett Packard, I-Logic Intellicorp, IBM, entre otras. Resultando en una versión del lenguaje de modelado bien definida, expresiva, poderosa, y aplicable a una amplia gama de problemas.

En julio de 1997, se le presentó a la OMG (Object Management Group), la versión revisada y estandarizada de UML. En septiembre del mismo año esta versión fue aceptada por la Fuerza de Análisis y Diseño de Tareas de la OMG (ADTF), y por el bufete de arquitectura de la OMG, y por el voto de todos los miembros de la OMG, UML 1.1 fue adoptado como estándar por la OMG en noviembre 14 de 1997.

## 2.2. Definición

El Lenguaje Unificado de Modelado, es un lenguaje estándar para escribir anteproyectos de software, UML, permite visualizar, especificar, construir y documentar los artefactos de los sistemas de software.

Para el aprendizaje del UML, se requiere aprender tres elementos principales:

1. Bloques de construcción básicos.
2. Las reglas para poder unir estos bloques.
3. Algunos mecanismos comunes que aplican a través del lenguaje.

UML, es un lenguaje de modelado cuyo vocabulario y reglas se enfocan en la representación física y conceptual de un sistema.

En el proceso de desarrollo de software, se debe decidir que modelos se deben crear ya que ni el vocabulario ni las reglas del lenguaje de modelado lo dicen. El lenguaje del UML, dice como crear y leer los modelos, por lo que un proceso bien definido debe guiar para decidir los artefactos que se necesitan producir y las actividades que se tienen que realizar.

### 2.3. Conceptos y Terminología del UML

UML, es un lenguaje para:

➤ **Visualización.**

EL Modelado ayuda a visualizar un sistema, como está, o como queremos que sea.

➤ **Especificación.**

Modelar permite especificar la estructura o el comportamiento de un sistema, especificando las decisiones importantes del análisis, diseño e implementación, que deben ser tomadas en el desarrollo y distribución de un sistema de software.

➤ **Construcción.**

Modelar proporciona un patrón que nos guía en la construcción de un sistema.

UML, no es un lenguaje de programación, pero permite la conexión de sus modelos a una gran variedad de lenguajes de programación, tales como Java, C++, o Visual Basic, o en tablas de bases de datos relacionales.

➤ **Documentación.**

El modelado documenta las decisiones que se toman.

UML ayuda a manejar la documentación de la arquitectura de un sistema y sus detalles, provee además un lenguaje para expresar los requerimientos y pruebas.

## 2.4. Componentes

### Un modelo conceptual de UML

El modelo conceptual del lenguaje tiene tres elementos principales que son: los bloques de construcción básica de UML; las reglas que indican como estos bloques de construcción pueden unirse, y mecanismos comunes.

Dentro de los bloques se tienen:

1. Cosas

Son la abstracción y se consideran como los ciudadanos de primera clase en un modelo.

2. Relaciones

Unen las cosas entre sí.

3. Diagramas

Agrupan colecciones interesantes de cosas.

Existen cuatro tipos de cosas en UML, que son los bloques de construcción básicos orientados a objetos:

1. Cosas Estructurales

2. Cosas del Comportamiento

3. Cosas de Agrupamiento

4. Cosas de Anotación

### Cosas Estructurales

Las Cosas estructurales, son los sustantivos de los modelos de UML. Son la parte más estática del modelo, representando elementos que son conceptuales o físicos. Hay siete clases de cosas estructurales.

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

Gráficamente, una clase es presentada como un rectángulo. Usualmente incluye su nombre, atributos, y operaciones (figura 2-1).

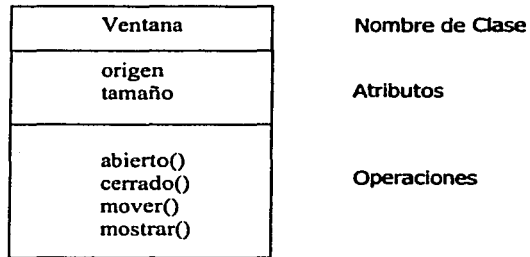


Figura 2-1: Clases

2. Una **interfaz** es una colección de operaciones que se emplea para especificar un servicio de una clase o un componente.

Una interfaz es un descriptor de las operaciones visibles externamente de una clase u otra entidad (tales como los paquetes), que no especifica la estructura interna.

Gráficamente una interfaz está presentada como un círculo junto con un nombre. Una interfaz raramente aparece sola, y típicamente esta unida a la clase o componente que realiza la interfaz (figura 2-2).



Nombre

Figura 2-2 Interfaces

3. Una **colaboración** define una interacción y es una sociedad de roles, y otros elementos que trabajan juntos para proveer algún comportamiento de cooperación, es mayor que la suma de sus elementos.

Las colaboraciones tienen una estructura, también un comportamiento, y dimensiones. Una clase dada puede participar en varias colaboraciones. Estas colaboraciones por tanto representan la implementación de patrones que maquillan un sistema.

Gráficamente la colaboración está representada como una elipse con líneas punteadas. Usualmente sólo incluye el nombre (figura 2-3).

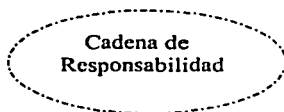


Figura 2-3 Colaboración

4. Un *caso de uso*, es una descripción de un conjunto de acciones que un sistema realiza, que produce un resultado observable de valor para un actor en particular.

Es usado para estructurar el comportamiento en un modelo. Un caso de uso, es realizado por una colaboración.

Gráficamente un caso de uso está representado como una elipse con líneas sólidas, usualmente incluye sólo el nombre (figura 2-4).

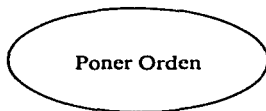


Figura 2-4 Casos de Uso

5. Una *clase activa* es una clase cuyos objetos pertenecen a uno o más procesos o hilos y por tanto pueden iniciar el control de la actividad.

Es sólo como una clase de excepción que son elementos que representan objetos cuyo comportamiento es concurrente con otros elementos.

Gráficamente una clase activa es representada como una clase, pero con líneas gruesas, usualmente incluye sus nombres, atributos, y operaciones (figura 2-5).



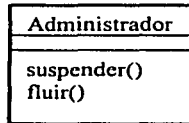


Figura 2-5 Clase Activa

6. Un **componente** es una parte física y reemplazable de un sistema que conforma y provee la realización de un conjunto de interfaces. En un sistema, se encuentran diferentes clases de componentes: de instalación, artefactos de proceso de desarrollo, tal como archivos de código fuente.

Típicamente representa el empaquetamiento físico de elementos lógicos como las clases, interfaces, y colaboraciones.

Gráficamente, un componente es representado como un rectángulo con lengüetas, usualmente incluye sólo su nombre (figura 2-6).

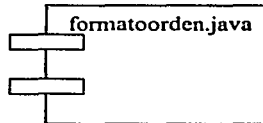


Figura 2-6 Componentes

7. Un **nodo** es una cosa física, que existe en el momento de proceso y representa un recurso computacional, generalmente teniendo al menos alguna memoria y, frecuentemente capacidad de proceso.

Gráficamente, un nodo es representado como un cubo, usualmente incluye sólo su nombre (figura 2-7).

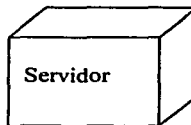


Figura 2-7 Nodos

Estas siete cosas -clases, interfaces, colaboración, casos de uso, clases activas, componentes, y nodos- son las cosas de la estructura básica que se pueden incluir en un modelo de UML.

**\*) Cosas del Comportamiento**

Las cosas del comportamiento son las partes dinámicas de los modelos de UML.

Existen dos tipos primarios de cosas del comportamiento.

1. Una interacción es un relación que comprende a un conjunto de mensajes de intercambio, entre un conjunto de objetos dentro de un contexto particular para cumplir un propósito específico.

Gráficamente, un mensaje es representado como una línea directa, casi siempre incluye el nombre de la operación (figura 2-8).



Figura 2-8 Mensajes

2. Máquina de estado, es una relación que especifica las secuencias de estados de un objeto en su tiempo de vida en respuesta a eventos.

Gráficamente, un estado es representado como un rectángulo redondeado, usualmente incluye su nombre y sus subestados (figura 2-9).

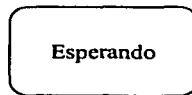


Figura 2-9 Estados

Estas dos cosas -interacción y máquinas de estado- son las cosas del comportamiento básico que se pueden incluir en un modelo UML.

Semánticamente estas cosas son conectadas usualmente a varias cosas estructurales, clases primarias, colaboraciones y objetos.

**\*) Cosas de Asociación**

Las cosas de asociación son las partes que organizan los modelos de UML.

Estas son cajas dentro de las cuales un modelo puede ser descompuesto.

Un paquete es un mecanismo de propósito general para organizar cosas dentro de grupos.

Cosas estructurales, cosas del comportamiento, y aun otras cosas de asociación pueden ser puestos en paquetes.

Los paquetes son las cosas básicas de agrupamiento con los que se puede organizar un modelo de UML. Además existen variaciones, tales como marcos, modelos, y subsistemas (tipos de paquetes).

Gráficamente, un paquete es representado como un fólder con pestaña, usualmente incluye sólo su nombre y algunas veces su contenido (figura 2-10).

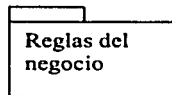


Figura 2-10 Paquetes

**\*) Cosas de anotación**

Las cosas de anotación son las partes que explican los modelos de UML. Son los comentarios que pueden aplicarse para describir y remarcar sobre cualquier elemento de un modelo.

Una nota es representada gráficamente como un rectángulo con una esquina doblada, junto con un texto o comentario gráfico (figura 2-11).

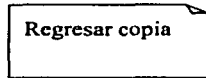


Figura 2-11 Notas

La nota es un elemento básico para anotaciones en modelos de UML.

#### \*) Relaciones en UML

Existen cuatro tipos de relaciones en UML y son los bloques de relación de construcción básicos en UML para la realización de modelos.

- Dependencia
- Asociación
- Generalización
- Realización

1. Una dependencia es una relación semántica entre dos cosas en las que un cambio a una cosa (cosa de dependencia), afecta a la cosa dependiente.

Gráficamente, una dependencia es representada como una línea punteada, posiblemente dirigida, y ocasionalmente incluye una etiqueta (figura 2-12).



Figura 2-12 Dependencias

2. Una asociación, es una relación estructural que describe un conjunto relación entre cosas. Agregación es un tipo especial de asociación, representando una relación estructural entre el todo y sus partes.

Gráficamente, una asociación es representada como una línea sólida, posiblemente dirigida, ocasionalmente incluye una etiqueta y con frecuencia contiene otros adornos, tales como el de multiplicidad y los nombres de rol (figura 2-13).

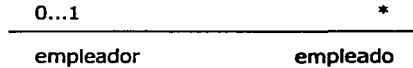


Figura 2-13 Asociaciones

3. Una **generalización** es una relación de generalización /especialización, en la cual objetos de los elementos de especialización son sustituibles por objetos de elementos generalizados. De esta forma la especialización comparte la estructura y el comportamiento de la general. Gráficamente una relación de generalización está representada como una línea sólida con una punta de flecha vacía apuntando hacia lo general (figura 2-14).



Figura 2-14 Generalizaciones

4. Una **realización** es una relación de semántica entre clasificadores, en donde un clasificador especifica un contrato que otro clasificador garantiza. Se encuentran relaciones de realización en dos lugares: entre las interfaces y clases o componentes que los realizan y entre casos de uso y colaboraciones que se realizan. Gráficamente, una relación de realización es representada como un cruce entre generalización y relaciones de dependencia (figura 2-15).

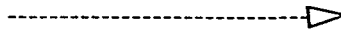


Figura 2-15 Realización

Estas cuatro cosas, son los elementos básicos de relación que se pueden incluir en el modelo de UML.

## 2.5. Diagramas en UML

Un diagrama es la representación gráfica de un conjunto de elementos frecuentemente representados como una grafica conectada de vértices (cosas) y arcos (relaciones).

Se dibujan diagramas para visualizar un sistema desde diferentes perspectivas, así un diagrama es una proyección dentro de un sistema.

UML tiene nueve clases de diagramas:

- Diagrama de Clases
- Diagrama de Objetos
- Diagrama de Casos de Uso
- Diagrama de Secuencia
- Diagrama de Colaboración
- Diagrama de Cartas de Estado
- Diagrama de Actividad
- Diagrama de Componentes
- Diagrama de Distribución

2.5.1. Un diagrama de clase muestra un conjunto de clases, interfaces y colaboraciones y sus relaciones. Los diagramas de clase manejan vistas estáticas de un sistema (figura 2-16).

Diagrama de Clases e interfaces

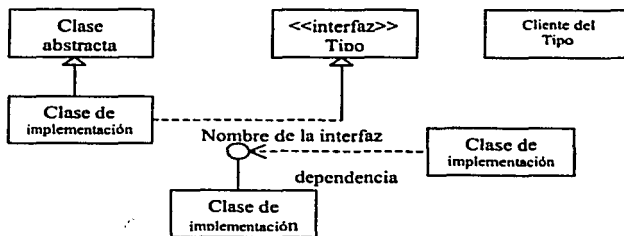


Figura 2-16 Diagrama de Clases

2.5.2. Un **diagrama de objetos** muestra un conjunto de objetos y sus relaciones. Los diagramas de objeto representan instantes estáticos de ejemplos de las cosas encontradas en los diagramas de clases (figura 2-17).

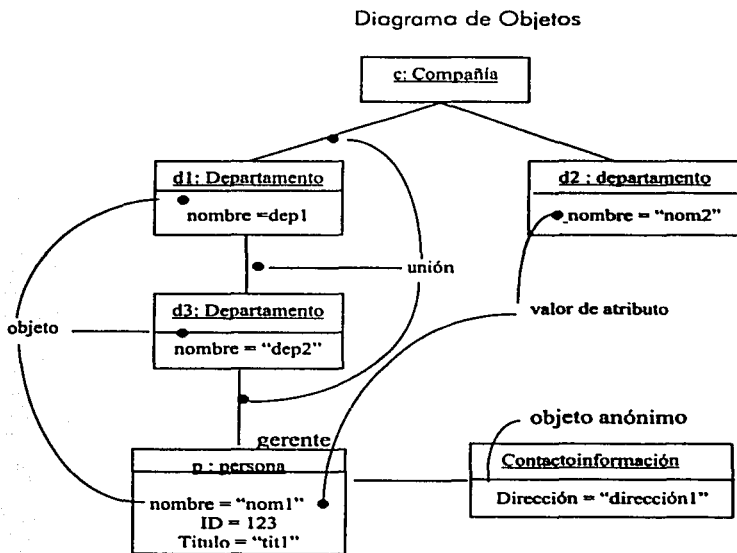


Figura 2-17 Diagrama de Objetos

2.5.3. Un **diagrama de casos de uso** muestra un conjunto de funcionalidades o casos de uso y sus actores y sus relaciones. El diagrama de casos de uso maneja una vista estática de casos de uso de un sistema. Estos diagramas son especialmente importantes en la organización y modelan el comportamiento del sistema (figura 2-18).

Diagrama de casos de uso

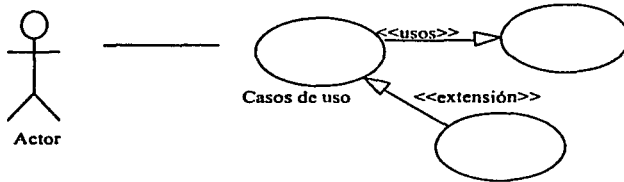


Figura 2-18 Diagrama de Casos de uso

**2.5.4. Los diagramas de secuencia y de colaboración** son diagramas de interacción de clases. Un diagrama de interacción muestra una interacción que consiste en un conjunto de objetos y sus relaciones. Los diagramas de interacción manejan las vistas dinámicas del sistema. Un diagrama de secuencia es un diagrama de interacción que enfatiza el tiempo y el orden del mensaje; un diagrama de colaboración es un diagrama de interacción que enfatiza la organización estructural de los objetos que envían y reciben mensajes (figura 2-19, figura 2-20).

Diagrama de secuencia

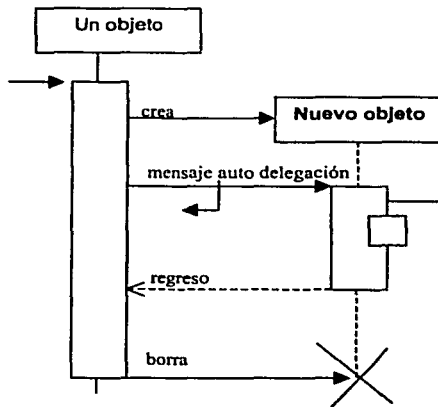


Figura 2-19 Diagrama de Secuencia



Diagrama de colaboración

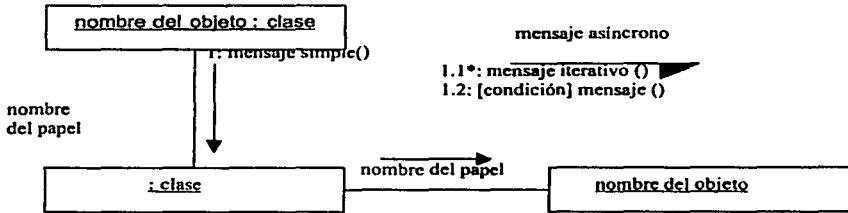


Figura 2-20 Diagrama de Colaboración

2.5.5. Un diagrama de estado muestra una máquina de estado que consiste en estados, transiciones, eventos, y actividades. Los diagramas de estado manejan las vistas dinámicas de un sistema (figura 2-21).

Diagrama de estados

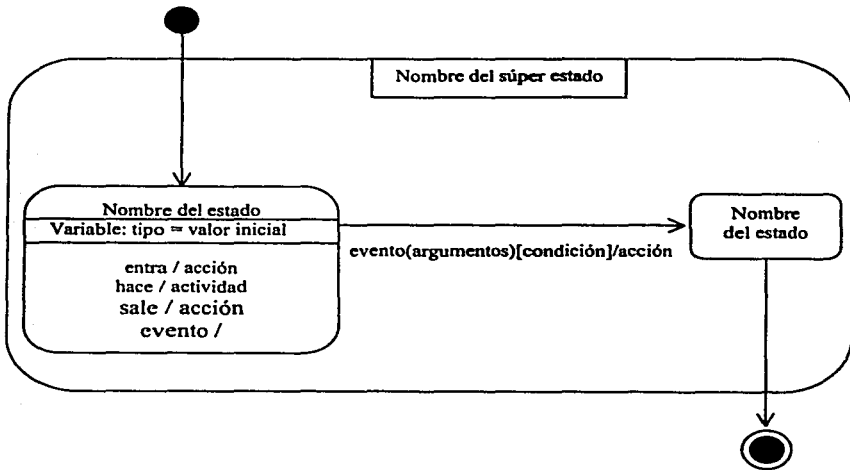


Figura 2-21 Diagrama de Estado

2.5.6. Un **diagrama de actividad** es una clase especial del diagrama de estado que muestra el flujo de las actividades dentro del sistema. Los diagramas de actividad manejan las vistas dinámicas del sistema (figura 2-22).

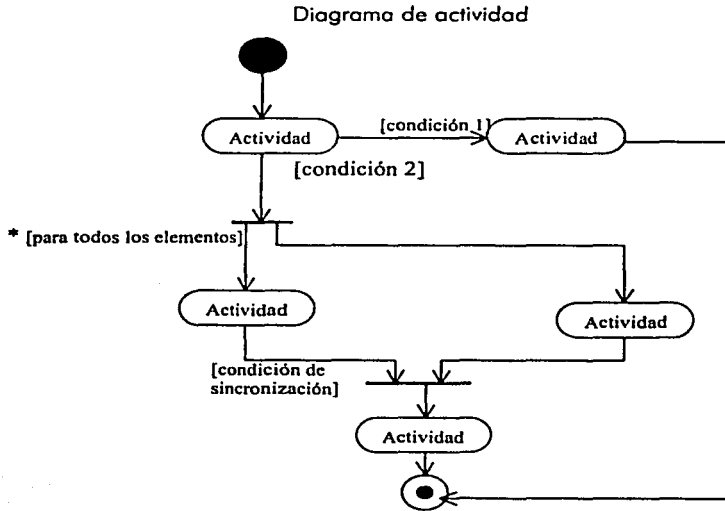


Figura 2-22 Diagrama de Actividad

2.5.7. Los **diagramas de componentes** muestran la organización y la dependencia entre un conjunto de componentes. Los diagramas de componentes manejan las vistas estáticas de implementación de un sistema (figura 2-23).

Diagrama de emplazamiento

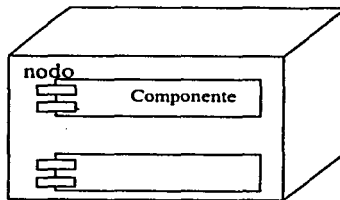


Figura 2-23 Diagrama de Componentes

**2.5.8. Los diagramas de instalación muestran la configuración de los nodos de proceso al momento de ejecución y los componentes que en ellos viven. Los diagramas de instalación manejan las vistas de instalación estáticas de una arquitectura (figura 2-24).**

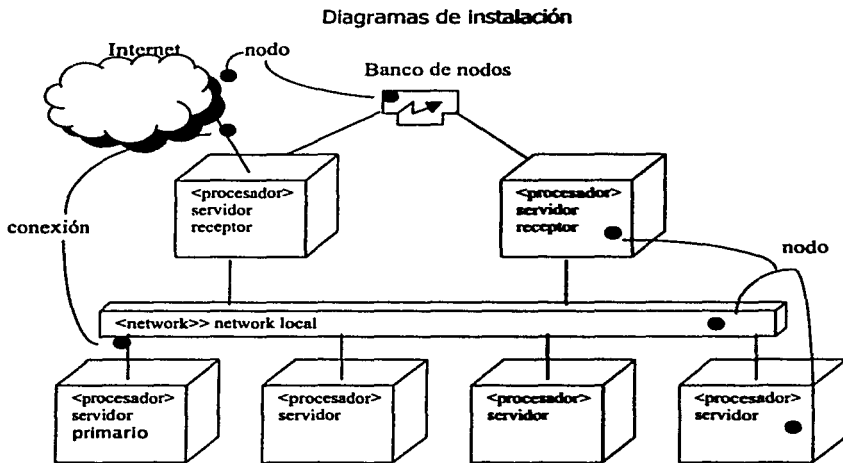


Figura 2-24 Diagrama de instalación

## 2.6. Arquitectura

Existen distintos puntos de vista entre las personas que intervienen en las etapas de desarrollo de un proyecto, por lo que la arquitectura de un sistema es posiblemente la parte más importante que puede utilizarse para manejar estos puntos de vista diferentes y así controlar el desarrollo a través del ciclo de vida de un sistema.

La arquitectura es un conjunto de decisiones significativas sobre la organización de un sistema de software.

## 2.7. Herramientas de apoyo al modelado

### 2.7.1. Evolución de las herramientas

Una parte importante de la evolución en el estudio y mejora de las herramientas del análisis estructurado como son las herramientas CASE (Ingeniería de Software Asistida por Computadora) surgió a mediados de los años sesentas (ver Fig. 2.25)

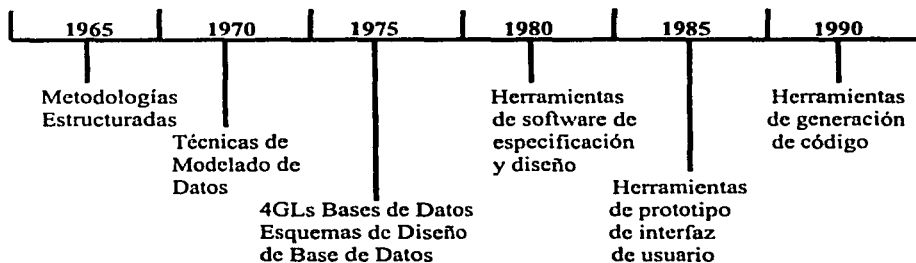


Figura 2.25 Evolución de las herramientas y metodologías CASE

### Lenguajes de cuarta generación de bases de datos

Son lenguajes de alto nivel las que proveen facilidades de acceso a las bases de datos. Estas son mucho más fáciles de usar que los lenguajes tradicionales usados por accesos a las bases de datos programados, como el COBOL y C. El objetivo es remover la carga del tedioso código de acceso a las bases de datos reemplazándolo con una cantidad pequeña de código escrito en un lenguaje de alto nivel de 4GL específicamente designado por acceso a bases de datos. Muchos 4GLs proveen formularios de descripción de archivos y de diseño usando editores de texto común y donde se muestra la primera tecnología apalancada en herramientas enfocadas en formularios de codificación.

### Herramientas de Software de Especificación y Diseño.

Generalmente usadas para el análisis y diseño estructurado fueron metodologías pioneras desarrolladas por Tom DeMarco y Edward Yourdon. Estas herramientas de diseño y especificaciones de propósito general pueden ser usadas para especificar y diseñar casi

cualquier pieza de software. Las herramientas de análisis y diseño usualmente implementan la creación de diagramas de flujo y técnicas de cartas estructuradas las que son excelentes para la representación gráfica del flujo entre los procesos computacionales.

Muchas herramientas de especificación y diseño tienen extensiones para demostrar iteraciones temporales comunes de controles del sistema en tiempo real.

### **Herramientas de prototipo de interfaces de usuario**

Las interfaces de usuario varían enormemente en estilo y en contenido. Algunas de ellas, como las máquinas parlantes automatizadas, están diseñadas para un uso fácil para usuarios no sofisticados, mientras que otras interfaces, como aquellas de proceso de palabras, son construidas para procesos de alto volumen.

El desarrollo de prototipos puede ser un largo y laborioso proceso y siempre es una fuerte tentación para usar los prototipos como una implementación final, mejor que un rediseño basado en la retroalimentación de los usuarios. Afortunadamente, las herramientas CASE para el diseño y generación de interfaces de usuarios están disponibles en el mercado.

Este tipo de prototipos rápidos ofrecen el apoyo necesario para mayor productividad y proyectos de software exitosos. Frecuentemente son capaces de desplegar pantallas que simulan la secuencia y son suficientes para abrir el canal de comunicación entre el usuario final y el diseñador de software. Si el diseñador de software tiene la capacidad de simular interfaces rápidamente (incluyendo las no funcionales) tiene un vehículo valioso de retroalimentación de la comunidad de usuarios finales.

#### **2.7.2. Herramientas CASE (Ingeniería de Software Asistida por Computadora),**

Las herramientas CASE de interfaces de usuario agregan valor al proceso de desarrollo de software durante las etapas de especificación de requerimientos, de análisis y diseño. Estas herramientas apalancan a los diseñadores reduciendo riesgos de implementación ey aumentan grandemente la aceptación del usuario final.

Herramientas de interfaces de usuario orientadas hacia un hardware específico (y algunas veces a los sistemas operativos), plataformas tales como IBM PC y ciertas microcomputadoras populares. Esta orientación es por necesidad. Las plataformas de hardware varían grandemente en términos del despliegue tecnológico. Una interfaz de

usuario aceptable en la IBM PC es muy diferente de las interfaces aceptables en una terminal de mainframe IBM 3278.

Por varios años paquetes de generación de formularios, tales como ISPF para mainframes de IBM y Panel para la IBM PC, han ayudado a los profesionales de diseño de software especializados en interfaces de llenado de formularios para una variedad de aplicaciones. Las interfaces de llenado de formularios son entre las más directas y ampliamente aceptadas de los estilos de interfaces de usuario disponibles. En un principio, muchos de estos paquetes empezaron como subrutinas de bibliotecas, no como las herramientas CASE. Sin embargo, se desarrollaron dentro de las herramientas CASE cuando sus vendedores agregaron editores de orientación gráfica que permiten el diseño de software en pantallas con salidas visibles que generan automáticamente el software para esas salidas.

La mayoría de las herramientas CASE implementaron el desarrollo de tecnologías de diseño estructurado en papel y lápiz durante los años 1960s y 1970s como se muestra en la figura 2.25. Estas metodologías se popularizaron en muchas tiendas comerciales de proceso de datos como una forma de manejar su acumulación de aplicaciones mediante la reducción del riesgo del desarrollo técnico.

Había muchas metodologías estructuradas de modelado y diseño, cada una con enfoques escasamente diferentes. Muchas de estas metodologías han sido popularizadas por implementaciones de herramientas CASE. Seleccionar la metodología correcta para un trabajo requería consideraciones cuidadosas.

Durante los años 1980s, estas metodologías de análisis y diseño, migaron a herramientas CASE como estaciones de trabajo gráficas y las computadoras personales fueron ampliamente disponibles.

Ni una sola de las herramientas o metodologías CASE podía realizar completamente los trabajos de diseño y especificaciones. Ciertamente varias metodologías complementarias fueron requeridas para manejar todas las facetas de trabajo de desarrollo de software de la estructura de diseño de datos a través de las especificaciones de las interfaces de usuario. Aunque nos dirigimos hacia metodologías complementarias combinadas en ambientes de herramientas integradas, la herramienta universal aun vive en el futuro.

### **2.7.3. Herramientas PSEE (Ambientes de Ingeniería de Software Centrados en el Proceso).**

Los entornos de Ingeniería de Software Centrados en el Proceso (PSEEs) ya son una realidad. Un número creciente de sistemas están mostrando y están siendo aplicados a procesos de producción de software real. Los productos y prototipos existentes están basados en una variedad de tecnologías y aproximaciones, tales como bases de datos y lenguajes orientados a objetos, notaciones orientadas a estados, lenguajes basados en regla, y lenguajes lógicos.

Una importante meta se ha alcanzado en el desarrollo de esta tecnología. Existen algunos intentos iniciales para utilizarlos en escenarios de la industria, y alguna retroalimentación ha sido provista. Por lo tanto, vale la pena evaluar las experiencias pasadas para recoger y organizar la retroalimentación y proveer un mapa tentativo para nuestras actividades de investigación y retroalimentación. Además, es esencial desarrollar líneas guía y métodos que soporten estas actividades de evaluación para facilitar la recopilación y organización de la información y su diseminación a los investigadores y practicantes.

#### **Tabla de evaluación**

La tabla 1 representa un comparativo de varios marcos de referencia de procesos de software organizada en tres partes principales.

- Tecnología de Lenguajes de Modelado de Procesos (PML)
- Arquitectura PSEEs (Entornos de Ingeniería de Software Centrados en el Proceso)
- Experiencias

Los PSEEs existentes están basados en una variedad de lenguajes de modelado de procesos (PMLs) que lo caracterizan. Podemos usar las siguientes categorías:

<b>Tecnología PML</b>	<b>Cobertura de Enfoque</b>
	<b>Paradigma(s) Lingüísticos</b>
	<b>Modelado de entidades de proceso:</b>
	Actividades
	Productos
	Herramientas
	Roles
	Organizaciones
	...
	<b>Modulación /composición /re-huso</b>
<b>Mecanismos para procesos de evolución y representación</b>	
<b>Cooperación y control de concurrencia</b>	
<b>Soporte metodológico</b>	
<b>Soporte de herramientas PML</b>	
<b>Arquitectura PSEE</b>	<b>Arquitectura de alto nivel</b>
	<b>Interacción con herramientas de producción y herramientas de facilidades de integración.</b>
	<b>Comportamiento del PSEE</b>
	<b>Integración de datos</b>
	<b>Administración de inconsistencias y sincronización del espacio de trabajo</b>
	<b>Distribución y heterogeneidad</b>
	<b>Soporte y medición</b>
	<b>Eficiencia de la implantación</b>
<b>Experiencias</b>	<b>Ejemplos de referencia</b>
	<b>Experimentos internos</b>
	<b>Proyectos industriales</b>

Tabla 1 Evaluación de la Tecnología de Lenguajes de Modelado de Procesos



(1) **cobertura de enfoque:** un PSEE y su PML pueden ser adoptados por una variedad de razones. Un PML puede ser usado para mejorar la comprensión del proceso y su documentación a través del modelado, análisis y simulación formal. Esto puede ayudar en las iniciativas de los procesos de implementación y reingeniería. Un modelo formal puede además ser representado para proveer soporte y dirección a los desarrolladores de software. Por lo tanto, un PSEE y su PML(s) puede ofrecer soporte a una o más fases del proceso del ciclo de vida (o meta proceso), que es, el conjunto de actividades que son perseguidas para mejorar un proceso de software. Podemos identificar las siguientes fases principales:

- **Especificación de requerimientos:** Esta fase apunta a definir y / o documentar los requerimientos para el proceso, tal como el desempeño esperado y sus objetivos generales.
- **Evaluación y problemas de provocación:** Esta fase se dirige a la (re)evaluación del desempeño del proceso y a identificar sus problemas, de acuerdo a los objetivos definidos.
- **(Re)diseño:** en esta fase, el proceso es (re)diseñado (o se le a aplicado la reingeniería) para manejar los requerimientos y necesidades provocadas en la fase previa y las arregla identificando las desventajas.
- **Implementación:** En esta fase, el proceso diseñado o modificado es puesto en operación, modificando modelos, procedimientos, organizaciones, pólizas, y posiblemente el soporte al proceso.
- **Monitoreo y recuperación de datos:** En esta fase, los datos cualitativos y (cuando sea posible) cuantitativos en el desempeño de los procesos son recuperados y analizados para soportar iniciativas futuras del desempeño (i.e., retroalimentación en las fases primarias).

(2) **Paradigma(s) lingüísticos** usados en el lenguaje (e.g., orientación-a-objetos, basados-en-reglas. orientación-a-estados).

(3) **Disponibilidad de construir entidades de proceso de modelado** específicos como productos, actividades, y roles.

- (4) Construcción lingüística para soportar la modulación, composición de especificación, configuración de modelos, y reúso.
- (5) Mecanismos para soportar representación de modelos de procesos (e.g., interpretación .vs. compilación .vs. aproximaciones mixtas, hilos sencillos .vs. ejecución concurrente), y evolución (e.g., facilidades reflexivas).
- (6) Construcción lingüística para modelar cooperación (entre agentes humanos) y control de concurrencia, lo distinguimos entre sincrónico (e.g., tele conferencia) y cooperación asincrónica (e.g., correo electrónico).
- (7) Soporte metodológico a las diferentes fases del proceso del ciclo de vida.
- (8) Herramientas de soporte para PML (e.g., editores, simuladores, impacto, analizadores, interpretes).

## Capítulo 3

### **3. Modelo de Operaciones del Negocio del Seguro de Vida (Nota Técnica)**

#### **3.1. Definición de la Nota Técnica**

Para efectuar la definición de productos o planes, las compañías de seguros se basan en el desarrollo previo de un estudio actuarial que comprende los aspectos relacionados con la mortalidad así como financieros en la definición de cada producto.

En la actualidad existen herramientas de desarrollo, simulación y modelado de productos. Dichas herramientas incluyen métodos, fórmulas, tablas financieras, tablas de mortalidad, cálculo de primas, reservas, etc.

A estos estudios se les denomina Nota Técnica, que cuando se ha generado, debe contar con los requerimientos y expectativas de la organización y cumplir con las normas establecidas por la CNSF (Comisión Nacional de Seguros y Fianzas).

Por otro lado, al ser autorizado el plan de seguros este tiene que ser soportado administrativamente por todas las áreas de la empresa.

La compañía debe contar con hardware, software y comunicaciones que soporten la administración del nuevo Plan y puedan dar el servicio a las áreas que componen la estructura de la empresa.

#### **3.2. Sistema Integral para la Administración del Modelo de Seguros**

Consideremos un sistema integral para la administración de las operaciones de una compañía de seguros, el cual debe contemplar los siguientes módulos:

- Módulo de Administración de la Nota Técnica
- Módulo de Expedición y Suscripción de Nuevos Negocios
- Módulo de Comisiones y Agentes
- Módulo de Servicio a Asegurados

- Referencias a tablas de tarifas y factores
- Módulo de clientes y agentes

#### Módulo de Administración de la Nota Técnica

Es el módulo más importante del sistema, en virtud de que es en donde se registran los productos o planes, incluyendo las tarifas, valores y tablas del sistema. El módulo de administración de la nota técnica es donde se reflejan las políticas de la empresa, por lo que se puede considerar como la columna vertebral del sistema.

En éste módulo se dictan prácticamente todas las condiciones del negocio.

#### Módulo de Expedición y Suscripción de Nuevos Negocios

Este módulo se puede utilizar para efectuar la generación de estudios o pólizas prospecto, se efectúa también la suscripción de nuevos negocios y la emisión de los mismos. Aquí se realiza la cotización de planes, de acuerdo al tipo (vida, dotal, temporal, anualidad, vida universal, vida universal variable, etc.), edad, sexo, suma asegurada y en muchos casos en las condiciones de salud del asegurado.

#### Módulo de Comisiones y Agentes

En este módulo se encuentran los contratos de comisiones de los agentes de acuerdo a los planes de seguro que él puede vender. Cada contrato refleja los cuadros con las distintas escalas de comisiones por plan y duración. También encontramos las distintas jerarquías según los grados de supervisión.

#### Módulo de Servicio a Asegurados

Este módulo permite a los agentes y asegurados enterarse del estado de sus pólizas, así como de la facturación y cobranza, poniendo conocer a solicitud del asegurado, la distinta información que va generando su póliza a través del tiempo.

Dentro de esa información sería útil conocer cuales son los valores garantizados de la póliza, si tiene derecho a préstamo y cual es el importe, si puede comprar un seguro saldado o un seguro prorrogado, el monto de los dividendos, el costo de las primas futuras.

También el cliente debe poder efectuar cambios a su póliza; estos cambios pueden ser simples, esto es que no afecten el recálculo de sus primas y reservas, o que si se afecten.

#### Referencias a tablas de tarifas y factores

En el módulo de administración de la nota técnica, los factores y las tarifas se configuran en tablas externas al archivo de descripción del plan. Después, al añadir la descripción de plan para un plan nuevo, deben definirse los nombres de las tablas que contienen los factores o tarifas necesarios. Con este método se pueden compartir tablas de factores y tarifas entre compañías y planes.

#### Módulo de clientes y agentes

Este módulo administra la base de datos de clientes y agentes, aquí se registra la información de los clientes o asegurados al emitir una póliza. También se da de alta el catálogo de agentes, se dan de alta en la base de datos de clientes y agentes (como si fuera un cliente); en esta base de datos se dan de alta otras personas, como serían los asegurados mancomunados y /o coasegurados, pagador de primas, contratante, etc.

En este módulo se registran los datos personales de los clientes y agentes, incluyendo direcciones personales y de cobro.

La Figura 3.1 ilustra la relación entre los diferentes módulos, como son: el módulo de expedición, el módulo de agentes, el módulo de clientes, el módulo de contratos, el módulo de archivos de soporte y valores y el módulo de clientes y agentes.

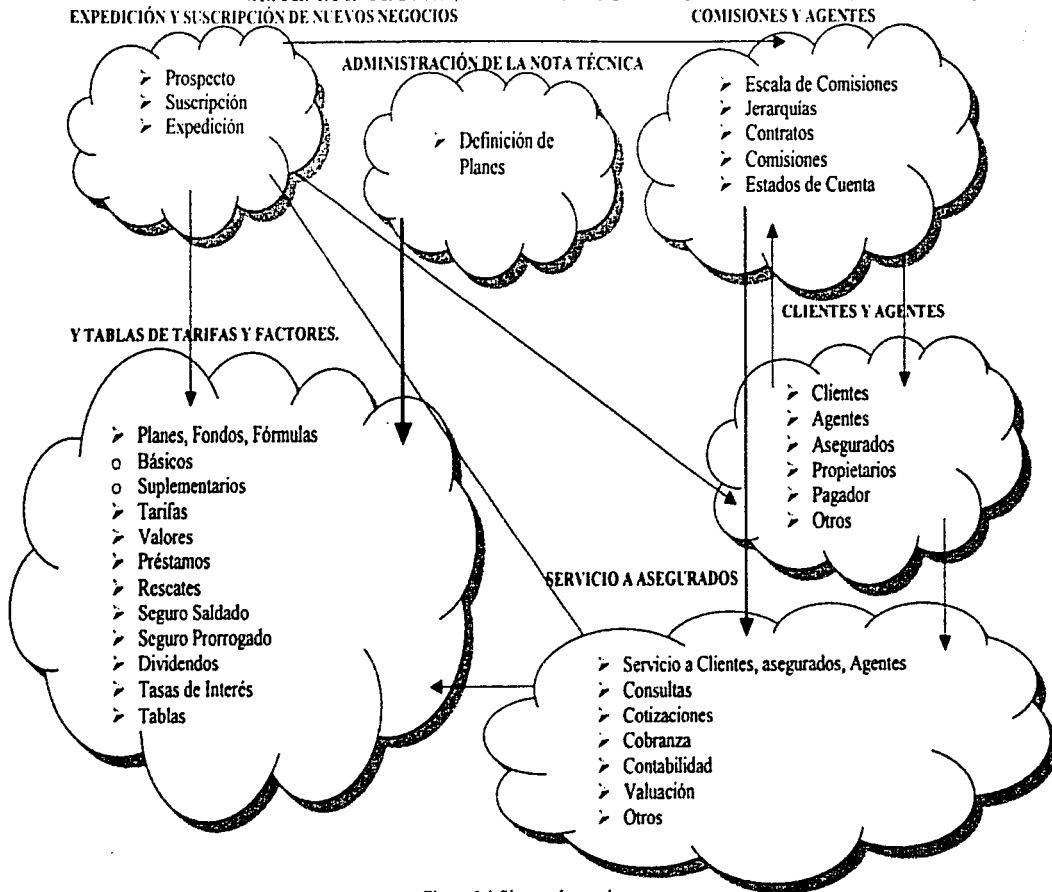


Figura 3.1 Sistema Integral

### 3.2.1. Descripción de la Información

El módulo de descripción del plan debe contener los sub-módulos que permiten describir las condiciones del negocio para las distintas áreas de la empresa.

Los módulos más importantes serían los siguientes:

- Plan básico
- Control de prima
- Comisión de prima
- Préstamo sobre póliza
- Valor en efectivo
- Valores garantizados
- Dividendos
- Reserva básica
- Plan no tradicional
- Siniestro
- Cuentas del libro mayor

#### **Plan básico.**

El módulo de descripción del plan básico establece las condiciones que describen las características del plan, siendo éstas las que regirán al producto durante toda la vida de las pólizas de seguros emitidas bajo las condiciones iniciales.

Las condiciones que se establecen en la definición del plan van desde cuáles serían las fechas de emisión (inicial y final) permitidas para la venta de este producto.

Aquí se establece el tipo de seguro, esto es, si el producto corresponde a una de las siguientes definiciones:

- Anualidad de prima fija
- Anualidad de prima flexible
- Dotal
- Enfermedad o Salud
- Vida entera con interés variable
- Ordinario de Vida

- Temporal
- Vida Universal
- Vida Universal Variable

Este módulo nos permite definir si el plan puede ser vendido como plan básico o como cláusula adicional o como ambos, o si el plan será vendido a prima única o no. También se establecen las edades (mínimas y máxima) a las que se podrá emitir el plan.

Los montos de seguro se deben manejar de acuerdo al número de unidades (mínimas y máximas) utilizando la condición del valor por unidad.

El módulo de descripción del plan básico debe contar con distintas tablas, las que permitirán las siguientes funciones:

- Estados de emisión permitidos
- Cláusulas adicionales válidas
  - exención de prima
  - beneficio por fallecimiento del pagador
  - beneficio por fallecimiento e incapacidad del pagador
  - beneficio por fallecimiento accidental
- Límites de suscripciones (capital asegurado y edad de emisión)
- Alturas / pesos de suscripciones (por edad y sexo)

Este módulo debe establecer si las tarifas se deben clasificar por sexo del asegurado y /o si es fumador o no fumador. Las tarifas pueden estar sobreprimadas o no.

Se pueden establecer los métodos de cálculo para la edad de emisión, para la fecha en cálculo de la edad y si es para un plan de vidas conjuntas o no, de acuerdo a las siguientes condiciones:

- Edad del último cumpleaños
- Edad del próximo cumpleaños
- Edad del cumpleaños más cercano



- Calcular edad desde la fecha de aplicación
- Calcular edad desde la fecha de emisión
- 58CSO (tabla de mortalidad para vidas conjuntas)
- 80CSO (tabla de mortalidad para vidas conjuntas)
- Edad de emisión del asegurado mayor
- Edad de emisión del primer asegurado

Establecer la opción de valores garantizados (NFO). Los valores garantizados son beneficios disponibles para el propietario de la póliza de seguro de vida que constituye valores en efectivo. Las pólizas con seguros de vida con coberturas temporales proporcionan sólo protección de seguro temporal y no constituyen valores en efectivo (los valores en efectivo se generan a partir del producto de la inversión de los valores de la póliza, como fondos, reservas, otros).

Si el asegurado deja de pagar sus primas a la renovación, entonces la póliza se cancela, sin embargo las pólizas de seguro con planes ordinario de vida proporcionan la protección del seguro y construyen valores en efectivo. Si el asegurado deja de pagar las primas de renovación el asegurado puede decidirse por una opción de valores garantizados con los valores en efectivo generados por su póliza; siendo el pago automático de las primas pendientes, comprar un seguro saldado reducido (de por vida pero con menor suma asegurada), o bien comprar un seguro prorrogado (por la misma suma asegurada pero por menos tiempo).

Existen planes con participación que generan dividendos o cupones, estos valores pueden ser cobrados en efectivo, o bien aplicarlos para la compra de seguros de adiciones saldadas.

### **Control de prima.**

Este módulo administra la información utilizada para determinar las tarifas de primas de todos los tipos de planes. ANEXO A.

Es necesario especificar las edades máximas a las que los asegurados deben pagar primas para mantener el beneficio vigente. Si la extensión del período de pago de prima no queda determinado por la edad del asegurado se debe especificar el número de años durante los

que deben pagarse las primas. Si el plan es de prima única el valor de este campo así lo debe indicar.

La fórmula de cálculo de prima indica cómo se calculan las primas anuales, con base en la tabla de tarifas de primas. Las tarifas sólo se determinan mediante factores estándar como, por ejemplo, edad, códigos de fumador, código de banda, etc.

El método de acceso a la tabla de tarifas de primas, puede ser por:

- Tarifas a edad alcanzada del asegurado
- Tarifas edad de emisión, o duración
- Tarifas unisexo
- Tarifas de fumador o no fumador
- Tarifas de primas si tiene bandas de volumen

Se debe incluir la fórmula de cálculo de sobreprima, si el plan admite tarifas sobreprimadas:

- 1 utilizar una cantidad fija extra y / o un porcentaje sobreprimado, pudiendo efectuarse en los subsistemas de nuevos negocios y servicios a asegurados.
- 2 utilizar la tabla de cargos por mortalidad actuales.

### **Comisión de prima**

Este módulo permite el manejo de la información utilizada para determinar la comisión que se debe pagar a un agente. Se define el tipo de producto del plan, éste se utiliza en los reportes de producción de los agentes, el tipo de producto de la comisión que representa el plan (es decir, producto seguro temporal, vida entera u otros).

Se especifica el tipo de producto del plan que hay en el reporte de producción del agente. El reporte de producción del agente agrupa la actividad por tipo de producto y tipo de producción. Definir si las cuotas son con comisión, para pagar comisiones sobre derechos de póliza y cobranza.

### **Préstamos sobre póliza**

Esta opción permite determinar cómo se administran los préstamos sobre póliza en el plan. Por ejemplo, se puede determinar si el interés del préstamo se calcula anticipado o con atraso (vencido) y si el plan ofrece tasas de interés distintas sobre los fondos cuyo préstamo se ha tomado del importe principal por contraposición a los fondos cuyo préstamo se ha tomado del interés.

Especificar el porcentaje del valor de la póliza que el cliente puede obtener como préstamo. Por ejemplo, 0.5. (50%), el porcentaje del préstamo debe ser mayor que cero y menor o igual que 1.00. (100%), si las tasas de interés cargadas por el préstamo se fijan al emitir la póliza indicar la tabla de tasa de interés.

### **Valor en efectivo**

Es necesario definir los métodos y tablas necesarios para administrar los valores en efectivo. La fórmula de cálculo, para determinar cómo debe calcularse la planificación de los valores en efectivo se puede elegir de las siguientes opciones:

- Utilizar los factores de valores en efectivo de la tabla de valores en efectivo
- Utilizar una fórmula basada en los valores en efectivo mínimos CSO 1958
- Utilizar una fórmula basada en los valores en efectivo mínimos CSO 1980
- Utilizar una fórmula basada en los valores en efectivo mínimos CSO 1958
- Utilizar una fórmula basada en los valores en efectivo mínimos CSO 1980 con un beneficio por fallecimiento de renovación mínimo.
- Utilizar fórmula definida por el usuario.

La tabla proporciona los factores de valores en efectivo por unidad, a los que se accede mediante la edad alcanzada o edad de emisión, el código de fumador, el sexo, la duración final de la póliza y la banda de prima.

Método de acceso, para determinar el método de acceso de la tabla de valores en efectivo, seleccionar una de las opciones siguientes:

- Edad alcanzada

- Edad de emisión, duración.

Tarifas unisexo o tarifas de fumador. Especificar si la tabla de valores en efectivo contiene factores independientes para varones y mujeres, o si la tabla de valores en efectivo contiene factores independientes para fumadores y no fumadores. O si la tabla de valores en efectivo tiene bandas de volumen.

### Valores garantizados

Este módulo permite seleccionar las opciones de valores garantizados disponibles para los clientes así como los métodos y las tablas necesarios para administrar las opciones.

Tablas necesarias ~ ANEXO A

Se deben crear las tablas siguientes, para el manejo de las distintas opciones de valores garantizados:

- Tabla de tasas de interés RPU (Seguro Saldado Reducido)
- Tabla de mortalidad RPU (Seguro Saldado Reducido)
- Tabla de tasas de interés ETI (Seguro Prorrogado)
- Tabla de mortalidad ETI. (Seguro Prorrogado)

Préstamo automático para prima. Si el plan ofrece el préstamo automático para prima como opción de valores garantizados se debe indicar en este módulo. El préstamo será por la cantidad de la prima vencida.

Rescate en efectivo. Si el plan ofrece el rescate en efectivo como opción de valores garantizados, así indicarlo.

Saldado reducido. Si el plan permite a los asegurados convertir el valor en efectivo en un seguro saldado reducido en lugar de caducar el beneficio.

Seguro prorrogado. Si el plan permite a los asegurados convertir el valor en efectivo en un seguro prorrogado, en lugar de caducar el beneficio.

Opción por omisión. Si los asegurados no seleccionan una opción de valores garantizados, elegir la opción que se les debe asignar:

- Préstamo automático para prima
- Rescate en efectivo
- Seguro saldado reducido
- Seguro prorrogado

Método de préstamo automático para prima. Indicar si el plan admite préstamos automáticos para primas (APL) como opción de valores garantizados,

#### **Dividendos**

Esta opción determina cómo se administran los dividendos del plan ( el dividendo es el producto de las ganancias de la inversión de la reserva, que se reparte entre los asegurados) y se puede elegir una de las siguientes opciones:

- Especificar qué opciones de dividendos están disponibles para los asegurados
- Establecer tarifas de dividendos y tasas de interés de acumulación de dividendos
- Especificar si los dividendos se incluyen en el cálculo de seguro prorrogado y saldado reducido.

Especificar si se permite pagar dividendos en adiciones saldadas, dividendos por las cantidades de adiciones saldadas adquiridas con dividendos, o bien, para pagar dividendos sólo por el capital asegurado del beneficio.

Definir si el plan permite acumular los dividendos con intereses, definir el nombre de la tabla de intereses de dividendos. Esta tabla proporciona la fecha de efecto y la tasa de interés correspondiente aplicada a los dividendos.

Acumulación utilizada para incrementar el valor nominal ETI. Para incluir los dividendos en acumulación en el valor nominal del seguro prorrogado al calcular el período de seguro prorrogado.

Rescatar adiciones saldadas para comprar RPU. Para incluir las adiciones saldadas compradas con dividendos en el valor neto de la póliza al calcular las cantidades de seguro saldado reducido. Adiciones saldadas utilizadas para incrementar el valor nominal ETI. Para incluir las adiciones saldadas compradas con dividendos en el valor nominal de seguro prorrogado al calcular el período del seguro prorrogado. Pagado al contado. Si el plan

admite que los dividendos se paguen al contado, o para reducir prima. Compra de adiciones saldadas. Si los dividendos se pueden utilizar para comprar un seguro saldado adicional.

#### **Reserva básica**

La reserva de un plan de seguros se constituye del excedente entre el monto de las primas pagadas y la cantidad neta en riesgo. Indicar en qué grupo debe incluirse el plan para efectos de la declaración anual, generando totales por clase de reserva con base en los siguientes grupos:

- Vida
- Anualidad
- Enfermedad
- Incapacidad

**Función de conmutación:** indica el tipo de fórmula utilizado para calcular los valores en efectivo y las reservas:

- Acortado
- Continuo.

**Tabla de mortalidad.** Nombre de la tabla de mortalidad utilizada para calcular los valores en efectivo. Si las reservas se calculan utilizando una tabla de mortalidad, la tabla que se especifique se utilizará también en el proceso de reservas estatutarias, tributarias y GAAP. Las reservas GAAP (Principios Contables Generalmente Aceptados) se calculan de acuerdo con la normativa que establece el organismo American Institute of Certified Public Accountants (Instituto Americano de Contadores con Certificado Público).

**Reservas estatutarias**

El proceso de reservas estatutarias cumple los requisitos estatales sobre reservas según estipula la Comisión Nacional de Seguros y Fianzas (CNSF).

Cálculo de reservas CARVM (Método de Valuación de Reservas de Anualidades de Comisarios de Seguros). En los productos de anualidades de prima flexible, inclusive las anualidades diferidas de prima única, también se calculan las reservas mínimas de acuerdo con la normativa CARVM. Tablas necesarias **ANEXO A**

Método de reservas. Utilizar una de las opciones siguientes para indicar cómo deben clasificarse los reportes de reservas:

- CRVM
- CRVM (planes de vida universal)
- CRVM (planes de vida entera con interés variable)
- Método de valuación especial
- CARVM (anualidades de prima flexible, inclusive las diferidas de prima única)
- Método de nivel neto
- Método estándar de Illinois
- Método estándar de New Jersey
- Definido por el usuario.

El método de cálculo de las reservas:

- Igual al 50% de la prima bruta (sólo productos de prima nivelada)
- Igual al 100% de la prima bruta (sólo productos de prima nivelada)
- Método de cálculo de media
- Método de media-terminal
- Método de terminal interpolado
- Método de reserva media, pero no inferior al 50% de la prima neta
- La cantidad mayor del valor de rescate en efectivo y del método de cálculo de media
- Igual al valor en efectivo

- Igual al valor de rescate en efectivo
- No calcular reservas. Calcular vencido /diferido /anticipado.
- No calcular reservas. Calcular vencido /no devengado /anticipado.
- CARVM (año de emisión)
- No calcular reservas ni vencido /diferido / no devengado / anticipado.
- Definido por usuario.

### **Siniestros**

Este módulo permite mantener la información del proceso de siniestros por fallecimiento. Normalmente, las compañías exigen pruebas de que el asegurado es un enfermo terminal para procesar un siniestro de reducción de prestación en vida.

Para indicar la base del cálculo se debe definir si el valor del beneficio es a la fecha de la reclamación o si el valor del beneficio debe ser proyectado a la fecha del siguiente aniversario de la póliza.

### **Cuentas del libro mayor**

Este módulo contiene los números de cuentas del libro mayor usadas en el sistema mostrando las cuentas del libro mayor necesarias en el plan nuevo.

- Prima del primer año.
- Prima de renovación.
- Comisión del primer año
- Comisión de renovación
- Principal del préstamo
- Interés del préstamo
- Cargo administrativo por rescate
- Rescate parcial para pagar prima
- Beneficio de rescate
- Cargos por rescate
- Vencimiento /dotal
- Intereses recibidos al vencimiento
- Siniestro de prestación en vida del primer año



- **Siniestro de prestación en vida de renovación**
- **Siniestro por fallecimiento del primer año**
- **Siniestro por fallecimiento de renovación**
- **Interés recibido por siniestro por fallecimiento**
- **Siniestro por fallecimiento ADB del primer año**
- **Siniestro por fallecimiento ADB de renovación**
- **Exención de prima de primer año**
- **Exención de prima de renovación**

Estas cuentas son utilizadas por el sistema en el transcurso de la operación del negocio, pudiendo utilizar un programa de interfase que convierta las cuentas utilizadas por el sistema al catálogo de cuentas utilizado por la compañía y requerido por la autoridad CNSF (Comisión Nacional de Seguros y Fianzas).

## Capítulo 4

### 4. Modelo de Datos en UML, del Negocio del Seguro de Vida

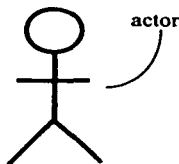
#### 4.1. Determinar los Actores

##### Definición

*actor* es un conjunto coherente de roles que juegan los usuarios de los casos de uso cuando interactúan con éstos.

El proceso de generación e instalación de un nuevo producto de seguros en el área técnica de vida, en una compañía de seguros, comprende el siguiente flujo de actividades en la jerarquía de la compañía:

1. La dirección general de la empresa solicita al área técnica el desarrollo de un nuevo producto (plan de seguros), con determinadas condiciones, tanto técnicas como de mercado; de acuerdo a los productos que en ese momento tengan en el mercado las compañías de seguros de la competencia o en algunos casos estén por sacar al mercado (figura 4.1).



Director General

Figura 4.1 Actor, Dirección General

2. La dirección técnica, por su parte, inicia las actividades para el diseño del nuevo plan de seguros generando la nota técnica, la cual deberá cumplir con las expectativas de la dirección general; y cumplir con los aspectos financieros, de mercadeo y legales. También deberán cubrir los aspectos correspondientes a los incentivos de los agentes y por último, el nuevo producto debe ser soportado por la infraestructura de la compañía, tanto de las áreas operativas como de la de sistemas, la figura 4.2, representa el actor denominado Dirección Técnica.

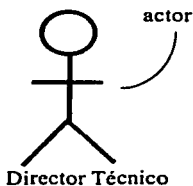


Figura 4.2 Actor, Dirección Técnica

3. Una vez aprobado el nuevo producto por la dirección general, la dirección técnica envía la Nota Técnica a la Comisión Nacional de Seguros y Fianzas (CNSF), para su registro y aprobación, por lo que la figura 4.3, representará el actor CNSF.

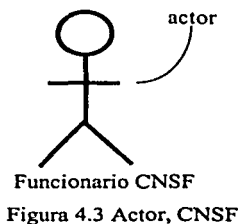


Figura 4.3 Actor, CNSF

Una vez recibida la nota técnica del nuevo plan de seguros por las autoridades de la CNSF, estas tendrán un plazo de 30 días para aceptar, corregir o rechazar el nuevo plan de seguros. Sin embargo la CNSF pueden tomar más tiempo del mencionado, por lo que la compañía de seguros puede iniciar la operación con el nuevo producto una vez transcurridos los 30 días, si es que no tiene alguna respuesta por parte de las autoridades.

Pero al recibir la compañía de seguros la respuesta de la CNSF, ésta debe realizar los ajustes al producto (y a la operación, de ser necesario) señalados por las autoridades correspondientes sin importar las pólizas ya emitidas.

4. La Dirección Técnica debe reunirse (solicitar) con el área de sistemas (figura 4.4) con el fin de darle a conocer los requerimientos del nuevo producto, con el fin de que ésta efectúe los cambios

y adaptaciones que se requieran a los distintos sistemas (o al sistema integral), con el fin de que el sistema soporte la operación de las distintas áreas de la empresa.

Así mismo, el área técnica se debe encargar del diseño del producto dentro del sistema, y el llenado de todos los factores y tablas correspondientes; como son los factores de tarifas, recargos, tasas de interés, tablas de mortalidad, factores de reservas, rescates, etc., etc.



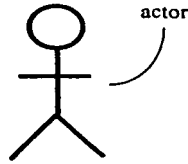
Figura 4.4 Actor, área de sistemas

5. El área técnica, debe explicar al área de expedición, las condiciones del nuevo plan, para que esta área aprenda el manejo operativo del nuevo producto, estableciendo los cambios y condiciones del nuevo producto, ver figura 4.5.



Figura 4.5 Actor, Funcionario de expedición

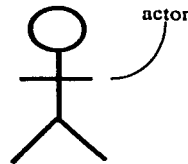
6. El área de servicio a asegurados debe conocer las características del nuevo producto para poder efectuar las operaciones del negocio que correspondan al nuevo plan, figura 4.6.



Funcionario servicio a asegurados

Figura 4.6 Actor, funcionario de Servicio a Asegurados

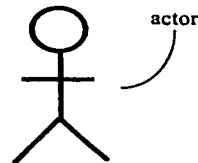
7. El personal de Agentes debe conocer las nuevas escalas de comisiones que pagará el nuevo plan, y si esto implica efectuar cambios a los procedimientos existentes, figura 4.7 .



Funcionario Agentes

Figura 4.7 Actor, Funcionario de Agentes.

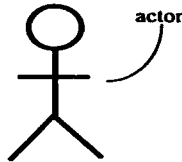
8. El área de contabilidad debe conocer si hay afectaciones a las cuentas contables, ya sea por cambios de las autoridades o por alguna razón generada por el nuevo producto figura 4.8.



Funcionario Contabilidad

Figura 4.8 Actor, Funcionario de Contabilidad

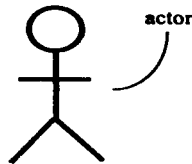
9. El área de cobranzas se encarga de la facturación y de la cobranza de los recibos generados por el sistema y que corresponden al pago de las primas de seguros, figura 4.9.



Funcionario Cobranzas

Figura 4.9 Actor, Funcionario de Cobranzas

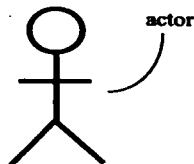
10. El área de actuaría depende directamente de la dirección técnica y es quien se encarga de la definición de los nuevos productos en el sistema, así como de la preparación de los reportes para la CNSF, figura 4.10.



Funcionario Actuaría

Figura 4.10 Actor, Funcionario de Actuaría

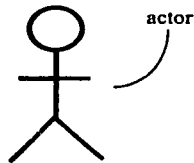
11. El cliente es la persona que inicia el proceso del negocio en una compañía de seguros al decidir la compra de una póliza de seguros, figura 4.11.



Funcionario Cliente

Figura 4.11 Actor, Funcionario de Contabilidad

12. El agente es la persona que realiza la venta del seguro ante el cliente, y es un representante de la compañía de seguros ante él, figura 4.12.



Funcionario Agente

Figura 4.12 Actor, Funcionario de Agente

#### 4.2. Determinar los Diagramas de Casos de Uso

##### Definición

*Los casos de uso* son la descripción de un conjunto de secuencias de acciones, incluyendo variantes, que ejecutan un sistema para producir un resultado observable de valor para un actor.

Los diagramas de casos de uso se emplean para modelar la vista de las funcionalidades de un sistema. La mayoría de las veces esto implica modelar el contexto del sistema.

Los diagramas de casos de usos son importantes para visualizar, especificar y documentar el comportamiento del sistema. Estos diagramas facilitan que los sistemas y subsistemas sean abordables y comprensibles al presentar una vista externa de cómo pueden utilizarse estos elementos en un contexto dado. Los diagramas de casos de uso también son importantes para probar sistemas ejecutables a través de ingeniería directa y para comprender sistemas ejecutables a través de ingeniería inversa.

1. El caso de uso de la Dirección General solicita el desarrollo de un nuevo plan de seguros a la dirección técnica, quien lo desarrolla y solicita el registro a la CNSF y su autorización corrección o rechazo de la nota técnica (figura 4.13).

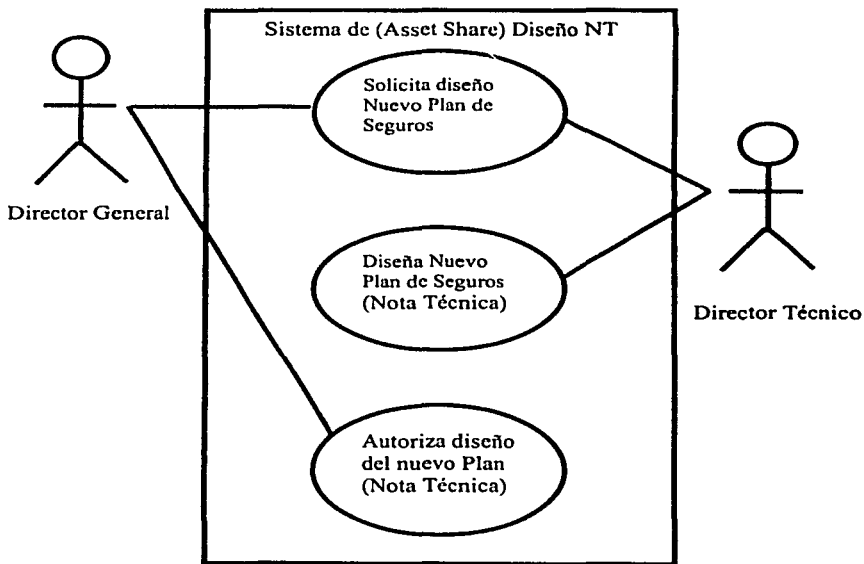


figura 4.13 Solicitud de diseño de nuevo plan



Una vez autorizada la nota técnica del nuevo plan por la dirección general, el área técnica lo envía para su registro y autorización ante la CNSF, figura 4.14.

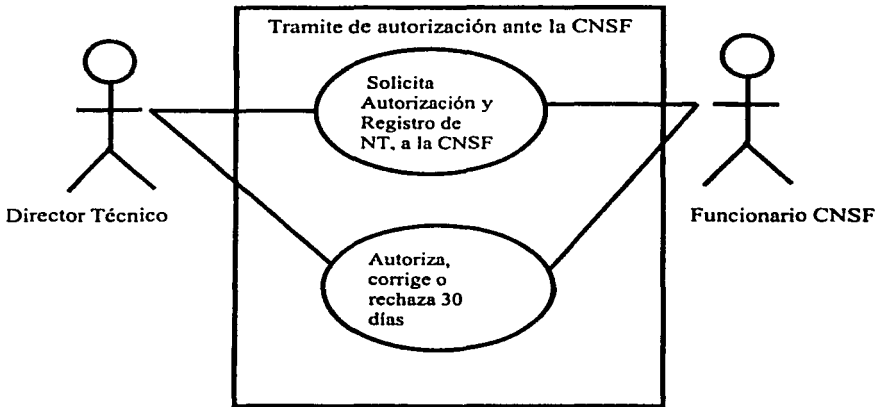


figura 4.14 Tramite de autorización y registro de la Nota Técnica ante la CNSF

2. Habiendo recibido la autorización del nuevo producto por parte de la CNSF y de la dirección general, la dirección técnica crea en el sistema integral el nuevo plan, solicitando al área de sistemas el apoyo para que si es necesario se efectúen las adecuaciones correspondientes (figura 4.15).

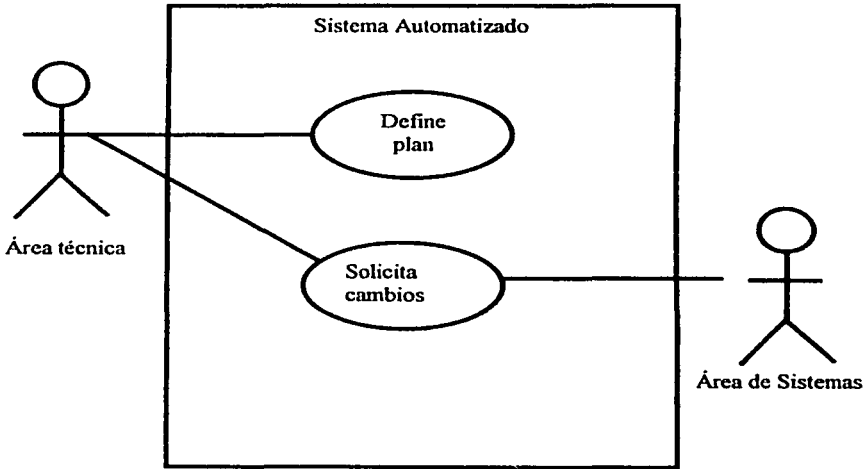


figura 4.15 Definición del plan en el sistema integral

Así mismo él área técnica es responsable de definir en el sistema tanto la información del plan, como las tarifas y valores correspondientes de acuerdo a la figura 4.16.

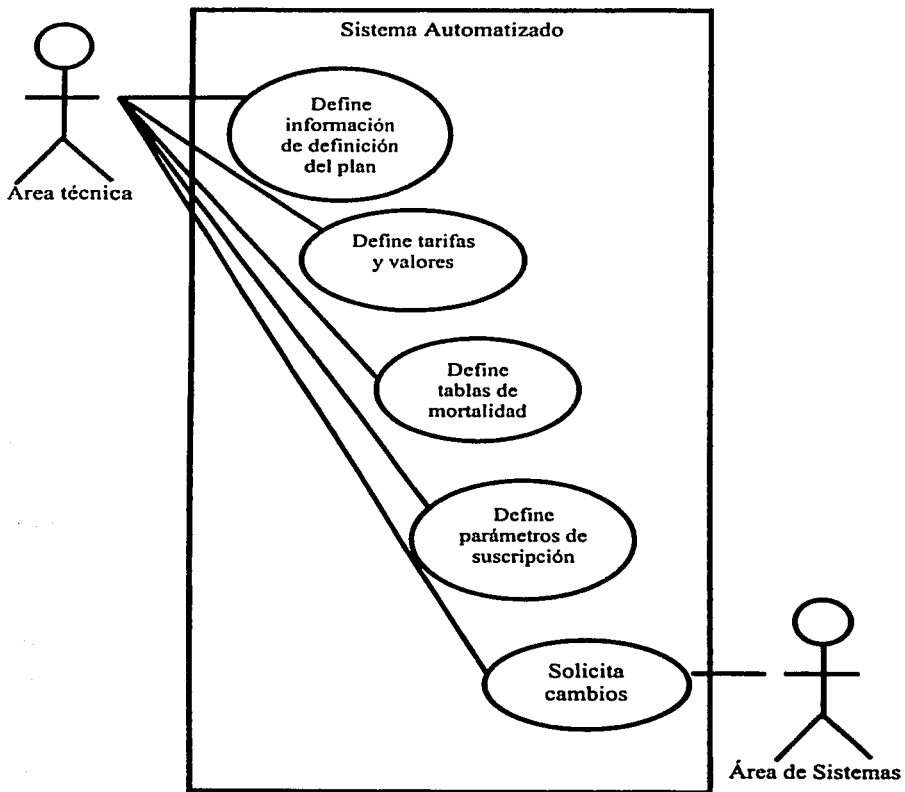


figura 4.16 Definición de información del plan en el sistema integral

También es necesario que el área técnica cargue en la definición del producto las opciones de Seguro Saldado y Seguro Prorrogado, ver figura 4.17.

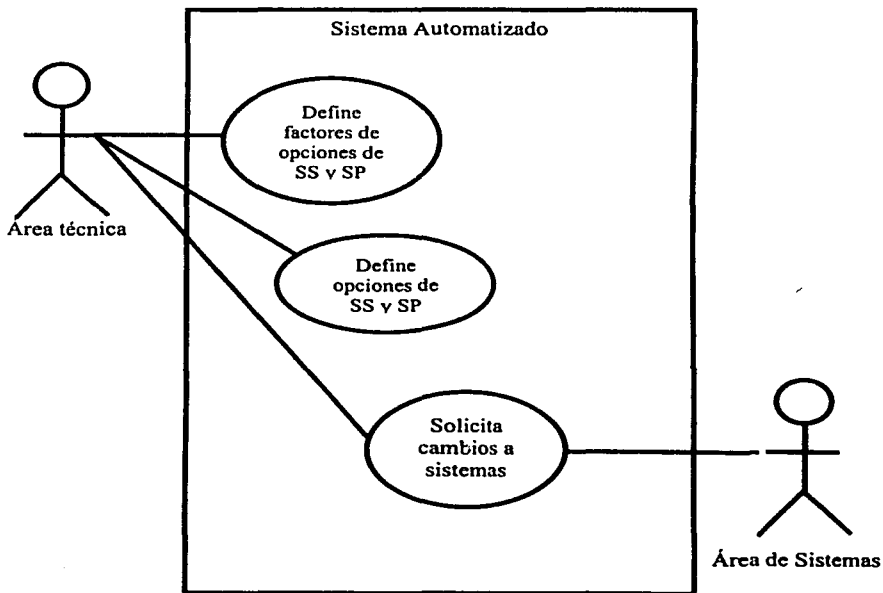


figura 4.17 Define opciones del plan en sistema integral

El área técnica define las opciones de prestamos dividendos y rescates del plan, así como los factores correspondientes figura 4.18, y figura 4.19.

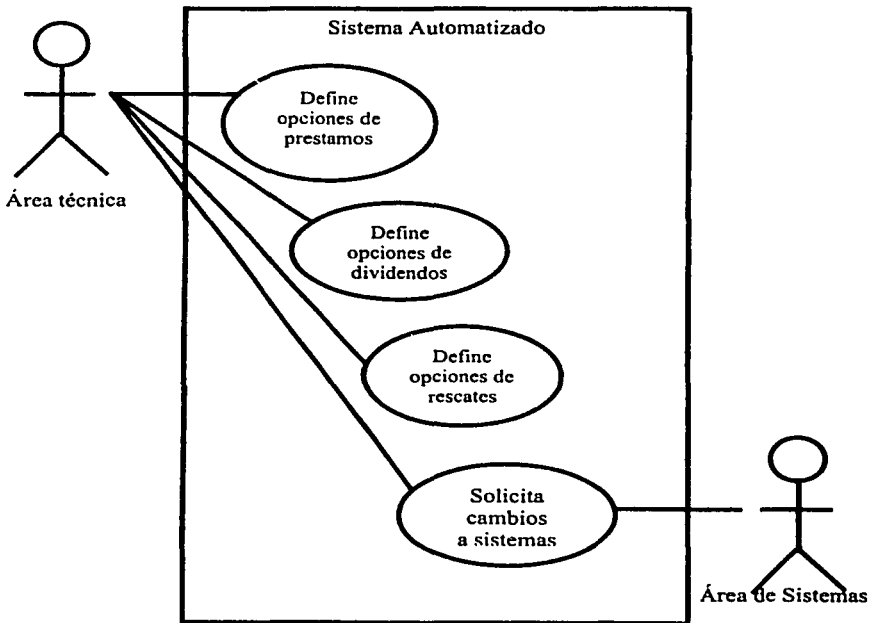


figura 4.18 definición de opciones de valores garantizados

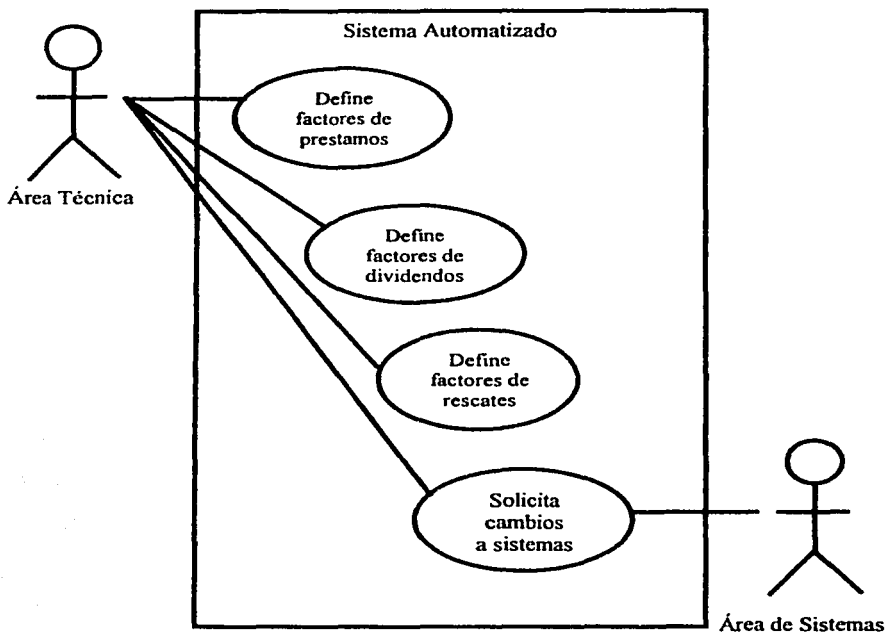


figura 4.19 Definición de factores de valores

La figura 4.20 muestra la función del área de sistemas, quien se encarga de diseñar, construir y administrar tanto el propio sistema como las bases de datos que contienen la información definida por las distintas áreas de la empresa.

En el área de sistemas se realizan algunas veces, los procesos de producción correspondientes

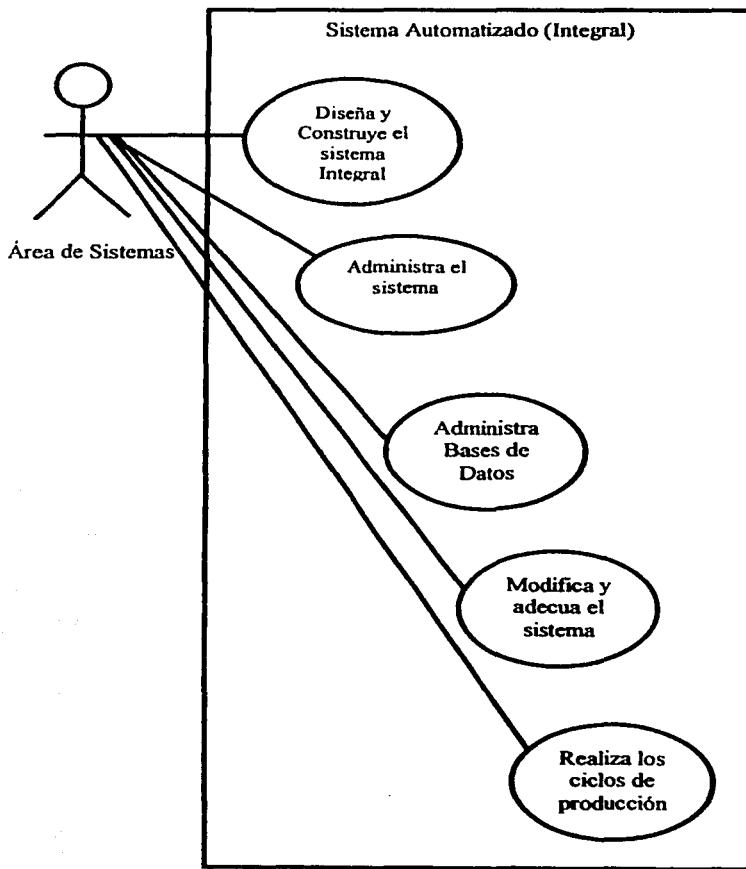


figura 4.20 Sistema Integral

### 4.3. Determinar los Diagramas de Actividades

#### Definición

Los diagramas de actividades se utilizan para modelar los aspectos dinámicos de un sistema. La mayoría de las veces, esto implica modelar los pasos secuenciales (y posiblemente concurrentes) de un proceso computacional. Con un diagrama de actividades también se puede modelar el flujo de un objeto conforme pasa de estado a estado en diferentes puntos del flujo de control. Los diagramas de actividades pueden utilizarse para visualizar, especificar, construir y documentar la dinámica de una sociedad de objetos, o pueden emplearse para modelar el flujo de control de una operación. Los diagramas de actividades destacan el flujo de control entre actividades.

Una actividad es una ejecución no atómica en curso, dentro de una máquina de estados. Las actividades producen alguna acción, compuesta de computaciones atómicas ejecutables que producen un cambio en el estado del sistema o el retorno de un valor.

Los diagramas de actividades, no son sólo importantes para modelar los aspectos dinámicos de un sistema, sino también para construir sistemas ejecutables a través de ingeniería directa e inversa.

En la figura 4.21 se muestra el diagrama de actividades para la definición del plan. La dirección general solicita al área técnica el diseño de un nuevo plan de seguros, el área técnica lo diseña y lo envía a la CNSF para su aprobación y registro.

Una vez que la CNSF, registró y autorizó el nuevo plan de seguros, el área técnica define el plan en el sistema.

El área de sistemas habilita el sistema para que el nuevo plan sea definido, y a su vez debe participar para efectuar las modificaciones que se requieran.



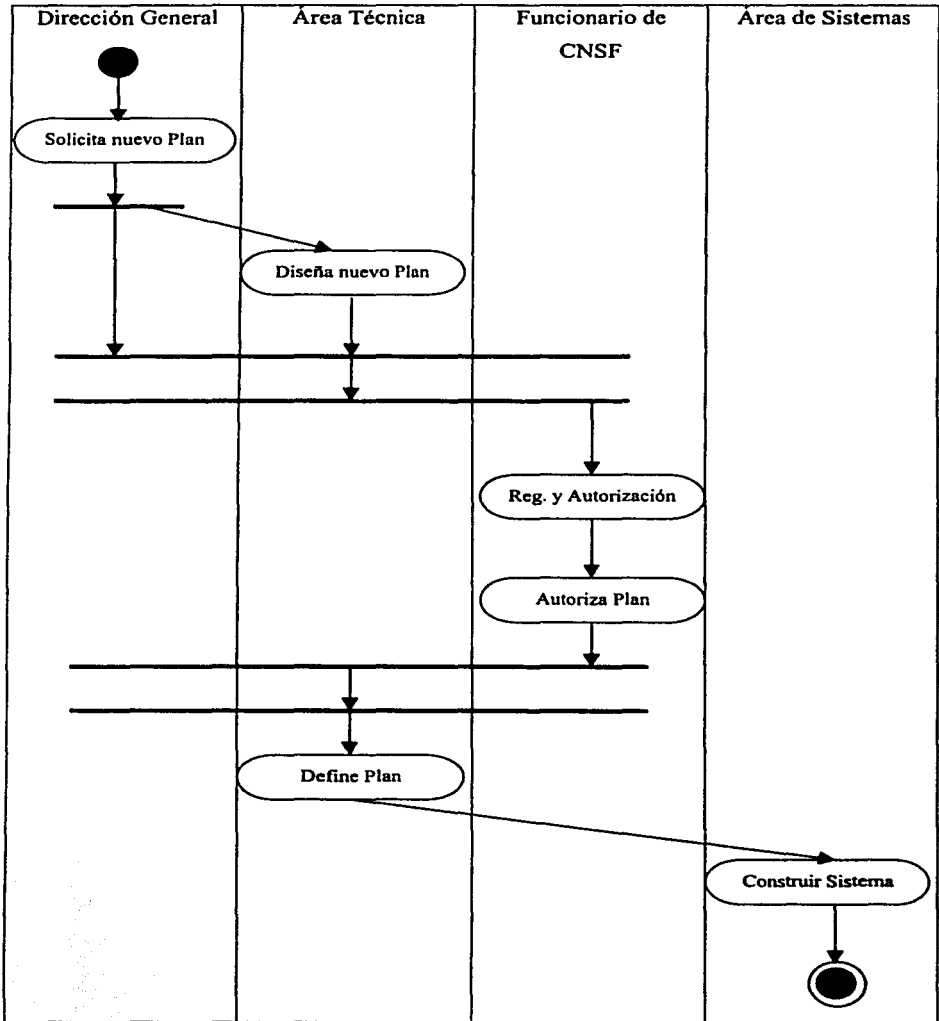


Figura 4.21 Definición del plan

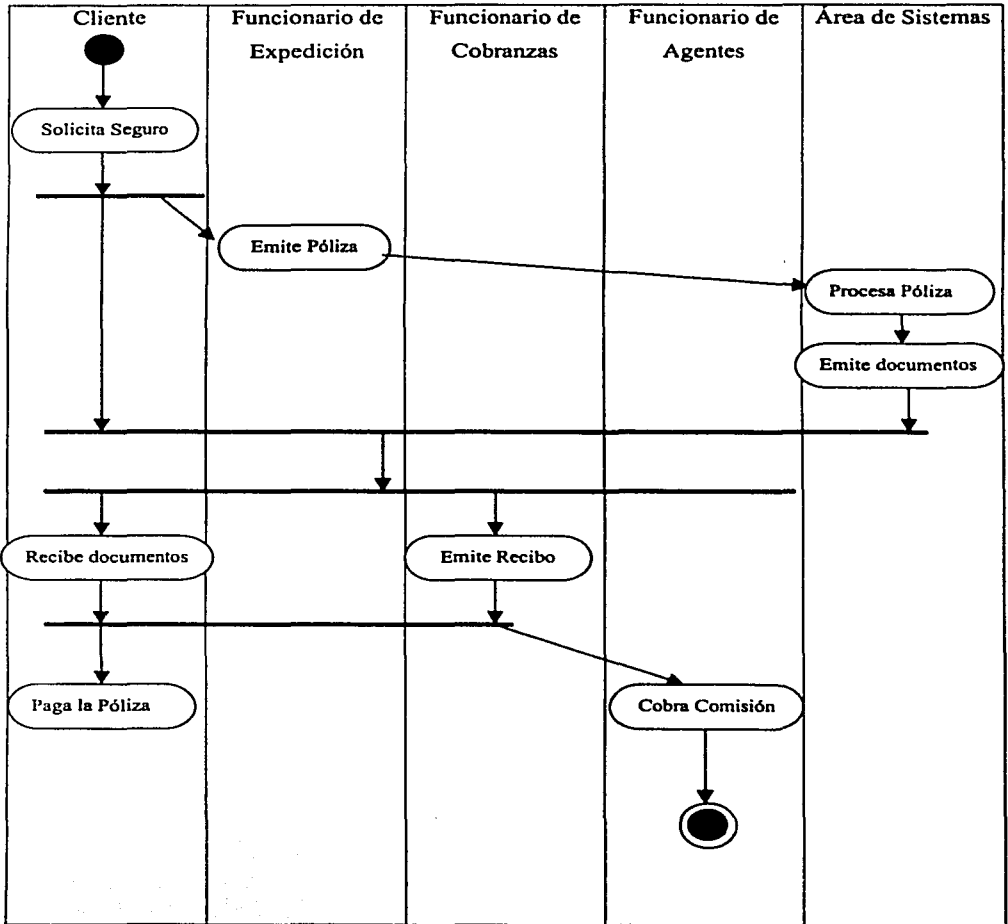


Figura 4.22 Expedición de la póliza

La figura 4.22, describe el modelo del proceso de expedición de la póliza; el cliente solicita la emisión de una nueva póliza de seguros, el departamento de emisión emite la póliza a través de sistemas, ahí se generan los documentos correspondientes, entregando al cliente tanto la póliza como el recibo de cobro, al pagar el cliente la prima correspondiente el área de cobranzas registra el pago y el agente puede efectuar el cobro de sus comisiones.

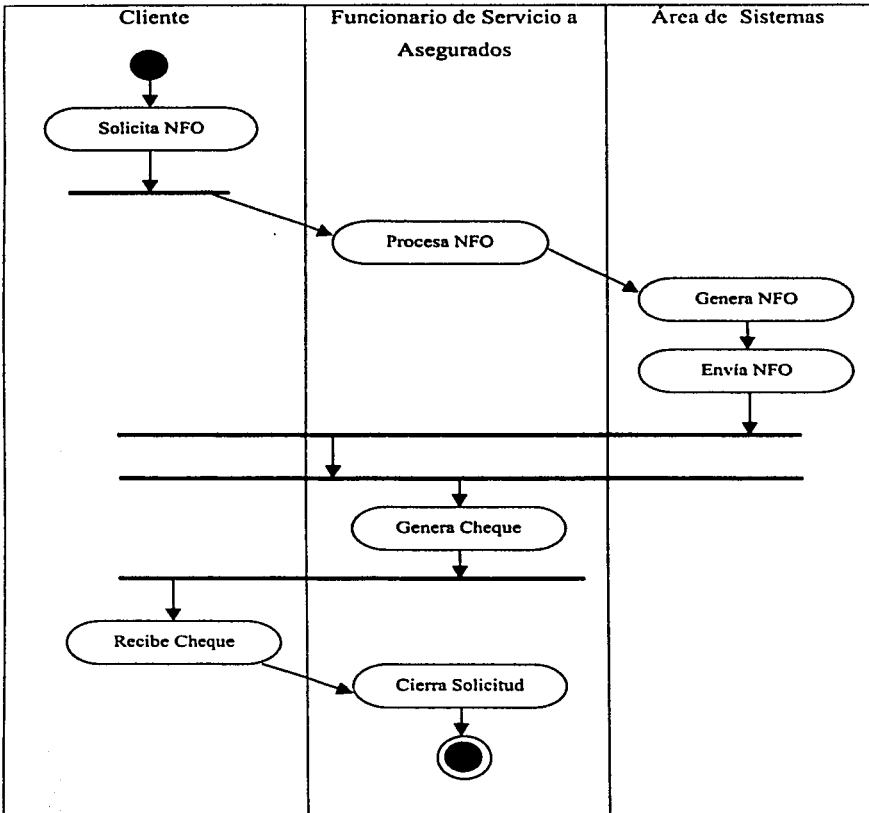


Figura 4.23 Modelo de solicitud de Valores Garantizados (NFO)

El modelo de la figura 4.23 nos describe el proceso de solicitud de valores garantizados por parte del cliente sobre los valores generados por su póliza.

El tramite lo realiza ante el área de servicio a asegurados, quien a su vez utilizando el sistema integral solicita el cálculo y proceso de los valores garantizados de la póliza del cliente. Quien puede solicitar: un préstamo, rescate de dividendos o de sus valores en efectivo.

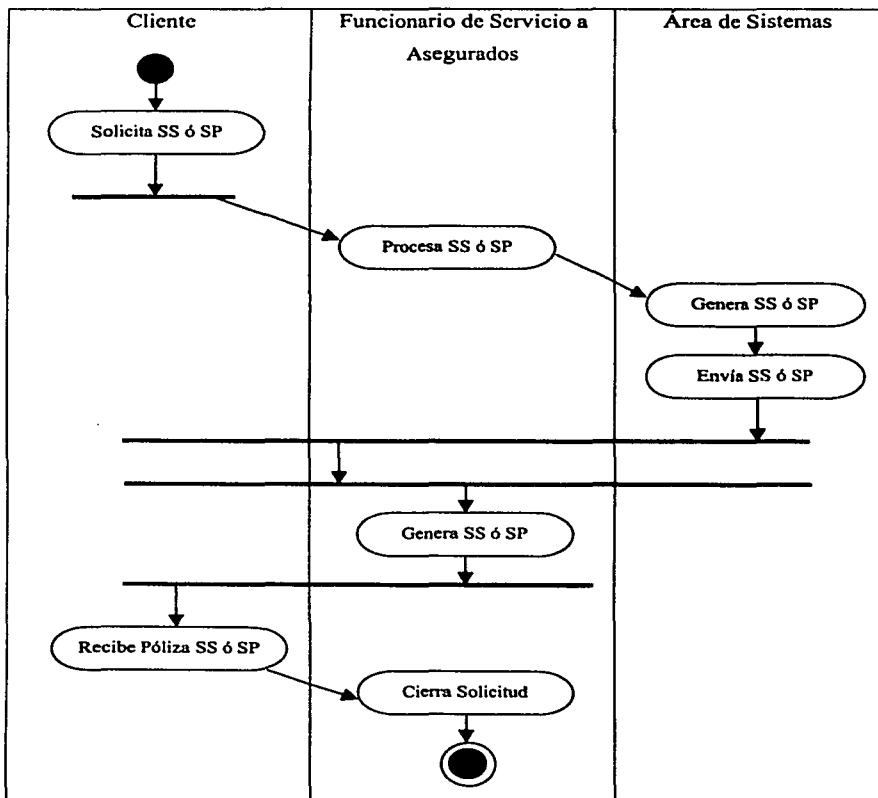


Figura 4.24 Modelo de solicitud de Valores Garantizados SS ó SP

El modelo de la figura 4.24 nos describe el proceso de solicitud de valores garantizados por parte del cliente sobre los valores generados por su póliza.

El tramite lo realiza ante el área de servicio a asegurados, quien a su vez utilizando el sistema integral solicita el cálculo y proceso de los valores garantizados de la póliza del cliente. Quien puede solicitar: con base en sus valores garantizados, rescatarlos y comprar o convertir su póliza a Seguro Saldado (reducido) ó a Seguro Prorrogado.

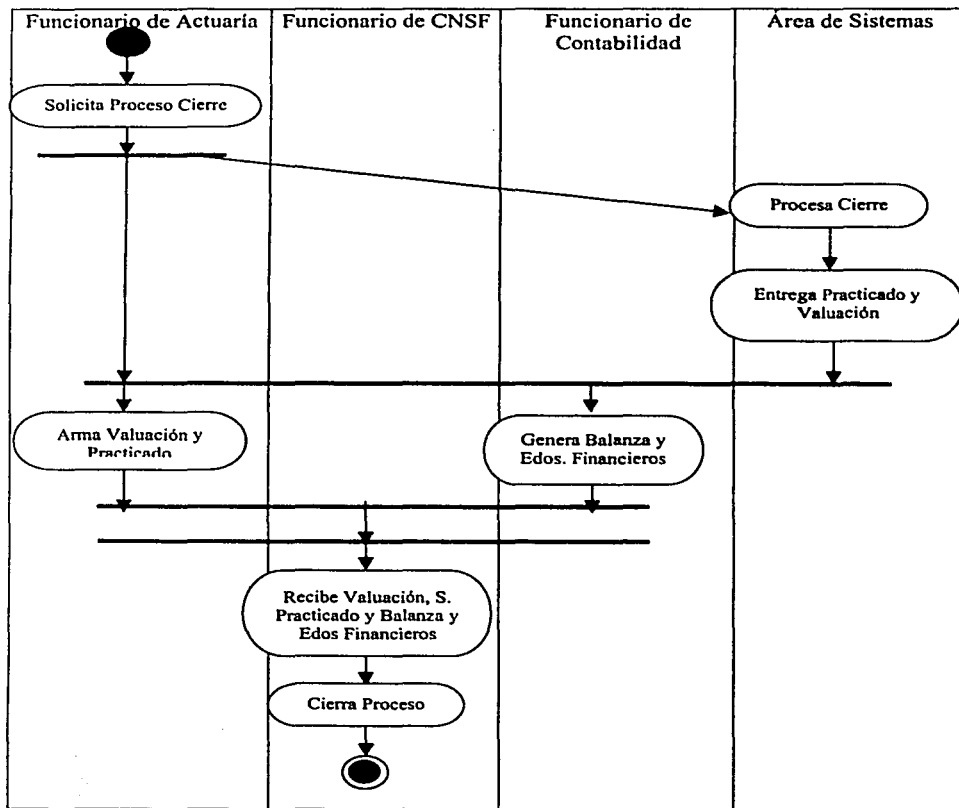


Figura 4.25 Proceso de Cierre Mensual

El diagrama de la figura 4.25 describe el modelo del proceso de cierre de mes, en el que el área de actuaría solicita a sistemas realice los procesos correspondientes para que tanto ellos como el área contable puedan generar los informes oficiales que la CNSF solicita mensual o trimestralmente.

Los informes que se entregan a la CNSF por parte de actuaría, son la valuación de reservas de planes y el seguro practicado.

Los informes que se entregan a la CNSF por parte de contabilidad, son los estados financieros y la balanza de comprobación consolidada, entre otros.

#### **4.4. Determinar los Diagramas de Clases o Estados**

##### **Definición**

Los diagramas de clases son más utilizados en el modelado de sistemas orientados a objetos. Un diagrama de clases muestra un conjunto de clases, interfases y colaboraciones, así como sus relaciones.

Los diagramas de clases se utilizan para modelar la vista de diseño estática de un sistema. Principalmente, esto incluye modelar el vocabulario del sistema, modelar las colaboraciones o modelar esquemas.

Los diagramas de clases son importantes no sólo para visualizar, especificar y documentar modelos estructurales, sino también para construir sistemas ejecutables, aplicando ingeniería directa e inversa.

La compañía de seguros se compone de varios departamentos en los que se lleva a cabo la operación de la empresa, ver figura 4.26.

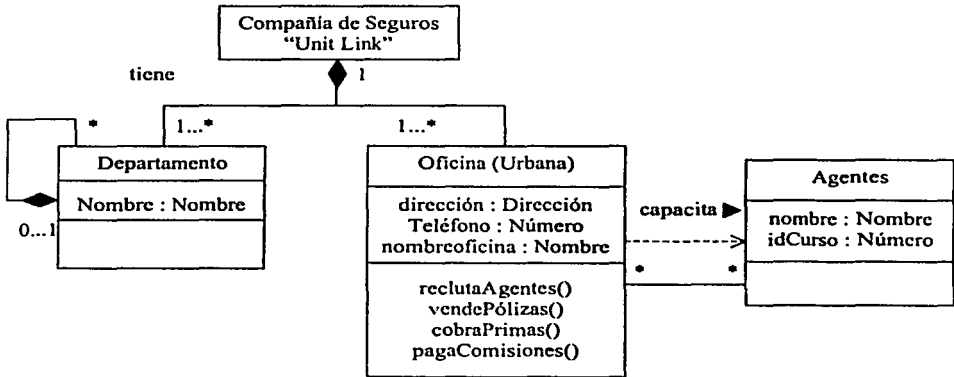


figura 4.26. Compañía de Seguros "Unit Link"

Los departamentos que hay en una compañía de seguros, serían:

- Área Técnica
- Expedición
- Cobranzas
- Agentes
- Contabilidad
- Sistemas

El área técnica se avoca a definir nuevos planes de seguros, definir los planes en el sistema definiendo todos los parámetros correspondientes, definir todos los factores del plan como son tarifas, reservas, tablas de mortalidad, valores de rescate, dividendos, etc. Ver figura 4.27.

<b>Departamento</b>
<b>Nombre : técnica</b>
<b>Responsabilidades</b>
--diseñarProductos
--definirPlan
--definirFactores
--elaborarSP
--elaborarValuación

figura 4.27. Definición de planes

El departamento de expedición se encarga de emitir pólizas a los asegurados efectuando un proceso de suscripción, cotizar primas de acuerdo al plan de seguros, aplicar extra primas en caso de algún padecimiento o sobrepeso , etc., emitir y generar documentos ver figura 4.28.

<b>Departamento</b>
<b>Nombre : expedición</b>
<b>Responsabilidades</b>
--suscribirAsegurado
--tarificarPlan
--extraPrimar
--emitirPóliza

figura 4.28. Expedición de pólizas

El departamento de servicios a asegurados efectúa a solicitud del asegurado o el agente, la cotización de prestamos, cotiza los valores de rescate que haya generado la póliza, realiza el cálculo de dividendos de acuerdo al plan, también hace el cálculo del seguro saldado reducido o el seguro prorrogado, etc, ver figura 4.29.



<b>Departamento</b>
Nombre : servicioasegurados
Responsabilidades
--cotizarPréstamo
--cotizarValoresRescate
--cotizarDividendos
--cotizarSeguroSaldadoRed
--cotizarSeguroProrrogado
--etc

figura 4.29. Cotizar préstamo

El área de cobranzas efectúa el proceso de facturación y recolección de la cobranza, así como de la aplicación de los pagos recibidos en el sistema, y del seguimiento de la cobranza apoyado en el sistema integral, ver figura 4.30.

<b>Departamento</b>
Nombre : cobranzas
Responsabilidades
--emitirRecibos
--cobrarRecibos
--aplicarPagos
--seguimientoCobranza

figura 4.30. Facturación y Cobranzas

El departamento de agentes tiene entre sus funciones es la de reclutar agentes noveles y capacitarlos; también debe definir los contratos de los agentes, definir alimentar los cuadros de comisiones por contrato y plan, establecer las jerarquías de la estructura de los agentes, y supervisores y efectuar los pagos de comisiones, ver figura 4.31.

<b>Departamento</b>
<b>Nombre : agentes</b>
<b>Responsabilidades</b>
--reclutarAgentes
--capacitarAgentes
--establecerContratos
--definirEscalaComisiones
--definirJerarquiaAgentes
--pagoComisiones

figura 4.31 Área de agentes

El departamento de sistemas tiene dentro de su responsabilidad el desarrollo de sistemas, modificaciones y adaptaciones, capacitación, custodia y respaldo de los datos, ejecución de procesos, etc., figura 4.32.

<b>Departamento</b>
<b>Nombre : sistemas</b>
<b>Responsabilidades</b>
--crearSistemas
--administrarSistema
--custodiaDatos
--respaldoDatos
--ejecutarProcesos

figura 4.32 Construir y administrar sistemas.

La figura 4.33, nos muestra el modelo del proceso del negocio de una compañía de seguros entre las distintas áreas que la integran.

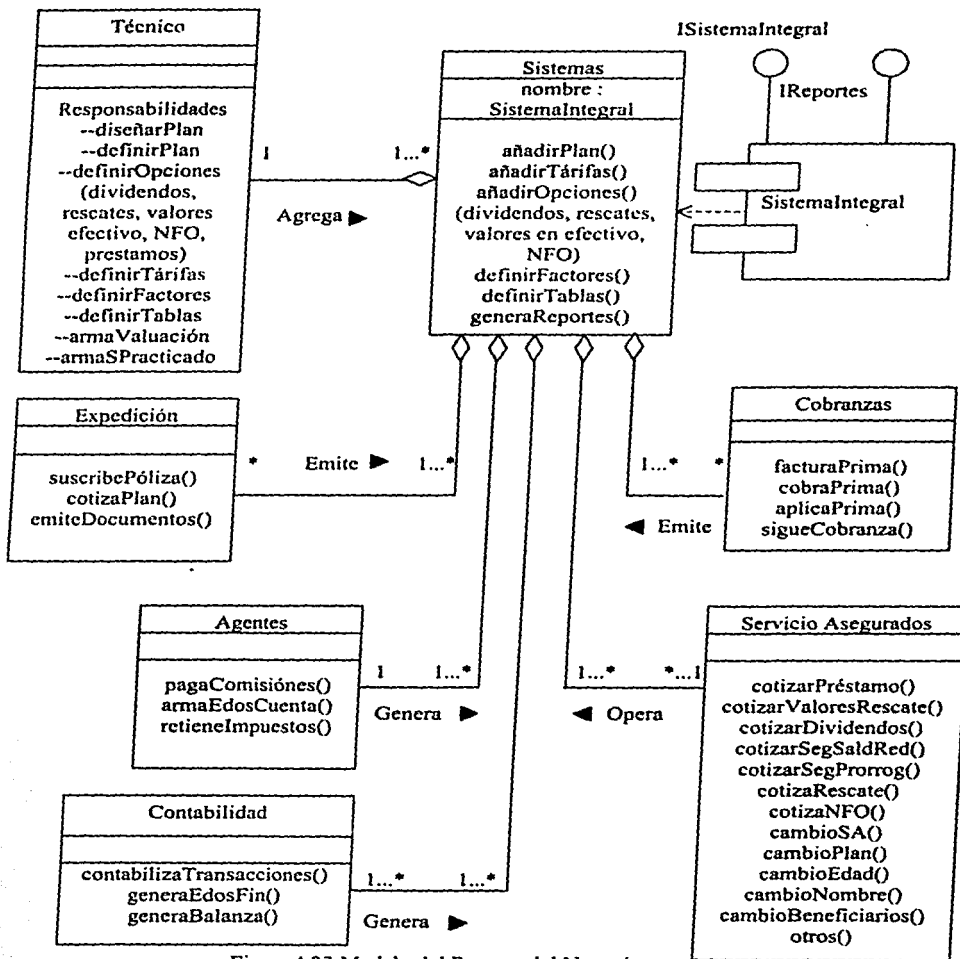


Figura 4.33 Modelo del Proceso del Negocio

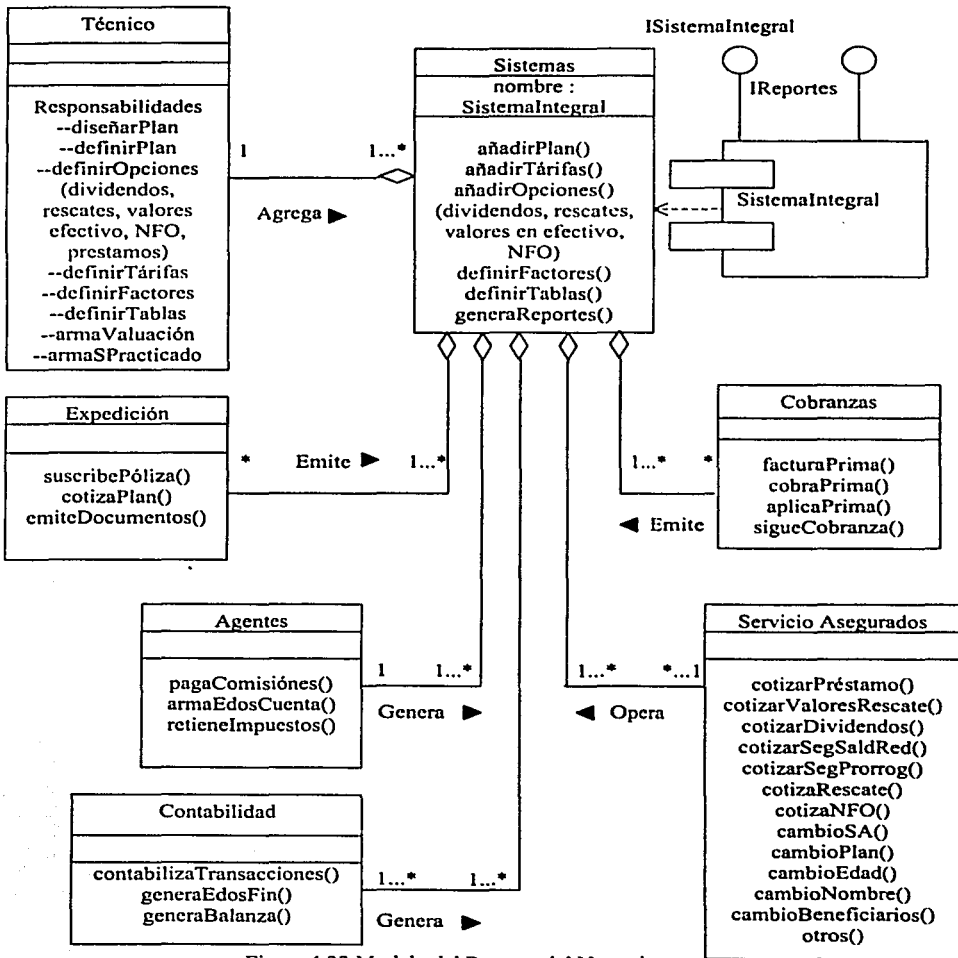


Figura 4.33 Modelo del Proceso del Negocio

Como se observa en la figura 4.33 encontramos los procesos de desarrollo y las responsabilidades dentro de la compañía de seguros por parte de cada una de las áreas que la integran.

El proceso del negocio inicia cuando el área técnica crea un nuevo producto, el cual debe definir en el sistema integral, diseñado y desarrollado por el área de sistemas, quien debe también administrar tanto el sistema como las bases de datos generadas por la propia operación del negocio y alimentadas por las diferentes áreas durante la vida de las pólizas, de acuerdo a la definición que efectuó el área técnica al elaborar y definir los distintos productos.

El diseñar estos modelos, nos permite mostrar los procesos de operación del negocio de una compañía de seguros, visualizando las interrelaciones que existen entre las áreas involucradas en cada etapa.

La figura 4.33 representa el flujo de la información desde el inicio del proceso del negocio, determinando la responsabilidad que existe en cada una de las áreas que participan en este proceso. Además nos permite identificar la interrelación entre las áreas operativas así como la estructura del sistema.

## Conclusiones

El modelado de datos se da a través de las herramientas y sus características estas nos permitieron construir los modelos aplicados al área de seguros.

Por otro lado, se presentó el desarrollo del Lenguaje Unificado de Modelado (UML), que es la unión de otros lenguajes de modelado basados en la tecnología orientada a objetos, habiendo sido este aprobado y adoptado por la OMG (Object Management Group, o bien el Grupo de Administración de Objetos). En virtud de ser un lenguaje de modelado muy completo el cual puede ser aplicado en las diversas ramas de la industria.

UML permite la visualización de los sistemas; permite también especificar la estructura y el comportamiento del sistema; en cuanto a la construcción, el modelado nos proporciona patrones de guía en la construcción de los sistemas; y el modelado documenta las decisiones que se tomen.

Se planteó la descripción del modelo de un sistema integral de seguros en el que se definen las actividades y responsabilidades de las áreas que intervienen en el proceso del negocio en una compañía de seguros.

Dicho sistema debe permitir que el usuario del área técnica defina y determine tanto las condiciones de nuevos planes de seguros como de las políticas de la compañía y las condiciones del mercado y la legislación del mismo.

Dentro de las condiciones del modelo del sistema integral de seguros, tenemos que el área técnica es la responsable de definir las condiciones del plan y el área de servicios a asegurados, que es el área responsable de la administración de la vida de la póliza, es quien debe cotizar y efectuar las transacciones correspondientes del manejo de los valores del plan, así como de las opciones de valores garantizados.

---

Por último, se propuso un modelo de operación del seguro de vida aplicado UML para modelado de datos, el cual nos permitió entender el flujo de cada etapa de la operación del negocio del seguro de vida (en este caso), utilizando herramientas que nos dan una metodología que nos permite visualizar, especificar, construir y documentar cada fase del sistema.

En este modelo se identifica que el proceso del negocio inicia cuando el área técnica crea un nuevo producto, el cual se debe definir en el sistema integral diseñado y desarrollado por el área de sistemas, quien debe también administrar tanto el sistema como las bases de datos generadas por la propia operación del negocio y alimentadas por las diferentes áreas.

## Anexo A

### Cálculos del valor de efectivo

NOTA: Métodos A y B definidos por la tabla de mortalidad

Función	Derivación	Redondeo
$\partial$	$\partial = \ln(1+i)$	$(9) = \ln(\#)$
$\ln(1+i)$	$\ln(1+i) = \sum_{m=1}^{15} \frac{i^m}{n} \bullet - j^{n+1}$	$(9) = \frac{(9)}{(9)} \bullet (0)$
$C_{[x]:t}$	Método A: $C_{[x]:t} = \frac{1}{(1+i)^{t+1}} \bullet d_{[x]:t}$	$(3) = (9) \bullet (0)$
	Método B: $C_{[x]:t} = (1+i)^{100-[x]-(t+1)} \bullet d_{[x]:t}$	$(3) = (9) \bullet (0)$
$\bar{C}_{[x]:t}$	$\bar{C}_{[x]:t} = \frac{i+1}{\partial_{t+1}} \bullet C_{[x]:t}$	$(3) = \frac{(\#)}{9} \bullet (3)$ $= (9) \bullet (3)$
$D_{[x]:t}$	Método A: $D_{[x]:t} = l_{[x]:t} \bullet \frac{1}{(1+i)^t}$	$(1) = (9) \bullet (0)$
	Método B: $D_{[x]:t} = (1+i)^{(100-[x]-t)} \bullet l_{[x]:t}$	$(1) = (9) \bullet (0)$
$D_{[x]:t+\left(\frac{365-k}{365}\right)}$	$D_{[x]:t+\left(\frac{365-k}{365}\right)} = \frac{k}{365} \bullet D_{[x]:t} + \left(\frac{365-k}{365}\right) \bullet D_{[x]:t+1}$	$(3) = \frac{(0)}{(0)} \bullet (3) + \frac{(0)-(0)}{(0)} \bullet (3)$ $= (5) \bullet (3) + (5) \bullet (3)$ $= (3) + (3)$

donde k = número de días hasta el siguiente aniversario de la póliza



### Prima única neta para una cobertura de \$1 en la edad alcanzada

NOTA: Para el cálculo de la función continua, sustituya  $\bar{C}_{[x]}$  por  $C_{[x]}$ .

(Prorateado para cálculo a medio año)

Función	Derivación	Redondeo
$A_{[x]+1:\overline{a} } \left(\frac{365-k}{365}\right)$	$A_{[x]+1:\overline{a} } \left(\frac{365-k}{365}\right) = \frac{\left(\frac{k}{365} \cdot C_{[x]+1}\right) + \left(\sum_{n=1}^{a-[x]-1} C_{[x]+1+n}\right) + D_a}{D_{[x]+1:\overline{a} } \left(\frac{365-k}{365}\right)}$ <p>donde a= edad al vencimiento</p>	$(7) = \frac{\frac{(0)}{(0)} \cdot (3) + (3) + (1)}{(3)}$ $= \frac{(5) \cdot (3) + (3) + 1}{(3)}$ $= \frac{(3) + (3) + (1)}{(3)}$ $= \frac{(3)}{(3)}$

### Cantidad de nominal saldado reducido

Función	Derivación	Redondeo
${}_{1:\overline{a} } \left(\frac{365-k}{365}\right) W_{[x]}$	${}_{1:\overline{a} } \left(\frac{365-k}{365}\right) W_{[x]} = \frac{{}_{1:\overline{a} } \left(\frac{365-k}{365}\right) CV_{[x]}}{A_{[x]+1:\overline{a} } \left(\frac{365-k}{365}\right)}$	$(0) = \frac{(2)}{(7)}$ <p>NOTA: Redondee por arriba.</p>

### Valor de efectivo saldado reducido

Función	Derivación	Redondeo
${}_{1:\overline{a} } \left(\frac{365-k}{365}\right) CV_{[x]}$	${}_{1:\overline{a} } \left(\frac{365-k}{365}\right) CV_{[x]} = {}_{1:\overline{a} } \left(\frac{365-k}{365}\right) W_{[x]} \cdot A_{[x]+1:\overline{a} } \left(\frac{365-k}{365}\right)$	$(2) = (0) \cdot (7)$

### Valor de efectivo prorrogado

NOTA: Para el cálculo de la función continua, sustituya  $\bar{C}_{[x]}$  por  $C_{[x]}$ .

Función	Derivación	Redondeo
${}_{1:\overline{a} } \left(\frac{365-k}{365}\right) CV_{[x]}$	${}_{1:\overline{a} } \left(\frac{365-k}{365}\right) CV_{[x]} = \frac{\left(\frac{k}{365}\right) \cdot C_{[x]+1}}{D_{[x]+1:\overline{a} } \left(\frac{365-k}{365}\right)} \cdot ET_{face} +$	$(2) = \left[ \frac{(0)}{(0)} \cdot (3) \right] + \left[ \frac{(3)}{(3)} \cdot (0) \right] +$

$$\frac{\sum_{n=1}^m C_{|x|+1+n} \cdot ETI_{face} + \left(\frac{j}{365}\right) \cdot C_{|x|+m+1}}{D_{|x|+1} \left(\frac{365-k}{365}\right) \cdot ETI_{face}}$$

m = número de años completos de póliza prorrogados

j = número de días desde el aniversario del último año completo de póliza

$$\left[ \frac{\binom{0}{0} \cdot (3)}{\binom{0}{3}} \cdot (0) \right] = \frac{(5) \cdot (3)}{\binom{3}{3}} + \frac{(3)}{\binom{3}{3}} + \frac{(5) \cdot (3)}{\binom{3}{3}} = \frac{(5) \cdot (3) + (3) + (5) \cdot (3)}{\binom{3}{3}} = \frac{(3) + (3) + (3)}{\binom{3}{3}} = \frac{(3)}{\binom{3}{3}}$$

## Métodos de edad conjunta

Se proporcionan cuatro métodos para calcular la edad conjunta. Los pasos y las fórmulas para cada método se describen a continuación. La tabla siguiente define las variables utilizadas en las fórmulas.

Variable	Definición
A1	edad del asegurado principal
A2	edad del asegurado secundario
F1	edad del asegurado principal mujer
F2	edad del asegurado secundario mujer
D	diferencia de edad
Z	factor de ajuste
J	edad conjunta

### Método 1 — 58CSO

1. Reste 3 años a la edad de cualquier asegurado mujer:

$$A1 = F1 - 3$$

$$A2 = F2 - 3$$

2. Calcule la diferencia de edad:

$$D = A1 - A2$$

3. Determine el factor de ajuste:

si	$D < 2,$	$Z = 0$
si	$2 \leq D < 3,$	$Z = 1$
si	$3 \leq D < 5,$	$Z = 2$
si	$5 \leq D < 8,$	$Z = 3$
si	$8 \leq D < 12,$	$Z = 4$
si	$12 \leq D < 17,$	$Z = 5$
si	$17 \leq D < 25,$	$Z = 6$

si  $D \geq 25$ ,  $Z = 7$

4. Calcule la edad conjunta:

si  $A1 \geq A2$ ,  $J = A1 - Z$   
 si  $A1 < A2$ ,  $J = A2 - Z$

### Método 2 — 80CSO

1. Calcule la diferencia de edad:

$$D = A1 - A2$$

2. Según la combinación de sexo de la póliza, determine el factor de ajuste:

#### Dos asegurados varones

si  $D < 2$ ,  $Z = -2$   
 si  $2 \leq D < 4$ ,  $Z = -1$   
 si  $4 \leq D < 7$ ,  $Z = 0$   
 si  $7 \leq D < 10$ ,  $Z = 1$   
 si  $10 \leq D < 14$ ,  $Z = 2$   
 si  $14 \leq D < 20$ ,  $Z = 3$   
 si  $20 \leq D < 33$ ,  $Z = 4$   
 si  $33 \leq D < 56$ ,  $Z = 5$   
 si  $D \geq 56$ ,  $Z = 6$

#### Dos aseguradas mujeres

si  $D < 2$   $Z = 3$   
 si  $2 \leq D < 4$   $Z = 4$   
 si  $4 \leq D < 7$   $Z = 5$   
 si  $7 \leq D < 10$   $Z = 6$   
 si  $10 \leq D < 14$   $Z = 7$   
 si  $14 \leq D < 21$   $Z = 8$   
 si  $21 \leq D < 49$   $Z = 9$   
 si  $49 \leq D < 71$   $Z = 8$   
 si  $D \geq 71$   $Z = 7$

#### Mujer mayor (varón/mujer)

si  $D < 1$ ,  $Z = 0$   
 si  $1 \leq D < 3$ ,  $Z = 1$   
 si  $3 \leq D < 5$ ,  $Z = 2$   
 si  $5 \leq D < 7$ ,  $Z = 3$   
 si  $7 \leq D < 9$ ,  $Z = 4$   
 si  $9 \leq D < 12$ ,  $Z = 5$   
 si  $12 \leq D < 15$ ,  $Z = 6$   
 si  $15 \leq D < 19$ ,  $Z = 7$   
 si  $19 \leq D < 26$   $Z = 8$   
 si  $26 \leq D < 48$   $Z = 9$   
 si  $48 \leq D < 71$   $Z = 8$   
 si  $D \geq 71$ ,  $Z = 7$

#### Varón mayor (varón/mujer)

si  $D < 2$   $Z = 0$   
 si  $2 \leq D < 5$   $Z = 1$   
 si  $5 \leq D < 9$   $Z = 2$   
 si  $9 \leq D < 15$   $Z = 3$   
 si  $15 \leq D < 28$   $Z = 4$   
 si  $28 \leq D < 55$   $Z = 5$   
 si  $D \geq 55$   $Z = 6$

3. Calcule la edad conjunta:

si  $A1 \geq A2$ ,  $J = A1 - Z$   
 si  $A1 < A2$   $J = A2 - Z$

**Método 3 — Edad de emisión del asegurado mayor**

- Determine qué asegurado tiene la fecha de nacimiento anterior:

$$\text{si } A1 \geq A2, J = A1$$

$$\text{si } A1 < A2 \quad J = A2$$

**Método 4 — Edad de emisión del asegurado principal**

- Determine la edad del asegurado principal:

$$J = A1$$

**Métodos de cálculo del modo de prima**

Se proporcionan tres métodos para determinar cómo afectan los factores modales a la prima total de una póliza. Los pasos y las fórmulas para cada método se describen a continuación. La tabla siguiente define las variables utilizadas en las fórmulas.

Variable	Definición
P1	prima para el beneficio base
P2	prima para el primer beneficio de cláusula adicional
PN	prima para el enésimo beneficio de cláusula adicional
BM	factor modal para el beneficio base
RM	factor modal para el beneficio de cláusula adicional
C	cuota de cobranza
F	cuota de póliza
T	prima total para póliza

**Método 1**

Aplique el factor modal del beneficio base a las primas de todos los beneficios de la póliza:

$$T = (P1 \times BM) + (P2 \times BM) + \dots + (PN \times BM) + C + F$$

**Método 2**

Determine el factor modal de cada beneficio de la póliza y aplíquelo a la prima del beneficio:

$$T = (P1 \times BM) + (P2 \times RM) + \dots + (PN \times RM) + C + F$$

**Método 3**

Aplique el factor modal del beneficio base a la prima calculada para toda la póliza:

$$\{[(P1 + P2 + \dots + PN) + F] \times BM\} + C$$

**Cálculos de cargo por mortalidad**

Variable	Definición
----------	------------

AV	valor acumulado
AVp	valor acumulado anterior
AVb	valor acumulado al principio de la modalidad
MRT	cargo por mortalidad
Face	capital asegurado del beneficio
P	prima
LD%	% de carga sobre la prima
CORR	factor corredor
\$AMT	cantidad fija de corredor en dólares
CI	interés actual
DI	tasa de descuento del beneficio por fallecimiento
q	tasa de coste del seguro por dólar
PC	cargos por póliza
RC	cargos por cláusula adicional

1. Calcule el valor de fondos acumulado:

$$AV = [AVp + P(1 - LD\%) - PC - RC - MRT] \times (CI + 1)$$

2. Calcule el valor acumulado al principio de la modalidad:

$$AVb = AVp + P(1 - LD\%) - RC - PC$$

3. Utilice los valores calculados en los pasos 1 y 2 para calcular los cargos por mortalidad:

**Dependiendo del valor proyectado del fondo al principio de la modalidad de deducción**

Si se ha utilizado la opción A, realice la comprobación de corredor:

Si  $AVb \times CORR > Face$ , entonces  $Face = AVb \times CORR$

$$MRT = \left( \frac{Face}{1 + DI} - AVb \right) \times q$$

Si se ha utilizado la opción B, realice la comprobación de corredor:

Si  $AVb \times CORR > Face$ , entonces  $Face = AVb \times CORR$

$$MRT = \left( \frac{Face}{1 + DI} \right) \times q$$

**Dependiendo del valor proyectado del fondo al final de la modalidad de deducción**

Si es la opción A,

$$MRT = q \times \left[ \text{Face} - \frac{(1 + CI) \times AVb}{1 - q \times (1 + CI)} \right]$$

NOTA: si  $AVb \times (1 + CI) \times CORR > \text{Face}$ , entonces

$$MRT = AVb \times \frac{q \times (1 + CI)(CORR - 1)}{1 + [q \times (1 + CI)(CORR - 1)]}$$

Si es la opción B,

$$MRT = \text{Face} \times q$$

NOTA: si  $AVb \times (1 + CI) \times CORR > \text{Face} + [AVb \times (1 + CI)]$ , entonces

$$MRT = AVb \times \frac{q \times (1 + CI)(CORR - 1)}{1 + [q \times (1 + CI)(CORR - 1)]}$$

4. Suponiendo que  $ANN_{(n+1)}$  es igual a 0 para la primera iteración, calcule el factor de anualidad según la siguiente fecha de vencimiento de pago más próxima. Continúe iterando hasta alcanzar la fecha de emisión de la póliza. El pago de descuento es igual a \$1.00 desde la edad de 95 a la edad de efecto.

**Si se debe realizar un pago anual (mes = 1)**

Si es la opción A,

$$ANN_n = ANN_{(n+1)} \times \left( \frac{1}{1 + GI} - q \right) + (1 - LD\%_n)$$

Si se ha utilizado la opción B,

$$ANN_n = ANN_{(n+1)} \times \left( \frac{1}{1 + GI} \right) + (1 - LD\%_n)$$

**Si no se debe realizar ningún pago anual**

Si se ha utilizado la opción A,

$$ANN_n = ANN_{(n+1)} \times \left( \frac{1}{1 + GI} - q \right)$$

Si se ha utilizado la opción B,

$$ANN_n = ANN_{(n+1)} \times \left( \frac{1}{1 + GI} \right)$$

5. Calcule la Prima directriz única.  $PVB(n)$  es según la fecha de cálculo.

$$GDS = \frac{PVB_n}{1 - LD\%_{year1}}$$

6. Calcule la Prima directriz nivelada. PVB(n) es según la fecha de cálculo.

$$GDL = \frac{PVB}{ANN}$$

### Cantidades de reserva prescritas, GAAP y estatutarias

Las ecuaciones de las dos páginas siguientes indican cómo se calculan las cantidades de reserva estatutarias y GAAP, así como las cantidades de reserva prescritas que a veces se utilizan en el Proceso de reserva tributaria.

Variable	Definición
CF	factor actual
Days	número de días desde el último aniversario
IntF	factor media-terminal interpolado
MF	factor medio
Mos	número de meses desde el último aniversario
MTF	factor media-terminal
NPF	factor de prima neta
PF	factor anterior
RES	reservas
U	unidades

**Método de cálculo de medias utilizando factores medios**

**O terminal utilizando factores terminales**

**O media-terminal utilizando factores media-terminales**

Calcule la reserva y, a continuación, redondee:

$$RES = U \times CF$$

**Método de cálculo de medias utilizando factores terminales**

Calcule el factor medio y, a continuación, redondee:

$$MF = \frac{PF + CF + NPF}{2}$$

Calcule la reserva y, a continuación, redondee:

$$RES = U \times MF$$

**Método media-terminal utilizando factores terminales**

Calcule el factor terminal y, a continuación, redondee:

$$MTF = \frac{PF + CF}{2}$$

Calcule la reserva y, a continuación, redondee:

$$RES = U \times MTF$$

**Método media-terminal interpolado****Diario**

Calcule el factor media-terminal interpolado y, a continuación, redondee:

$$\text{IntF} = \frac{[(365 - \text{Days}) \times \text{PF}] + (\text{Days} \times \text{CF})}{365}$$

Calcule la reserva y, a continuación, redondee:

$$\text{RES} = U \times \text{IntF}$$

**Mensual**

Calcule el factor media-terminal interpolado y, a continuación, redondee:

$$\text{IntF} = \frac{[(115 - \text{Mos}) \times \text{PF}] + [(\text{Mos} + 5) \times \text{CF}]}{12}$$

Calcule la reserva y, a continuación, redondee:

$$\text{RES} = U \times \text{IntF}$$

**Cantidades negativas de reserva tributaria**

El campo Reserva neg. cero de la descripción del plan de reserva tributaria, permite cambiar cantidad negativas de reserva prescrita a cero.

**Pasos para los procedimientos de cálculo****Procedimiento A: El mayor del valor de reserva prescrita ajustada y de rescate de efectivo**

Para cada beneficio,

1. Calcule la cantidad de reserva prescrita según la Fórmula de cálculo prescrita que se ha seleccionado.
2. Cambia las cantidades negativas de reserva prescrita a cero, si como se indica en el Archivo de descripción del plan de reserva.
3. Redondea la cantidad de reserva prescrita.
4. Ajusta la cantidad de reserva prescrita:
  - si la Fórmula de cálculo prescrita es cualquier opción de la A a la E, reste la prima de 1er año calculada después de valoración y la prima de renovación vencida.
  - si la Fórmula de cálculo prescrita es A, B o C, reste la prima de 1er año y la prima de renovación diferida.
  - si la Fórmula de cálculo prescrita es D o E, añada la prima de primer año y las primas de renovación no devengadas.
5. Calcule el valor de rescate de efectivo (valor de efectivo menos el cargo por rescate).
6. Compare el valor de rescate de efectivo con el valor de rescate de efectivo garantizado, si hay alguno.



**Procedimiento B: El mayor del valor de reserva prescrita y de rescate de efectivo, pero no superior a la reserva estatutaria ajustada**

Para cada beneficio,

1. Calcula la cantidad de reserva prescrita según la Fórmula de cálculo prescrita que se ha seleccionado.
2. Cambia las cantidades negativas de reserva prescrita a cero, si así se ha indicado en el Archivo de descripción del plan de reserva.
3. Redondea la cantidad de reserva prescrita.
4. Ajusta la cantidad de reserva prescrita:
  - si la Fórmula de cálculo prescrita es cualquier opción de la A a la E, reste la prima de 1er año calculada después de valoración y la prima de renovación vencida.
  - si la Fórmula de cálculo prescrita es A, B o C, reste la prima de 1er año y la prima de renovación diferida.
  - si la Fórmula de cálculo prescrita es D o E, añada la prima de primer año y las primas de renovación no devengadas.
5. Calcula el valor de rescate de efectivo (valor de efectivo menos el cargo por rescate).
6. Compara el valor de rescate de efectivo con el valor de rescate de efectivo garantizado, si hay alguno.
7. Ajusta la cantidad de reserva estatutaria.
  - si la Fórmula de cálculo prescrita es cualquier opción de la A a la E, reste la prima de 1er año calculada después de valoración y la prima de renovación vencida.
  - si la Fórmula de cálculo prescrita es A, B o C, reste la prima de 1er año estatutaria y la prima de renovación diferida.
  - si la Fórmula de cálculo prescrita es D o E, añada la prima de primer año estatutaria y las primas de renovación no devengadas.

**Procedimiento C: Reserva estatutaria ajustada**

Para cada beneficio,

1. Resta la prima de 1er año estatutaria calculada después de valoración y la prima de renovación vencida de la cantidad de reserva estatutaria.
2. Resta la prima de 1er año estatutaria y la prima de renovación diferida de la cantidad de reserva estatutaria.
3. Añade la prima de 1er año estatutaria y las primas de renovación no devengadas a la cantidad de reserva estatutaria.

**Procedimiento D: Valor de efectivo**

Para cada beneficio,

1. Compara el valor de efectivo con el valor de efectivo garantizado; la cantidad mayor es la cantidad de reserva estatutaria..

**Procedimiento D: Valor de efectivo del rescate**

Para cada beneficio,

1. **Calcula el valor de rescate de efectivo (valor de efectivo menos el cargo por rescate).**
2. **Compara el valor de efectivo del rescate con el valor de efectivo garantizado; la cantidad mayor es la cantidad de reserva estatutaria..**

## Bibliografía

- Alan S. Fisher: **CASE Using Software Development Tools**; Wiley, usa, 1991
- Antigono Donati: **II contratto di Assicurazioni nel Codice Civile**, Roma, 1943.
- Booch Grady, Rumbaugh James, Jacobson Ivar; **The Unified Modeling Language, User Guide**; Assison-Wesley, 1988
- C. J. Date: **An Introduction to Database Systems**; Addison-Wesley, IBM (UK).
- C. W. Jordan: **Life Contingencies; Society of Actuaries**, Chicago, Illinois 1967.
- Carmen Fabela, Juan José López (traducción y revisión); **Life Support Plus Documentation**; Genelco Inc; St. Louis Mo. 1997.
- Gilles y Smith: **Managing Software Engineering CASE Studies and Solutions**.
- Grady Booch, James Rumbaugh, Ivar Jacobson; **El Lenguaje Unificado de Modelado**; Addison Wesley Iberoamericana, Madrid, 1999.
- Harriett E. Jones and Dani L. Long: **Principles of Insurance: Life, Health, and Annuities**; LOMA, usa, 1996.
- Ivar Jacobson, Grady Booch, James Rumbaugh; **El Proceso Unificado de Desarrollo de Software**; Pearson Education, S.A. Madrid, 2000.
- James Martin: **Organización de las Bases de Datos**; Prentice Hall, México, 1985.
- James Martin and Carma McClure: **Diagramming Techniques for Analysts and Programmers**; Prentice-Hall, Englewood Cliffs, New Jersey 1985.
- Joseph B. Maclean: **El Seguro de Vida**; CECSA, 1962.
- Larman Craig; **UML y Patrones, Introducción y Diseño Orientado a Objetos**; Prentice Hall, México, 1999
- Martín Fowler con Kendall Scott: **UML Gota a Gota**; Addison Wesley Longman de México, S.A. de C.V., 1999.
- Natalie López, Jorge Miguéis, Emmanuel Pichón: **Integrar UML en los proyectos; Gestión 2000, S.A., Barcelona, 1998.**
- Ocampo Camilo, Begoña Albizuri y Pere Botella: **De la Tecnología CASE a la Tecnología de procesos**; Soluciones avanzadas, enero de 1999, pp. 50-57.

---

Pierre Alain Muller: **Modelado de Objetos con UML**; Eyrolles, Gestión 2000, S.A., Barcelona, 1997.

James Rumbaugh, Ivar Jacobson Grady Booch; **El Lenguaje Unificado de Modelado. Manual de Referencia**: Pearson Education, S.A. Madrid, 2000

Rumbaugh James, Michael Blana, William Premertasi, Frederick Hedi, William Lorensen; **Object Oriented Modeling & Design**: Prentice Hall, Englewood Cliffs, NJ 1991

Rumbaugh James; **Object Oriented Modeling & Design**: Prentice Hall, Englewood Cliffs, NJ 1991

Vincenzo Ambriola; **Assessing Process-Centered Software Engineering Environments**: Università di Pisa, 1997