

**UNIVERSIDAD MAYOR DE SAN ANDRES
FACULTAD DE CIENCIAS PURAS Y NATURALES
CARRERA DE INFORMATICA**



Proyecto de Grado

**TEMA: SISTEMA INTEGRADO ESTADISTICO Y CONTROL DE LA RABIA EN
BOLIVIA**

DOCENTE (TUTOR) :

LIC. EFRAIN SILVA

DECENTE (REVISOR):

LIC. GERMAN HUANCA

ALUMNA

:

**CINTHIA FRANCO FARFAN
CI. 5638796 CH.**

**LA PAZ – BOLIVIA
2008**

Dedicatoria

Este proyecto esta dedicado a dos grandes hombres que me dieron un gran ejemplo de vida; a mi abuelito Félix Franco Raya una persona muy intelectual y sabia que me inculco sabiduría y me impulso al fantástico mundo de la universidad y a mi abuelito Luís Farfán Aviles (Q.D.D.G.) un gran deportista y hombre noble que siempre me supo llenar de confianza y me enseñó el valor del esfuerzo de la vida.

Pedirles a estos dos grandes hombres me permitan compartir esta dedicatoria a la mujer más sabia y noble que conozco, mí madre Nayda Farfán Martínez, que siempre confió en mí y que gracias a ella me atrevo a decir que soy una persona de bien.

Cinthia Franco Farfán

Agradecimientos

A Jehová por que nunca me abandono y siempre estuvo conmigo en los momentos mas difíciles que pase.

A mis padres Guido y Nayda que siempre me dieron su confianza y las fuerzas para salir adelante, gracias por haber esperado un buen tiempo, ahora no los defraudare.

A mis cuatro abuelitos Félix, Luisa, Lucho, Teresa porque me enseñaron los grandes valores de la vida, valores que siempre llevare conmigo y trataré de transmitir a quienes sea posible.

Al Dr. Freddy Lizon; Lic. Efraín Silva; Lic. German Huanta. Por sus consejos, confianza y enseñanzas que hicieron que yo llegue a una de mis metas que es terminar la universidad como una verdadera profesional.

A mis tíos que me apoyaron y ayudaron cuando lo necesite, pero un agradecimiento muy especial para una gran tía, mi tía Carmen que me ayudo en la universidad y nunca dejo de ayudarme dándome su apoyo y su cariño incondicional en todo.

A Elio que siempre estuvo conmigo cuando mas necesite de alguien y supo darme su apoyo incondicional y todo su amor.

A mis amigos Reinaldo (PG), Karina, Ardy, Rubén, Sara, Cristhian y muchos mas. Que siempre me ayudaron, me dieron el tesoro de una grande amistad y momentos inolvidables; pero quiero dar un agradecimiento muy especial a Ronald que es un gran amigo y una gran persona que me ayudo mucho a que pudiera llegar al final de esta etapa.

Cinthia Franco Farfán

Resumen

En el Proyecto de Grado denominado “SISTEMA INTEGRADO ESTADISTICO Y DE CONTROL DE LA RABIA EN BOLIVIA” se muestra de manera eficiente estadísticas numéricas, barras y geográficas de los resultados de las muestras de rabia en Bolivia para el Programa Nacional de Zoonosis. Este sistema tiene por objetivo facilitar la alimentación de datos de los laboratorios para dicha institución, representar gráficamente sus estadísticas de manera automática.

En el proyecto se hace uso de principios y conceptos de la metodología OMT (Object Modeling Technique) que presta sostén a todo el ciclo de vida completo del software. Las herramientas que se utilizaron para su desarrollo son: PHP (Hypertext Pre-processor) Lenguaje de programación usado generalmente en la creación de contenidos para sitios Web, MySQL es un sistema de gestión de bases de datos (SGBD) multiusuario, multiplataforma y de código abierto, ArcView es la herramienta SIG más extendida en todo el mundo, dadas sus avanzadas capacidades de visualización, consulta y análisis de información geográfica, además de las numerosas herramientas de integración de datos desde todo tipo de fuentes y herramientas de edición.

La función de las metodologías y herramientas mencionadas anteriormente permitió desarrollar un sistema que cumple con todos los requerimientos de usuario. Es así como se da el cumplimiento al desarrollo del problema.

INDICE

INTRODUCCIÓN

1.1. INTRODUCCION.....	2
1.2. ANTECEDENTES.....	3
1.3. IDENTIFICACION DEL PROBLEMA.....	6
1.4. OBJETIVOS.....	6
1.4.1. OBJETIVO GENERAL.....	6
1.4.2. OBJETIVOS ESPECIFICOS.....	6
1.5. JUSTIFICACIONES.....	7
1.5.1. JUSTIFICACION ECONOMICA.....	7
1.5.2. JUSTIFICACION TECNICA.....	7
1.5.3. JUSTIFICACION SOCIAL.....	8
1.6. METODOS Y TECNICAS.....	8
1.6.1. MÉTODO OMT.....	8
1.7. ALCANCES Y APORTES.....	9
MARCO TEÓRICO	
2.1. SISTEMAS ESTADÍSTICOS.....	11
2.1.1. Aplicaciones de la estadística.....	11
2.1.2. Divisiones De La Estadística.....	12
2.2. SISTEMAS GEOGRÁFICOS.....	12
2.2.2. Propósito.....	12
2.2.3. Banco de Datos.....	14
2.2.4. Metodología sistémica para la implantación de sistemas de información geográficos.....	16
2.3. SISTEMAS DE CONTROL.....	20
2.3.1. Factores de Control.....	20
2.3.1.1. Importancias Del Control.....	20
2.3.1.2. Tipos De Controles.....	21
2.3.1.3. Áreas Del Control.....	21
2.3.3.1. Gráfica De Punto De Equilibrio.....	22
2.3.3.2. La gráfica de gantt.....	22

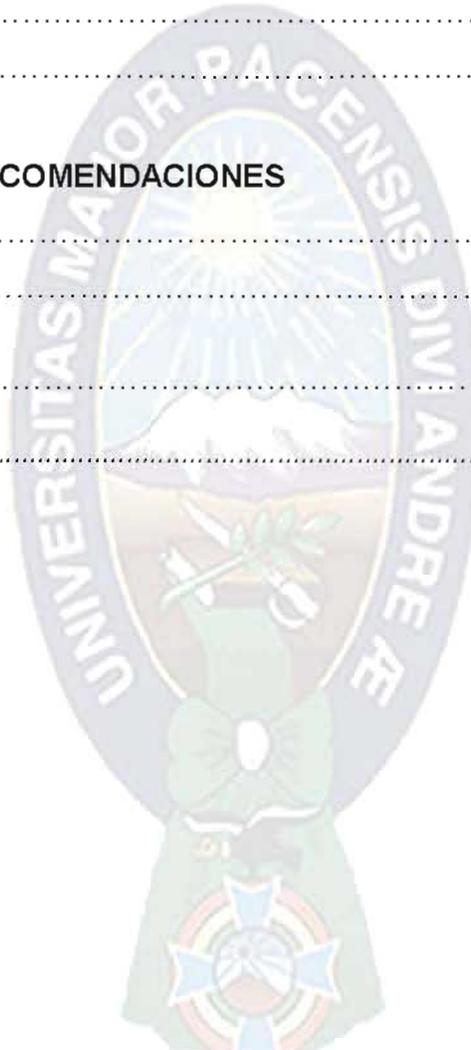
2.3.3.3. Pert (Técnica De Revisión Y Evaluación De Programas).....	23
2.4. METODOLOGÍA OMT.....	23
2.4.1. Fases de la Metodología OMT.....	24
2.4.1.1. Fase de Análisis.....	25
2.4.1.2. Fase de Diseño de sistemas.....	26
2.4.1.3. Fase de Diseño de objetos.....	27
2.4.2. Modelos que utiliza el OMT.....	28
2.4.2.1. Modelo de Objetos.....	29
2.4.2.2. Modelo Dinámico.....	38
2.4.2.3. Modelo Funcional.....	46
2.4.3. Ventajas y Desventajas de la Metodología OMT.....	54
2.4.3.1. Ventajas.....	54
2.4.3.2. Desventajas.....	54
2.4.4. Aplicaciones de la Metodología OMT.....	54
2.5. TECNOLOGIA.....	55
2.5.1. Red.....	55
2.5.2. Internet.....	56
2.5.3. Sitio Web.....	57
2.6. HERRAMIENTAS.....	58
2.6.1. PHP (Hypertext Pre-processor).....	58
2.6.2. MySQL.....	58
2.6.3. Arcview3.2.....	59
2.7. SEGURIDAD.....	63
2.7.1. Requisitos Fundamentales de la Seguridad en Cómputo.....	63
2.7.1.1. Política de Seguridad.....	64
2.7.1.2. Marcas.....	64
2.7.1.3. Identificación.....	65

2.7.1.4. Responsabilidad.....	65
2.7.1.5. Aseguramiento.....	65
2.7.1.6. Protección Continúa.....	66
2.7.2. Rutas Seguras.....	66
2.7.3. Auditoria.....	66
2.7.4. Recuperación Confiable.....	67
2.7.5. Diseño de Documentación.....	67

MARCO APLICATIVO

3.1. Fase de Análisis.....	69
3.1.1. Enunciado del Problema.....	69
3.1.2. Modelo de Objetos.....	69
3.1.2.1. Identificación de Objetos y Clases.....	69
3.1.2.2. Diccionario de datos.....	73
3.1.2.3. Construcción del diagrama de clases básico.....	74
3.1.3. Modelo Dinámico.....	75
3.1.3.1. Preparar escenarios para las secuencias de iteración típicas...	75
3.1.3.2. Preparar trazos de eventos para cada escenario.....	76
3.1.3.3. Diagrama de Flujo de Eventos para el Sistema.....	79
3.1.3.4. Diagrama de de estados para cada clase con comportamiento dinámico.....	80
3.1.4. Modelo Funcional.....	83
3.1.4.1. Diagrama de Contexto.....	83
3.1.4.2. Diagrama de Flujo de Datos (Padre).....	83
3.1.4.3. Diagrama de Flujo de Datos (Hijos).....	84
3.2. Fase de Diseño de Sistemas.....	85
3.2.1. Estructura de la Arquitectura Básica del Sistema.....	85
3.2.1.1. Organización del sistema en subsistemas.....	85
3.2.2. Dediciones de estrategias de alto nivel.....	86
3.2.2.1. Asignación de subsistemas a procesadores y tareas.....	86
3.3. Fase de Diseño de Objetos.....	87

3.3.1. Composición de los tres modelos detallado.....	87
 METRICAS DE CALIDAD	
4.1. ISO9126.....	89
4.1.1. Funcionalidad.....	90
4.1.2. Fiabilidad.....	92
4.1.3. Usabilidad.....	93
4.1.4. Eficiencia.....	93
 CONCLUSIONES Y RECOMENDACIONES	
5.1. Conclusiones.....	96
5.2. Recomendaciones.....	96
 Bibliografía	 97
 ANEXOS	 99

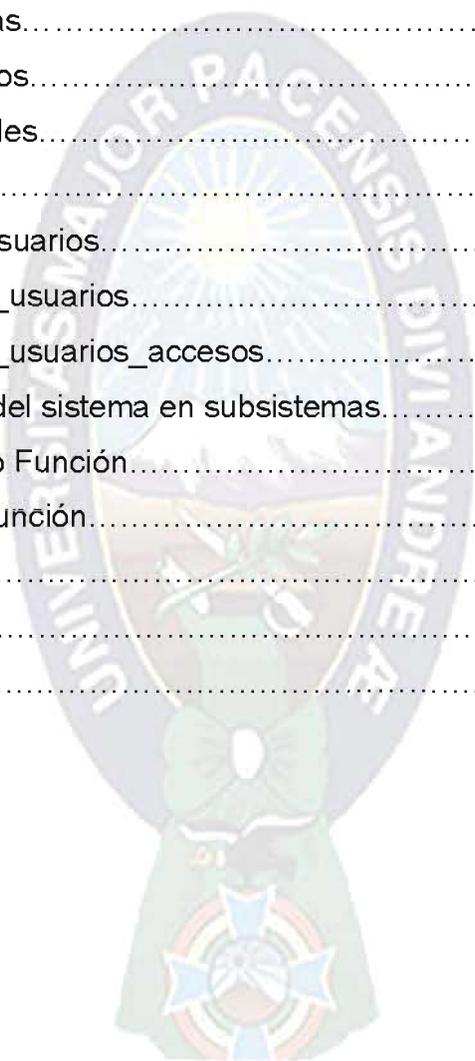


INDICE DE FIGURAS

Figura 1: Grafico de Tendencia Secular y Tasa de Incidencia de Rabia Canina Bolivia 1991 – 2007.....	3
Figura 2: Banco de Datos.....	14
Figura 3: Banco de Datos.....	15
Figura 4: Notaciones del modelo de objetos.....	37
Figura 5: Notaciones del modelo de objetos avanzado.....	38
Figura 6: Notaciones del modelo dinámico.....	46
Figura 7: Notaciones del modelo funcional.....	53
Figura 8.1.: Arcview.....	59
Figura 8.2.: Arcview.....	60
Figura 8.3.: Arcview.....	61
Figura 8.4.: Arcview.....	62
Figura 8.5.: Arcview.....	62
Figura 9: Identificación de objetos y clases.....	69
Figura 10: Diagrama de clases básico.....	74
Figura 11: Trazos de eventos.....	76
Figura 12: Diagrama de flujo de eventos.....	79
Figura 13: Diagramas de estados.....	80
Figura 14: Diagrama de contexto.....	83
Figura 15: Diagrama de flujo de datos.....	83
Figura 16: Diagramas de flujo de datos al detalle.....	83
Figura 17: Composición de los tres modelos detallado.....	84

INDICE DE TABLAS

Tabla 1: Objeto Rabia.....	70
Tabla 2: Objeto Medicamentos.....	70
Tabla 3: Objeto Control_medicamentos.....	71
Tabla 4: Objeto departamento.....	71
Tabla 5: Objeto provincias.....	71
Tabla 6: Objeto municipios.....	71
Tabla 7: Objeto localidades.....	72
Tabla 8: Objeto usuarios.....	72
Tabla 9: Objetos tipos_usuarios.....	72
Tabla 10: Objeto control_usuarios.....	72
Tabla 11: Objeto control_usuarios_accesos.....	72
Tabla 12: Organización del sistema en subsistemas.....	85
Tabla 13: Métricas Punto Función.....	90
Tabla 14: Factor Punto función.....	91
Tabla 15: Fiabilidad.....	92
Tabla 16: Usabilidad.....	93
Tabla 17: Eficiencia.....	93





CAPITULO I

INTRODUCCION

SISTEMA INTEGRADO ESTADISTICO Y DE CONTROL DE LA RABIA EN BOLIVIA

1.1. INTRODUCCION

Por naturaleza la rabia no es una enfermedad de la especie humana ya que ésta es transmitida en forma accidental a consecuencia de la sobre diseminación de reservorios animales. Sin embargo, el mayor impacto en salud pública es causado por la rabia urbana, cuyo transmisor es el perro. Desde hace varias décadas el país ha adoptado diversas estrategias de control contra este tipo de rabia, alcanzando generalmente sólo éxitos parciales y pocas veces sostenidos en el tiempo. Por tanto, los estudios relacionados con el conocimiento de la estructura poblacional canina y aspectos epidemiológicos y ecológicos de la rabia se tornan importantes a la hora de formular campañas de control. Este trabajo tiene por objetivo contribuir al conocimiento y control de la epidemiología de rabia canina en Bolivia.

El sistema se basara especialmente en el control de “rabia” a nivel nacional, el sistema deberá generar reportes estadísticos de tipo numérico, barras y geográficos. Se recibirá información de tres Laboratorios que están en las ciudades de La Paz, Cochabamba y Santa Cruz los cuales realizan análisis de “rabia” de todos los departamentos por ser los únicos Laboratorios en Bolivia que hacen ese tipo de análisis, estos Laboratorios mandan información semanal a la Jefatura Nacional de Zoonosis para que estos puedan manipular esa información y así tener un control de la “rabia” que existe en Bolivia.

La institución requiere que los reportes estadísticos sean generados por Departamento, Provincia, Municipio y Gerencia de Red, esto último se refiere a divisiones que tiene el Ministerio de Salud dentro de cada Ciudad.

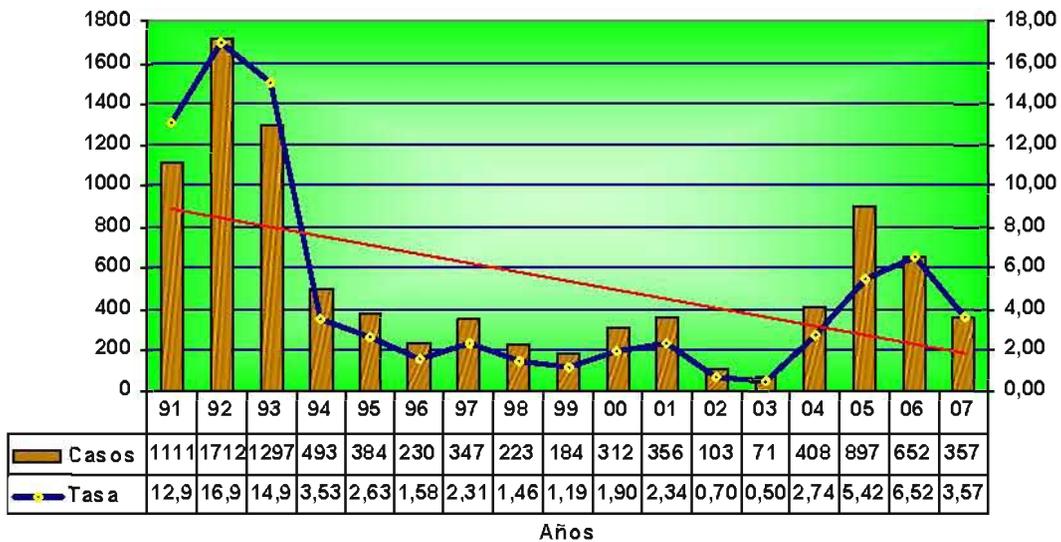
Este sistema funcionara vía Web, para que los laboratorios puedan llenar sus formularios desde cada departamento, a la vez el manipuleo de datos de forma interna será solo para la Jefatura Nacional de Zoonosis.

Los reportes realizados por el sistema serán publicados por la página Web, ya que esto dará mayor atención al funcionamiento de dicha institución, también el público en general podrá enterarse de los últimos reportes y novedades de la institución.

1.2. ANTECEDENTES

En Bolivia en el periodo 1970 – 1979 los datos de rabia eran dispersos y existía un sub registro de su incidencia por la ausencia de un sistema de vigilancia. Durante los años 1980-1989, el ex Ministerio de Salud y Previsión Social, decidió asumir el control de esta enfermedad; seguidamente, como resultado del Primer Seminario Nacional de Control de la Rabia Canina organizado por el sector salud y con la participación del Ex Ministerio de Asuntos Campesinos y Agropecuarios, el Servicio Nacional de control de la Aftosa, Rabia y Brucelosis (SENARB) e involucrando a otros sectores se da inicio a la sistematización de la vigilancia epidemiológica de esta zoonosis; aspecto que mejoró cualitativamente como cuantitativamente el registro de casos de rabia canina en el país a partir de 1991 (Grafico, facilitado por el Dr. Freddy Lizoni F. Responsable Programa Nacional Zoonosis).

**Grafico de Tendencia Secular y Tasa de Incidencia de Rabia Canina
Bolivia 1991 - 2007**



**Figura 1: Grafico de Tendencia Secular y Tasa de Incidencia de Rabia Canina
Bolivia 1991 - 2007**

De acuerdo a este Grafico, en el último decenio existen oscilaciones en el número de casos positivos por ejemplo; en el año 1997 se registraron 347 casos y por el contrario en el 2003 se registraron 71. Aparentemente en lo que respecta a este decenio la incidencia de rabia canina mas baja fue el 2003. Por lo tanto la incidencia no descendió en estos dos años comparados por los factores ya mencionados.

Nuevamente en la reunión de Ministros de Salud de las Américas (RIMSA - 2005), se planteó la meta de erradicación de la rabia canina urbana para el 2009. Bolivia en esa gestión registro la mayor epidemia de Sud América.

Ante esta situación el 2006 se decide implementar el Plan Operativo Nacional de Control de la Rabia (PONCRA) esta responsabilidad fue elaborada en coordinación con los Servicios Departamentales de Salud (SEDES) de todo el país, los gobiernos municipales de ciudades capitales y la OPS. Se elaboro el primer Plan Operativo en la ciudad de Santa Cruz.

En esta gestión y al haber concluido los Planes Operativos de las otras 8 ciudades capitales se logro que:

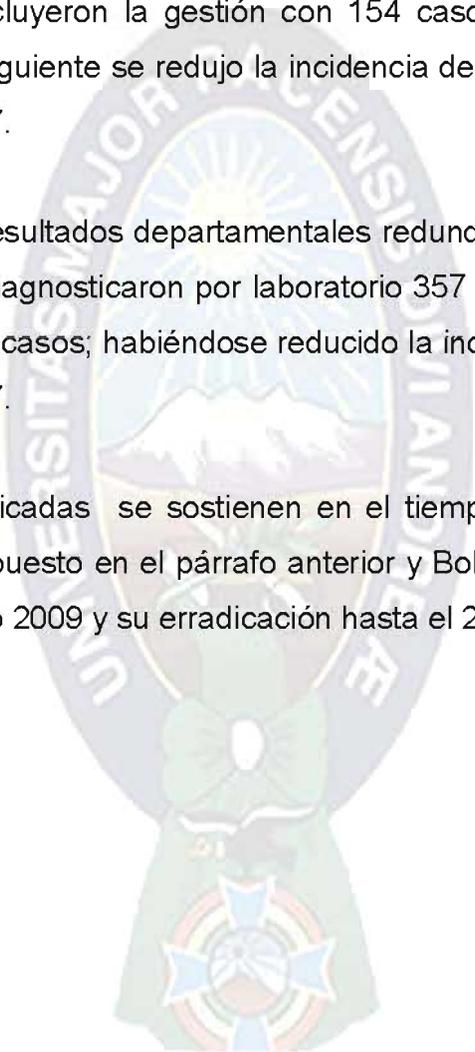
1. El Ministerio de Salud y Deportes otorgue los 100% de los biológicos requeridos por los departamentos.
2. Los SEDES y Gobiernos Municipales confieran recursos humanos y económicos suficientes para lograr coberturas masivas de vacunación mayores al 80%.
3. Los Gobiernos Municipales implementen políticas para el control de la sobrepoblación animal.

4. Los SEDES y Gobiernos Municipales organicen y planifiquen los Planes Operativos complementarios hasta el 2011.

Como ejemplo de esta organización y planificación el año 2006 el municipio de Santa Cruz y el SEDES lograron una cobertura de vacunación masiva del 73% y finalizaron el año con 399 casos de rabia canina. Para el 2007 al tener una cobertura del 75% concluyeron la gestión con 154 casos de rabia canina y 3 fallecimientos. Por consiguiente se redujo la incidencia de rabia canina en un 62% entre el año 2006 y 2007.

A nivel nacional estos resultados departamentales redundaron a nivel nacional es así que en el 2007 se diagnosticaron por laboratorio 357 casos positivos de rabia canina; el 2006 con 652 casos; habiéndose reducido la incidencia en mas del 55% entre el año 2006 y 2007.

Si las actividades planificadas se sostienen en el tiempo, la rabia se reducirá drásticamente por lo expuesto en el párrafo anterior y Bolivia podrá controlar esta enfermedad hasta el año 2009 y su erradicación hasta el 2011.



1.3. IDENTIFICACION DEL PROBLEMA

Observando los antecedentes mencionados, podemos aseverar una serie de falencias que se mencionaran a continuación:

- Los datos que mandan los Laboratorios de La Paz, Cochabamba y Santa Cruz a diferentes interventores no coinciden.
- Envío de Datos en Word o Excel por correo electrónico.
- Las planillas de datos entre los Laboratorios de La Paz, Cochabamba y Santa Cruz son diferentes.
- Transcripción de datos en forma manual de cada Laboratorio.
- Hacer una búsqueda por cada departamento para realizar sus reportes estadísticos.
- La no existencia de información a la población Nacional.

Con todos estos problemas podemos llegar a la conclusión del problema general que sería el de no tener un sistema independiente y necesario para realizar sus reportes estadísticos, control de rabia a nivel nacional y sus publicaciones al público en general.

1.4. OBJETIVOS

1.4.1. OBJETIVO GENERAL

Desarrollar un “Sistema Integrado Estadístico y de Control de la Rabia en Bolivia”, para la Jefatura Nacional de Zoonosis.

1.4.2. OBJETIVOS ESPECIFICOS

- Dar un solo formato a los datos enviados por los Laboratorios de La Paz (INLASA), Cochabamba (LIDIVECO) y Santa Cruz (LIDIVET).
- Registrar los datos de los Laboratorios semanalmente.

- Sacar reportes estadísticos numéricos a nivel Nacional, Departamental y por Provincias, semanales y anuales.
- Sacar reportes estadísticos en barras a nivel Nacional, Departamental y por Provincias, semanales y anuales.
- Sacar reportes estadísticos geográficos a nivel Nacional, Departamental y por Provincias, semanales y anuales.
- Tener la opción para que solo la jefatura manipule los datos adquiridos.
- Tener un control de las vacunas que se mandan a los distintos Departamentos de Bolivia.
- Dar a conocer al público en general las últimas estadísticas semanales de la “rabia” en Bolivia.

1.5. JUSTIFICACIONES

1.5.1. JUSTIFICACION ECONOMICA

El sistema ayudara a reducir el tiempo de proceso de datos, y recabar en el menor tiempo posible toda información necesaria, otro aspecto importante es la reducción del material de escritorio y a la vez reduce el costo de trabajo del personal del área.

1.5.2. JUSTIFICACION TECNICA

La institución para una justificación técnica respecto al hardware cuenta con la última tecnología en computadoras, ya que es una institución ministerial y pertenece al gobierno, este último distribuye a las distintas instituciones por ley material de última generación; también cuenta con un servidor exclusivo para todo el Ministerio de Salud y Deportes. Respecto al software se utilizara PHP 5.0.1 y MySQL 4.0.2.

Todas estas herramientas nos ayudaran a implementar el sistema vía Web.

1.5.3. JUSTIFICACION SOCIAL

El sistema que se implementara generara un gran beneficio hacia la institucion, puesto que ayudara a tener datos y estadísticas mas exactas para la toma de dediciones, también ayudara a que la poblacion pueda saber de la situacion de la "rabia" en Bolivia.

1.6. METODOS Y TECNICAS

1.6.1. MÉTODO OMT

Una de las metodologías mas importantes y completas para la implementación de un sistema es OMT con su traducción en ingles (Object Modeling Technique), ya que este método hace un cubrimiento de las etapas de análisis, diseño e implementación definidas por la OMG, dejando sin cubrir el modelamiento estratégico.

Podemos mencionar algunas de sus características principales que nos ayudaran a desarrollar el sistema:

1. *Modelo de Objetos*. Se define como un diagrama de objetos mas un diccionario de datos. El diagrama de objetos muestra clases y sus relaciones (generalización, agregación, asociación, instanciación). El diccionario de datos es el detalle de las clases en el diagrama de objetos.
2. *Modelo dinámico*. Se define como un conjunto de diagramas de estado más un diagrama de Flujo de eventos Global.
3. *Modelo funcional*. Es un diagrama de flujo con restricciones.

OMT tiene un conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema de software; y tiene las siguientes etapas:

Análisis

- Documento de análisis, que incluye:
 - Descripción del problema
 - Modelo de Objetos
 - Modelo dinámico
 - Modelo funcional

Diseño del sistema

- Definición de subsistemas

Diseño de objetos

- Documento de diseño, que incluye versiones detalladas de los modelos de objetos, dinámico y funcional

Implementación

- Diseño de bases de datos, si se requieren
- Código

1.7. ALCANCES Y APORTES

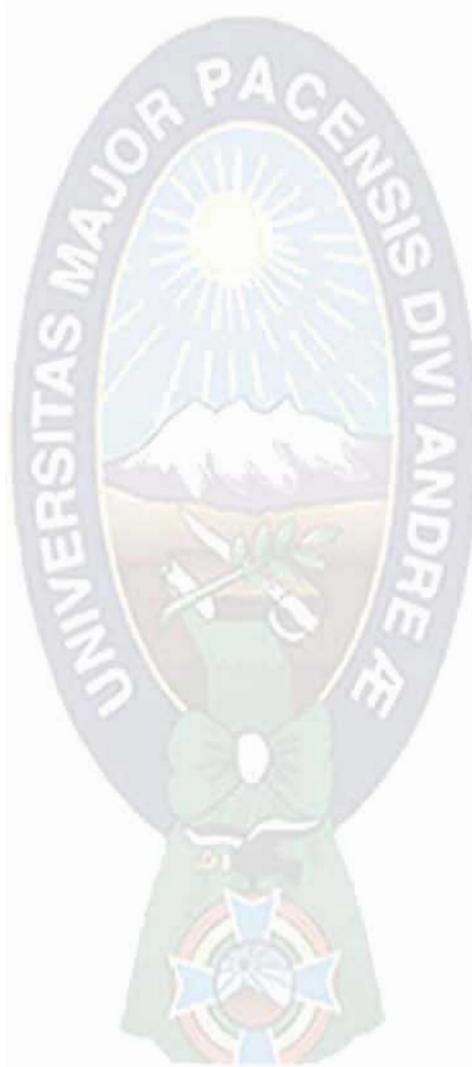
El proyecto se basa en el análisis y diseño de la implementación del sistema de información de Zoonosis vía Web:

La recopilación de todas las operaciones involucradas con a la rabia canina y otros.

Procesar y presentar cuadros indicadores referidos a la cantidad de rabia encontradas o contagiadas de los caninos.

Emitir estadísticos referentes a la clasificación en las áreas de servicios de los laboratorios de las diferentes ciudades.

Reportes especializados en el área de información sectorial de forma periódica.



CAPITULO II

MARCO TEORICO

Marco Teórico

En este capítulo se desarrollara todas las Teorías fundamentales para el proyecto partiendo de la recopilación de información hasta su implementación.

2.1. SISTEMAS ESTADÍSTICOS

Un Sistema Estadístico es un conjunto de informes numéricos derivados de los censos de población, de datos de registros y de informes.

La palabra estadística se entiende como una disciplina que estudia cuantitativamente los fenómenos de masa o colectivos, o sea, aquellos fenómenos cuyo estudio solo puede efectuarse a través de una colección de observaciones.

2.1.1. Aplicaciones de la estadística

La estadística es un potente auxiliar de muchas ciencias y actividades humanas: sociología, geografía humana, economía, informática, etc.

Es una herramienta indispensable para la toma de decisiones.

También es ampliamente empleada para mostrar los aspectos cuantitativos de una situación.

La estadística está relacionada con el estudio de proceso cuyo resultado es más o menos imprescindible y con la finalidad de obtener conclusiones para tomar decisiones razonables de acuerdo con tales observaciones.

El resultado de estudio de dichos procesos, denominados procesos aleatorios, puede ser de naturaleza cualitativa o cuantitativa y, en este último caso, discreta o continúa.

Son muchas las predicciones de tipo sociólogo, o económico, que pueden hacerse a partir de la aplicación exclusiva de razonamientos probabilísticas a conjuntos de datos objetivos como son, por ejemplo, los de naturaleza demográfica.

Las predicciones estadísticas, difícilmente hacen referencia a sucesos concretos, pero describen con considerable precisión en el comportamiento global de grandes conjuntos de sucesos particulares. Son predicciones que, en general, no acostumbran resultar útiles.

2.1.2. Divisiones De La Estadística

La estadística formula reglas y procedimientos para la presentación de una masa de datos en una forma más útil y significativa. Establece normas para la representación grafica de los datos. También son una base importante para el análisis en casi todas las disciplinas académicas.

- *Descriptiva
- *Cualitativa
- *Cuantitativa

2.2. SISTEMAS GEOGRÁFICOS

SIG es un sistema de hardware y software para gerenciamiento y visualización de datos espaciales. Es similar aun gerenciador de Bancos de Datos que se utiliza principalmente en datos espaciales y no solamente tabulares.

2.2.2. Propósito

Aunque algunos ejemplos de uso de los SIG puedan ser encontrados en las definiciones, serán listadas aquí algunas tareas ejecutadas por SIG.

Organización de datos: Almacenar datos de modo de sustituir una mapoteca analógica por una mapoteca digital posee ventajas obvias, entre las cuales pueden ser citadas una reducción en el espacio físico; el fin del deterioro de los productos en papel; la pronta recuperación de los datos; la posibilidad de ser producidas copias sin perdida de calidad; y diversas otras.

Visualización de datos: La posibilidad de seleccionar los niveles de información deseados, montándose mapas temáticos de acuerdo con el contexto supera cualquier producto en papel. A pesar de subestimada, la capacidad de análisis del ojo humano es esencial en un estudio que involucra la información espacial.

Producción de mapas: En general los SIG poseen Herramientas completas para la producción de mapas, tornando bastante simples la inclusión de grillas de coordenadas, escalas gráfica y numérica, leyenda, norte y textos diversos, siendo mucho mas indicados para la cartografía del que los simples sistemas CAD.

Consulta espacial: Posiblemente la función mas importante de los SIG. La posibilidad de preguntar cuales son las propiedades de un determinado objeto, o en cuales lugares tienen tales propiedades, Esto vuelve la interacción entre el usuario y los datos extremadamente dinámicos y poderosos.

Análisis espacial: Consiste en el uso de un conjunto de técnicas de combinación entre los niveles de información, de modo de evidenciar patrones dentro de datos anteriormente ocultos al analista. es una manera de inferir significado a partir de los datos.

Previsión: Uno de los propósitos de los SIG es el de verificación de escenarios, modificándose los parámetros de manera de evaluar como los eventos, naturales o no, ocurrirían si las condiciones fuesen diferentes, obteniendo un conocimiento más general de los objetos o área en estudio.

Abajo están listadas algunas consultas simples que pueden ser realizadas con un SIG:

- Simple recuperación de los datos
- Donde está el objeto A?
- Que es este objeto?
- Sumar los atributos de los objetos dentro de una distancia X
- Sumar los atributos de los objetos dentro de determinada área

- Cual es la mejor ruta?
- Muestre todos los objetos que satisfagan un criterio
- Use la relación entre los objetos
- Y diversas otras.

2.2.3. Banco de Datos

Una característica de los SIG es trabajar con datos que poseen un componente espacial (una posición geográfica definida) y un componente no-espacial (sus atributos: propiedades y valores) implica que un usuario debe tener conocimiento de las Herramientas de diseño (parte gráfica) y de tablas y relacionamientos (banco de Datos). En este contexto, cabe aquí una definición de algunos términos que serán útiles.

Datos son un conjunto de registros distintos y objetivos, relativos a eventos. Los datos son los registros en bruto, Registros detallados que existen en grandes volúmenes en toda organización. Una búsqueda mineral, corresponde aquí los que se han recolectado en el campo. Diferente de dato, la información tiene significado. Información son "datos dotados de relevancia y propósito", representadas por la formalización ordenada y útil de los datos. Típicamente, información son datos obtenidos a través de un procesamiento y presentados de una manera que sea relevante al receptor. Y, finalmente, conocimiento es la capacidad de transformar información en resultado de valor. Las informaciones son creadas a partir de una base de datos y los conocimientos, a su vez, son creados a partir de la base de informaciones en función de los resultados de valor.



Figura 2: Banco de Datos

Aunque existan diversos modelos de bancos de datos, como por ejemplo los bancos de datos jerárquicos, los relacionales e los orientados a objeto, en general el modelo relacional es una manera conveniente de representar la realidad, y será descrito aquí por este motivo. Quien se interese por las características de los datos espaciales y su adecuación a los diferentes modelos de bases de datos puede seguir este link para la University of British Columbia.

Un banco de datos relacional ésta compuesto por tablas, en la cuales se almacena la información sobre objetos. El contenido de una línea de la tabla, o registro, representa un objeto con todas sus características, y por tanto cada objeto está relacionado a un (y solo a un) registro. Cada columna, o campo, se refiere a una propiedad o atributo de este objeto.

Y aún es posible que sean establecidos relacionamientos entre diferentes tablas, basados en un campo común entre las mismas, de modo que sea posible consultar atributos de un objeto que esta almacenados en tablas diferentes.

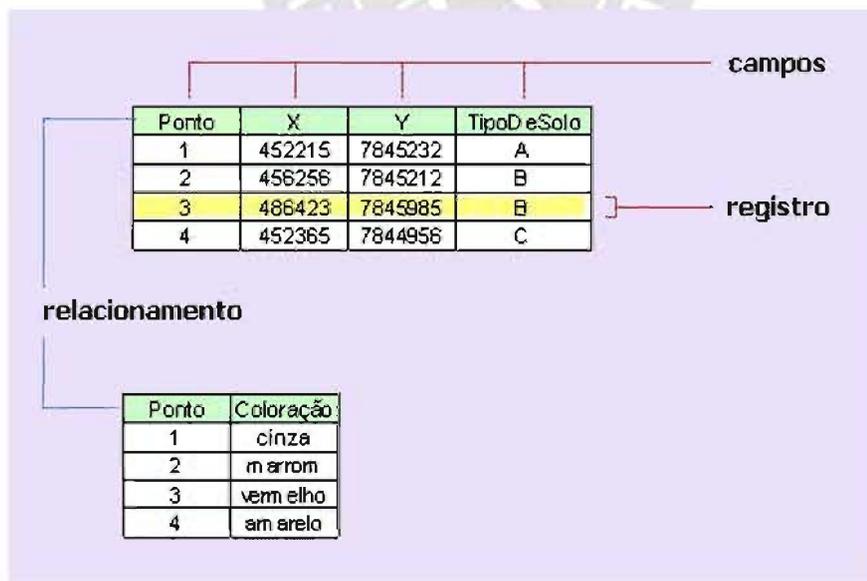


Figura 3: Banco de Datos

2.2.4. Metodología sistémica para la implantación de sistemas de información geográficos.

La metodología propuesta contempla la implantación de un SIG, como un proceso evolutivo, es decir la presentación de un proyecto piloto para definir evolutivamente los requerimientos de los usuarios, hasta lograr progresivamente la implantación completa de manera exitosa. Asimismo, es importante tener siempre en cuenta que implantar un SIG implica un período largo de tiempo, en el cual constantemente se incorporan nuevos requerimientos tecnológicos y de información. Es por ello, que la metodología propuesta persigue brindarle un enfoque evolutivo e incremental al proceso de implantación de un SIG.

A continuación se detallan los aspectos más relevantes a ser considerados en cada una de las fases de la metodología propuesta.

- **Innovación/Medio Ambiente.**

La innovación se propone para facilitar y optimizar los procesos tradicionales de una organización.

- **Organización Informada.**

La información en la organización se recibe por el personal que desea adoptar la nueva tecnología, y se crea la necesidad de implementarla.

- **Divulgación al Personal.**

En esta fase se decide dar conocimiento a todos los miembros de la organización, utilizando como estrategias la unión de las dos perspectivas de difusión propuestas por Aronoff para implantar una nueva tecnología.

- **Formación/Educación.**

En esta fase la organización tomó ya la decisión de implantar el SIG, y por ende asumir todo el proceso de implantación.

- **Análisis de las Necesidades de Información.**

Una vez consolidada la fase anterior, el grupo SIG debe identificar cuáles son las necesidades de información existentes en función de los objetivos y el modelo del negocio de la organización, cuáles son las actividades y procesos que se desean eliminar y/o mejorar considerando los beneficios potenciales, y definir las estrategias que debe seguir para el proceso de implantación.

- **Definición del Alcance del SIG.**

Simultáneamente, con el análisis detallado de las necesidades de información, es fundamental establecer de manera muy clara cuál sería el alcance del sistema.

- **Conversión de Datos.**

Realizado el análisis de las necesidades de información y en paralelo con la definición del alcance, se definen los procedimientos para transformar los datos e información disponibles en formato digital.

- **Contratación de Servicio.**

La organización decidió contratar los servicios externos para realizar el proceso de conversión de datos, lo cual es la decisión más común en todos los procesos de implantación de SIG.

- **Carga de Datos.**

Esta fase consiste en la incorporación de los datos gráficos y atributivos al sistema ya en formato digital.

- **Evaluación y Selección Tecnológica.**

Después de haber definido el alcance y las necesidades de información en esta fase se diseña un documento basado en evaluaciones tecnológicas sobre la plataforma del sistema en cuanto a hardware y software, seleccionando por supuesto la que satisfaga las necesidades de información, y cumpla con los criterios establecidos de evaluación tecnológica.

- **Adquisición e Instalación de la Plataforma Tecnológica.**

Realizada la evaluación y selección de la plataforma SIG adecuada a las necesidades organizacionales de información, se compra la plataforma de hardware y software que sustentará al SIG.

- **Diseño Conceptual y Lógico de la Base de Datos.**

La fase de diseño conceptual y lógico de la base de datos puede iniciarse inmediatamente después del análisis de necesidades de información o en paralelo con la fase de evaluación y selección tecnológica.

- **Diseño Físico de la Base de Datos.**

Esta fase se inicia después de adquirir e instalar la plataforma tecnológica y generalmente se ejecuta en paralelo con el desarrollo de aplicaciones y programas.

- **Desarrollo de Programas y Aplicaciones.**

Esta fase puede ejecutarse en paralelo con el diseño físico de la base de datos, ya que ambas son necesarias para desarrollar el proyecto piloto sobre el área determinada para mostrar el sistema.

- **Proyecto Piloto.**

Después de diseñada la base de datos, los programas y posibles aplicaciones, el grupo SIG debe mostrar a la organización un modelo del SIG con datos reales, para ejemplificar las potencialidades y bondades del sistema, y comprobar si se ajusta a las necesidades de información establecidas.

- **Pruebas del SIG.**

Como resultado del proyecto piloto, esta fase consiste en probar el funcionamiento del sistema ya diseñado, pero mostrando información para el área de estudio seleccionada.

- **Adiestramiento a Usuarios.**

Ya diseñado y probado el sistema o parte de él según el alcance definido, se ejecuta un plan de adiestramiento a los usuarios potenciales del SIG.

- **Implantación del SIG.**

Esta fase consolida la instalación y puesta en marcha del SIG y el inicio de la operación del sistema por parte de los usuarios de la organización.

- **Operación y Mantenimiento.**

Esta es la fase final del proceso de implantación, en la cual se ponen en producción las facilidades que brinda el SIG y se obtienen los productos deseados tales como mapas a diferentes escalas, reportes, consultas que facilitan la toma de decisiones, eficacia en el análisis espacial, entre otros.

2.3. SISTEMAS DE CONTROL

Un sistema de control está definido como un conjunto de componentes que pueden regular su propia conducta o la de otro sistema con el fin de lograr un funcionamiento predeterminado.

2.3.1. Factores de Control

Existen cuatro factores que deben ser considerados al aplicar el proceso de control.

Cantidad

Tiempo

Costo

Calidad

Los tres primeros son de carácter cuantitativo y el último es eminentemente cualitativo.

El factor cantidad se aplica a actividades en la que el volumen es importante.

A través del factor tiempo se controlan las fechas programadas.

El costo es utilizado como un indicador de la eficiencia administrativa, ya que por medio de él se determinan las erogaciones de ciertas actividades.

La calidad se refiere a las especificaciones que deben reunir un cierto producto o ciertas funciones de la empresa.

2.3.1.1. Importancias Del Control

Establece medidas para corregir las actividades, de tal forma que se alcancen los planes exitosamente.

Se aplica a todo: a las cosas, a las personas y a los actos.

Determina y analiza rápidamente las causas que pueden originar desviaciones para que no vuelvan a presentarse en el futuro.

Localiza los sectores responsables de la administración, desde el momento en que se establecen medidas correctivas.

Proporciona información acerca de la situación de la ejecución de los planes, sirviendo como fundamento al reiniciarse el proceso de la planeación. Reduce costos y ahorra tiempo al evitar errores.

Su aplicación incide directamente en la racionalización de la administración y consecuentemente, en el logro de la productividad de todos los recursos de la empresa.

2.3.1.2. Tipos De Controles

Control preliminar. Este control tiene lugar antes de principiar operaciones e incluye la creación de políticas, procedimientos y reglas diseñadas para asegurar que las actividades planeadas serán ejecutadas con propiedad. La consistencia en el uso de las políticas y procedimientos es promovida por los esfuerzos del control.

Control concurrente. Este control tiene lugar durante la fase de la acción de ejecutar los planes e incluye la dirección, vigilancia y sincronización de las actividades, según ocurran.

Control de retroalimentación. Este tipo de control se enfoca sobre el uso de la información de los resultados anteriores, para corregir posibles desviaciones futuras del estándar aceptable.

2.3.1.3. Áreas Del Control

El control, actúa en todas las áreas y en todos los niveles de una empresa. Prácticamente todas las actividades de una empresa están bajo alguna forma de control o monitoreo.

Preferentemente debe abarcar las funciones básicas y áreas clave de resultados como:

Control de producción.

Control de calidad.

Control de inventarios.

Control de compras.

Control de mercadotecnia.

Control de ventas.

Control de finanzas.

Control de recursos humanos.

2.3.3.1. Gráfica De Punto De Equilibrio:

La gráfica de punto de equilibrio es un interesante método de control, ella ilustra la relación entre las ventas y los gastos de manera que indica qué volumen de ingresos cubre en forma exacta los gastos. Un volumen inferior de las ventas con respecto al punto de equilibrio ocasionarían pérdidas y uno superior trae mayor utilidad. El punto de equilibrio también puede venir expresado en unidades de artículos vendidos, porcentaje usado de la planta o en términos similares. Con frecuencia se confunde el análisis de punto de equilibrio con presupuesto variable.

2.3.3.2. La gráfica de gantt:

Aunque el concepto es simple, esta gráfica, que muestra las relaciones de tiempo entre los eventos de un programa de producción, se ha considerado como la innovación revolucionaria de la administración. Lo que Gantt identificó fue que las metas totales del programa deben considerarse como una serie de planes derivados (eventos) interrelacionados, que las personas pueden comprender y seguir.

Los progresos más importantes de este tipo de control reflejan este principio simple y también principios básicos de control como la elección de elementos estratégicos o críticos de un plan para verificarlos en forma cuidadosa.

2.3.3.3. Pert (Técnica De Revisión Y Evaluación De Programas):

La técnica del PERT se utiliza se utiliza en forma amplia en muchas operaciones y en particular en proyectos de construcción. Esta técnica capacita a los administradores para saber que tendrán problemas en áreas como costos o entregas a tiempo, a menos que tomen una acción ahora mismo. El Pert obliga a los administradores a planear debido a que es imposible hacer un análisis del tiempo y eventos sin planeación y sin observar como encajan entre sí los elementos. Además, impone la planeación en toda línea de autoridad, debido a que cada administrador subordinado debe planear el evento del que es responsable.

2.4. METODOLOGÍA OMT

La metodología OMT (Object Modeling Technique) fue creada por James Rumbaugh y Michael Blaha en 1991, mientras James dirigía un equipo de investigación de los laboratorios General Electric.

OMT es una de las metodologías de análisis y diseño orientados a objetos, más maduros y eficientes que existen en la actualidad. La gran virtud que aporta esta metodología es que le permite ser de dominio público y, en consecuencia, sobrevivir con enorme vitalidad. Esto facilita su evolución para acoplarse a todas las necesidades actuales y futuras de la ingeniería de software.

2.4.1. Fases de la Metodología OMT

Las fases que conforman a la metodología OMT son:

- **Análisis.** El analista construye un modelo del dominio del problema, mostrando sus propiedades más importantes. El modelo de análisis es una abstracción resumida y precisa de lo que debe de hacer el sistema deseado y no de la forma en que se hará. Los elementos del modelo deben ser conceptos del dominio de aplicación y no conceptos informáticos tales como estructuras de datos. Un buen modelo debe poder ser entendido y criticado por expertos en el dominio del problema que no tengan conocimientos informáticos.
- **Diseño del sistema.** El diseñador del sistema toma decisiones de alto nivel sobre la arquitectura del mismo. Durante esta fase el sistema se organiza en subsistemas basándose tanto en la estructura del análisis como en la arquitectura propuesta. Se selecciona una estrategia para afrontar el problema.
- **Diseño de objetos.** El diseñador de objetos construye un modelo de diseño basándose en el modelo de análisis, pero incorporando detalles de implementación. El diseño de objetos se centra en las estructuras de datos y algoritmos que son necesarios para implementar cada clase. OMT describe la forma en que el diseño puede ser implementado en distintos lenguajes (orientados y no orientados a objetos, bases de datos, etc.).
- **Implementación.** Las clases de objetos y relaciones desarrolladas durante el análisis de objetos se traducen finalmente a una implementación concreta. Durante la fase de implementación es importante tener en cuenta los principios de la ingeniería del software de forma que la correspondencia con el diseño sea directa y el sistema implementado sea flexible y extensible. No tiene sentido que utilicemos AOO y DOO de forma que potenciemos la reutilización de código y la correspondencia entre el dominio del problema y el sistema informático, si luego perdemos todas estas ventajas con una implementación de mala calidad.

2.4.1.1. Fase de Análisis.

El objetivo del análisis es desarrollar un modelo del funcionamiento del sistema. El modelo se expresa en términos de objetos y relaciones, el control dinámico de flujo y las transformaciones funcionales. El proceso de capturar los requerimientos y consultar con el solicitante debe ser continuo a través del análisis. A saber:

1. Contar con una descripción inicial del problema (enunciado del problema).
2. Construir un modelo de objetos. Modelo de objetos = diagramas del modelo de objetos + diccionario de datos.
3. Desarrollar un modelo dinámico. Modelo dinámico = diagramas de estado + diagrama global de flujo de eventos.
4. Construir un modelo funcional. Modelo funcional = diagramas de flujo de datos + restricciones.
5. Verificar, iterar y refinar los tres modelos:
 - Agregar al modelo de objetos operaciones clave que sean descubiertas durante la preparación del modelo funcional. No deben mostrarse todas las operaciones durante el análisis, sólo las más importantes.
 - Verificar que las clases, asociaciones, atributos y operaciones sean consistentes y completos al nivel seleccionado de abstracción. Comparar los tres modelos con el enunciado del problema y el conocimiento relevante al dominio y probar los modelos usando varios escenarios.
 - Desarrollar escenarios más detallados (incluyendo condiciones de error) como variaciones de los escenarios básicos, para verificar aún más los tres modelos.
 - Iterar los pasos anteriores según sea necesario para completar el análisis.

Documento de análisis = enunciado del problema + modelo de objetos + modelo dinámico + modelo funcional.

2.4.1.2. Fase de Diseño de sistemas.

Durante el diseño de sistemas, se selecciona la estructura de alto nivel del sistema. Existen varias arquitecturas canónicas que pueden servir como un punto de inicio adecuado. El paradigma orientado a objetos no introduce vistas especiales en el diseño del sistema, pero se incluye para tener una cobertura completa del proceso de desarrollo de software. Los pasos son:

1. Organizar el sistema en subsistemas.
 2. Identificar la concurrencia inherente al problema.
 3. Asignar subsistemas a procesadores y tareas.
 4. Escoger la estrategia básica para implantar los almacenamientos de datos en términos de estructuras de datos, archivos y bases de datos.
 5. Identificar recursos globales y determinar los mecanismos para controlar su acceso.
 6. Seleccionar un esquema para implantar el control del software:
 - Usar la ubicación dentro del programa para mantener el estado.
 - o Implantar directamente una máquina de estado.
 - o Usar tareas concurrentes.
-
1. Considerar las condiciones de frontera.
 2. Establecer prioridades de decisión sobre características deseables del producto de software.

Documento de diseño de sistemas = estructura de la arquitectura básica del sistema + las decisiones de estrategias de alto nivel.

2.4.1.3. Fase de Diseño de objetos.

Durante el diseño de objetos se elabora el modelo de análisis y se proporciona una base detallada para la implantación. Se toman las decisiones necesarias para realizar un sistema sin entrar en los detalles particulares de un lenguaje o base de datos particular. El diseño de objetos inicia un corrimiento en el enfoque de la orientación del mundo real del modelo de análisis hacia la orientación en la computadora requerida para una implantación práctica. Los pasos son:

1. Obtener las operaciones para el modelo de objetos a partir de los otros modelos:

- Encontrar una operación para cada proceso en el modelo funcional.
- Definir una operación para cada evento en el modelo dinámico, dependiendo de la implantación del control.

2. Diseñar los algoritmos para implantar las operaciones:

- Escoger los algoritmos que minimicen el costo de implementación de las operaciones.
- Seleccionar las estructuras de datos apropiadas para los algoritmos.
- Definir clases internas y operaciones nuevas según sea necesario.
- Asignar las responsabilidades para las operaciones que no están asociadas claramente con una sola clase.

3. Optimizar las rutas de acceso a los datos:

- Agregar asociaciones redundantes para minimizar los costos de acceso y maximizar la conveniencia.
- Reacomodar los cálculos para una mayor eficiencia.
- Guardar los valores derivados para evitar recalcular expresiones complicadas.

4. Implantar el control del software introduciendo el esquema seleccionado durante el diseño de sistemas.

5. Ajustar la estructura de clases para incrementar la herencia:

- Reacomodar y ajustar las clases y las operaciones para incrementar la herencia.
- Abstractar el comportamiento común de los grupos de clases.
- Usar delegación para compartir comportamiento donde la herencia sea semánticamente inválida.

6. Diseñar la implantación de las asociaciones:

- Analizar las travesías de las asociaciones.
- Implantar cada asociación como un objeto distinto o agregando atributos objeto-valor a una o ambas clases en la asociación.

7. Determinar la representación de los atributos de los objetos.

8. Empaquetar las clases y las asociaciones en módulos.

Documento de diseño = modelo de objetos detallado + modelo dinámico detallado + modelo funcional detallado.

2.4.2. Modelos que utiliza el OMT

La metodología OMT emplea tres clases de modelos para describir el sistema:

- **Modelo de objetos.** Describe la estructura estática de los objetos del sistema (identidad, relaciones con otros objetos, atributos y operaciones). El modelo de objetos proporciona el entorno esencial en el cual se pueden situar el modelo dinámico y el modelo funcional. El objetivo es capturar aquellos conceptos del mundo real que sean importantes para la aplicación. Se representa mediante diagramas de objetos.

- **Modelo dinámico.** Describe los aspectos de un sistema que tratan de la temporización y secuencia de operaciones (sucesos que marcan los cambios, secuencias de sucesos, estados que definen el contexto para los sucesos) y la organización de sucesos y estados. Captura el control, aquel aspecto de un sistema que describe las secuencias de operaciones que se producen sin tener en cuenta lo que hagan las operaciones, aquello a lo que afecten o la forma en que están implementadas. Se representa gráficamente mediante diagramas de estado.
- **Modelo funcional.** Describe las transformaciones de valores de datos (funciones, correspondencias, restricciones y dependencias funcionales) que ocurren dentro del sistema. Captura lo que hace el sistema, independientemente de cuando se haga o de la forma en que se haga. Se representa mediante diagramas de flujo de datos.

2.4.2.1. Modelo de Objetos

Esta es la parte principal de la Técnica para modelado ya que se fundamenta en la teoría de OO. La definición clara de las entidades que intervienen en el sistema es un paso inicial necesario para poder definir qué transformaciones ocurren en ellas y cuándo se producen estas transformaciones. Esta forma de pensar es inherente al paradigma de OO donde las clases y su jerarquía determinan el sistema.

Los diagramas de objetos permiten representar gráficamente los objetos, las clases y sus relaciones mediante dos tipos de diagramas: los diagramas de clases y los diagramas de casos concretos (instancias).

Los diagramas de clases describen las clases que componen el sistema y que permitirán la creación de casos concretos, los diagramas de casos concretos describen la manera en que los objetos del sistema se relacionan y los casos concretos que existen en el sistema de cada clase. En los diagramas que componen este modelo se pueden representar los siguientes elementos del sistema: objetos y clases, atributos, operaciones, y relaciones o asociaciones.

Clases y Objetos

Los objetos y sus componentes se representan gráficamente en OMT de forma que es posible obtener una idea de los elementos que intervienen en el sistema estudiando el modelo. Los elementos y sus características con representación gráfica son los siguientes:

- **Objetos.** Un objeto es, sencillamente, algo que tiene sentido en el contexto de la aplicación. Se definirá un objeto como un concepto, abstracción o cosa con límites bien definidos y con significado a efectos del problema que se tenga entre manos.
- **Clases.** Describe un grupo de objetos con propiedades (atributos) similares, con relaciones comunes con otros y con una semántica común.
- **Diagramas de objetos.** Proporcionan un anotación gráfica formal para el modelado de objetos, clases y sus relaciones entre sí, son útiles, tanto para el modelado abstracto como, para diseñar programas reales. Hay dos tipos de diagramas de objetos

-Diagrama de clases. Esquema, patrón o plantilla para describir muchas instancias de datos posibles.

-Diagrama de instancias. Describe la forma en que un cierto conjunto de objetos se relacionan entre sí.

- **Atributos.** Los objetos pertenecientes a una clase presentan características que en OMT se denominan atributos. Sin embargo, no se deben confundir los atributos, que son características que todos los objetos de una clase comparten, con otros objetos que pueden formar parte del objeto que estamos tratando.
- **Operaciones y métodos.** Del mismo modo que los objetos en OMT se pueden representar las operaciones que se realizan sobre ellos o que éstos realizan sobre otros objetos del sistema. Los objetos realizan acciones sobre otros objetos y definen acciones que se realizan sobre ellos mismos.

Los objetos de una misma clase comparten estas operaciones, aunque también pueden añadir otras nuevas que no se definan en su clase a medida que se especializa el objeto en otras subclases. También pueden redefinir las operaciones en estas especializaciones ignorando las definiciones realizadas en las superclases. Las operaciones pueden llevar implícito el objeto sobre el que se realizan o que realiza la acción, de forma que es posible tener una misma operación que se efectúe de manera distinta según el objeto sobre el que se aplique. La implementación de las operaciones para cada uno de los objetos diferentes (o subclases) se denomina método. Los métodos implementan en cada una de las clases de forma específica para los objetos que representa.

Enlaces y Asociaciones

Las relaciones entre clases determinan el comportamiento del sistema y constituyen una parte muy importante del mismo ya que mediante las relaciones definimos la forma en que los objetos se comunican, lo que también se conoce como comportamiento.

Además las relaciones tienen una serie de características que son de interés para el modelado del sistema.

- **Relaciones.** En OMT se identifican a través de enlaces: conexiones físicas o conceptuales entre casos concretos de objetos. Una asociación en OMT abstrae un conjunto de enlaces con una estructura y un significado comunes. Desde el punto de vista de la implementación, una asociación es un puntero que apunta desde un objeto a otro.
- **Multiplicidad.** Este término se encuentra relacionado con las asociaciones e indica el número de casos concretos de una clase que puede relacionarse con otro caso concreto. Las relaciones más frecuentes son las binarias, aunque pueden darse de cualquier orden.

Conceptos Avanzados de Enlaces y Asociaciones

- Atributos de los enlaces. Los enlaces así como los objetos pueden tener atributos, que son propiedades de los enlaces de una asociación.
- Modelado de una asociación en forma de clase. A veces resulta conveniente modelar las asociaciones como clases en lugar de cómo relaciones, cuando los enlaces pueden participar en asociaciones con otros objetos o están sometidos a operaciones.
- Clasificación. También podemos encontrar en una asociación de objetos que pertenecen a una clase con multiplicidad "muchos" que deben estar ordenados. Esta característica de los objetos es una restricción, ya que implica una condición que deben cumplir los elementos de la clase.
- Nombre de rol. Las asociaciones conectan clases u objetos que pertenecen a dichas clases, pero en ocasiones necesitamos restringir el conjunto de los objetos que se relacionan dentro de la clase. Mediante estas restricciones podemos especificar qué objetos se relacionan. Podemos conseguir este objetivo mediante los llamados nombres de rol. Un nombre de rol es un atributo derivado de una clase cuyo valor es un conjunto de objetos relacionados. Los nombres de rol se utilizan cuando las asociaciones se producen entre objetos de la misma clase ya que suelen producirse entre subconjuntos de esas clases. Una asociación binaria puede tener dos roles, uno por cada extremo de la asociación que son identificados por sus nombres.
- Calificación. Las asociaciones muchos a muchos y uno a muchos pueden ser calificadas mediante un elemento calificador que reduce el conjunto de objetos relacionados, indicando un subconjunto de la clase que se califica en la asociación. Las asociaciones que se pueden calificar son las de multiplicidad uno a muchos y muchos a muchos.
- Agregación. Las relaciones de agregación son para OMT formas de asociación del tipo "es parte de", como tales se definen entre una clase agregado y una clase componente y se indican con un rombo en la parte de la clase que actúa como continente.

Las relaciones de agregación se establecen en los llamados objetos compuestos que contienen otros objetos y éstos pueden ser de dos tipos: aquellos que tienen existencia física más allá del objeto agregado, y los que no pueden existir sin el objeto agregado.

Generalización y Herencia

En el paradigma de la orientación a objetos uno de los elementos más importantes es la herencia. La cualidad que permite que los objetos hereden las características (atributos) y las operaciones (métodos) dentro de una estructura jerárquica conlleva una serie de consecuencias de máxima relevancia a la hora de diseñar un sistema informático. Los objetos heredan un comportamiento que puede ser modificado y unas estructuras de datos de forma que se permite y se facilita la reutilización de las clases y del código que implementa sus funcionalidades.

Ambos conceptos van unidos: herencia y estructura jerárquica, de forma que la herencia se produce por la existencia de una estructura entre los componentes del sistema y la estructura se consigue en la implementación del código a través de la herencia en los lenguajes OO.

La herencia está íntimamente relacionada con la forma concreta en que un lenguaje implementa la generalización, que es un término más abstracto.

- La generalización es la relación que existe entre una clase y las subclases que se derivan de la misma. A partir de una versión "en bruto" se generan versiones más especializadas de la misma que añaden características y operaciones. La superclase es la versión general y las subclases son especializaciones de la superclase. Las generalizaciones pueden tener discriminadores que indican qué aspecto de la superclase está siendo utilizado para obtener subclases más concretas.
- Anulación. Al implementar la herencia nos encontramos en numerosas ocasiones que las subclases redefinen operaciones que ya han sido definidas en las superclases. Las razones para esta nueva implementación de operaciones que existen en las superclases son variadas, a veces

simplemente se le añaden nuevas acciones que van en consonancia con las nuevas características que añade la subclase; otras, se consigue optimizar las operaciones debido a que las subclases tienen características nuevas que lo permiten, y a veces se produce por una mala práctica de análisis donde no se prevén las operaciones de manera óptima.

Se han propuesto una serie de reglas a la hora de implementar la herencia para minimizar los errores y maximizar la reutilización de código:

- a. Las operaciones de consulta, aquellas que no modifican valores de atributos, se heredan por todas las subclases.
- b. Las operaciones de actualización, que modifican valores de atributos, se heredan por todas las subclases y se añaden las nuevas operaciones para aquellas que añadan atributos.
- c. Las operaciones de actualización que se realizan sobre atributos que tengan algún tipo de restricción o asociación, se bloquean para nuevas subclases.

- d. Las operaciones no pueden volver a definirse para hacer que se comporten de distinta manera de cara al exterior, es decir; todos los métodos concretos de una operación deben tener el mismo protocolo.
- e. Las operaciones se pueden refinar añadiendo comportamientos en las subclases.

Agrupación de entidades

Los elementos que hemos estudiado en el Modelo de Objetos se pueden agrupar para construir el modelo completo, así, las clases, las asociaciones y las generalizaciones forman lo que se denomina módulo y varios módulos forman el modelo de objetos. En un módulo no se deben repetir los nombres de las clases y de las asociaciones, aunque se puede hacer referencia a la misma clase dentro de

distintos módulos. También se definen las denominadas hojas que se utilizan para descomponer un Modelo de Objetos en unidades que podemos manejar. Una hoja es una parte de un módulo que podemos manejar con facilidad, sea en el formato que sea.

Modelo Avanzado de Datos

- Clases abstractas. En ocasiones puede ser de utilidad tener clases que definan propiedades y operaciones de forma general, dejando para sucesivos refinamientos la implementación concreta de las mismas. Una clase abstracta es precisamente una clase donde se introducen métodos y datos que se definirán en las subclases de la misma. Las clases abstractas siempre tienen que tener clases derivadas donde se especificarán las operaciones que no se hayan definido en la clase abstracta, nunca podrán tener objetos, se utilizan como clases bases o superclases de otras clases. La existencia de este tipo de clases facilita aún más la posibilidad de abstracción del modelo, dejando para posteriores refinamientos del problema la definición más exhaustiva de operaciones y datos. La implementación concreta de las clases abstractas depende del lenguaje OO. En OMT una clase abstracta se identifica cuando no tiene casos concretos (instancias) pero una subclase suya sí los tiene. Por ejemplo, una clase abstracta números podría definir una operación multiplicar que no se implementara hasta la definición de las subclases números enteros, números reales y números complejos, teniendo la información precisa en cada una de las subclases de la forma en que se multiplican los números.
- Herencia múltiple. La herencia múltiple es una característica que algunos sistemas OO poseen, mediante la cual es posible que una clase herede de varias superclases al mismo tiempo. Sin embargo, la herencia múltiple aumenta radicalmente la complejidad de los sistemas que la implementan ya que la búsqueda de las operaciones se dificulta cuando no se define en la clase derivada y hay que realizarla en las superclases.

- **Clave candidata.** No es más que un conjunto mínimo de atributos que define de forma única un objeto o enlace. Es decir, mediante una clave candidata tenemos definido el objeto o el enlace con una serie de atributos de forma que se distingue del resto de objetos o enlaces. Las claves candidatas son restricciones y, por tanto, se representan como tales en OMT. Para encontrar una clave candidata en una asociación donde intervienen más de dos clases, debemos definir qué combinaciones de clases o enlaces en la asociación de los elementos definen la tercera clase o enlace de forma única.
- **Restricciones.** El modelo de objetos contiene diferentes entidades como son los objetos, las clases, los atributos, los enlaces y las asociaciones. Cada una de estas entidades tiene una serie de características inherentes a su naturaleza dentro del sistema que estamos modelando. Las características definen su significado dentro del sistema y también sus limitaciones. Estas limitaciones se denominan en el Análisis restricciones. Las restricciones pueden ser muy complejas y en este caso no pueden estudiarse en el modelo de objetos, sino que se especificarán en el modelo funcional. Las que intervienen en el modelo de objetos se representan mediante llaves junto a la entidad a que se refieran. Cuando la restricción implica más de una clase, se indica mediante una flecha discontinua que une los elementos que se vean implicados en la restricción. Las asociaciones también pueden tener restricciones.

Construcción de un modelo de objetos

- Identificar las clases de objetos.
- Iniciar un diccionario de datos que contenga descripciones de clases, atributos y asociaciones.
- Agregar asociaciones entre clases.
- Agregar atributos a objetos y ligas.
- Organizar y simplificar las clases de objetos usando herencia.
- Probar las rutas de acceso usando escenarios e iterar los pasos anteriores según sea necesario.

- Agrupar las clases en módulos, basándose en "acoplamiento cercano" y función relacionada.

Notaciones del modelo de objetos

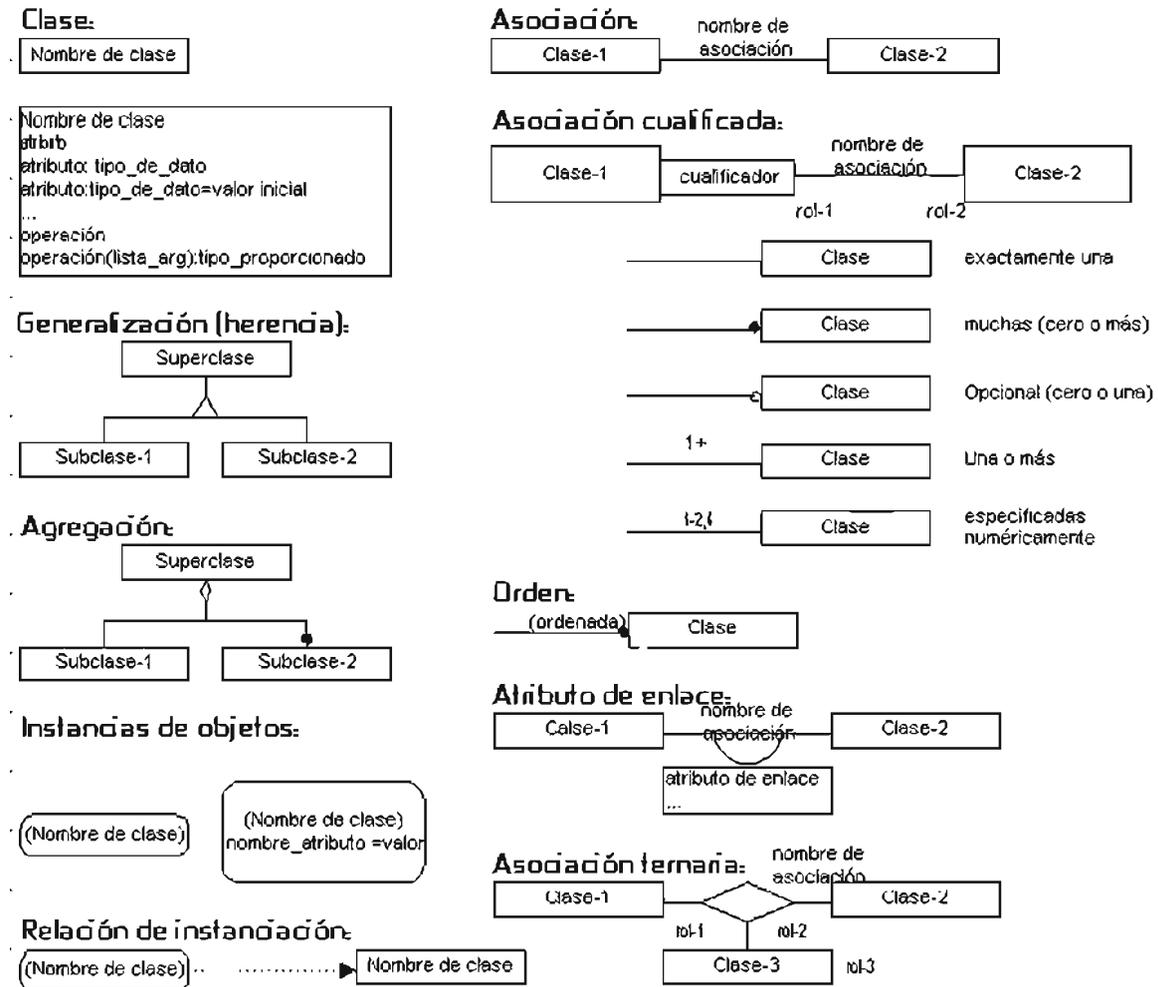
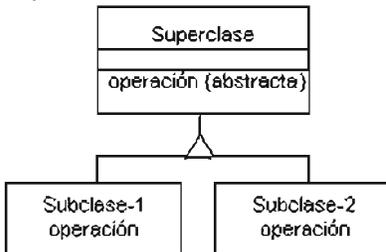


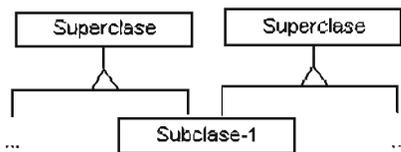
Figura 4: Notaciones del modelo de objetos

Notaciones del modelo avanzado de objetos

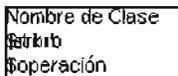
Operación abstracta



Herencia múltiple



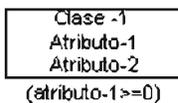
Atributos de clase y operaciones de clase



Propagación de operaciones



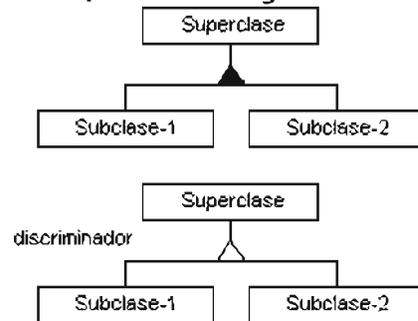
Restricciones en objetos



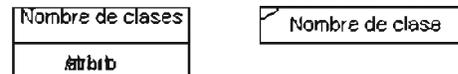
Asociación como clase



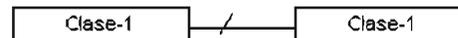
Propiedades de generalización



Atributo derivado Clase derivada



Asociación derivada



Restricciones entre asociaciones

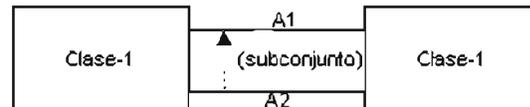


Figura 5: Notaciones del modelo de objetos avanzado

2.4.2.2. Modelo Dinámico

Los aspectos del sistema que están relacionados con el tiempo y con los cambios constituyen el modelo dinámico.

Los conceptos más importantes del modelado dinámico son los sucesos, que representan estímulos externos, y los estados, que representan los valores de los objetos. El diagrama de estados va a representar los sucesos y los estados que se dan en el sistema.

El modelo de objetos describe las posibles tramas de objetos, atributos y enlaces que pueden existir en un sistema. Los valores de los atributos y de los enlaces mantenidos por un objeto son lo que se denomina su estado. A lo largo del tiempo, los objetos se estimulan unos a otros, dando lugar a una serie de cambios en sus estados. Un estímulo individual proveniente de un objeto y que llega a otro es un suceso. La respuesta a un suceso depende del estado del objeto que lo recibe, y puede incluir un cambio de estado o el envío de otro suceso al remitente o a un tercer objeto.

La trama de sucesos, estados y transiciones de estados para una clase dada se puede abstraer y representar en forma de un diagrama de estados. El modelo dinámico consta de múltiples diagramas de estados, con un diagrama de estados para cada clase que posea un comportamiento dinámico importante, y muestra la trama de actividad para todo el sistema.

Sucesos y Estados

- Suceso. Un suceso, o evento, es algo que transcurre durante un período de tiempo. Un suceso puede preceder o seguir lógicamente a otro, o bien los dos sucesos pueden no estar relacionados. Se dice que dos sucesos que no tienen relación causal son concurrentes; no tienen efecto entre sí. Un suceso es una transmisión de información de dirección única entre un objeto y otro. No es como una llamada a subrutina, que proporciona un valor. En el modelo dinámico existe el concepto de clases de sucesos, que consiste en una estructura jerárquica, similar a la estructura de clases. Todo suceso aporta información de un objeto a otro. Los valores de datos aportados por un suceso son sus atributos.

- Escenarios y seguimiento de sucesos. Un escenario es una secuencia de sucesos que se produce durante una ejecución concreta de un sistema. El ámbito de un escenario es variable; puede incluir todos los sucesos del sistema, o que sean generados por ellos. Todo suceso transmite información de un objeto a otro. El primer paso para la construcción de escenario es identificar los objetos emisores y receptores de cada suceso.

La secuencia de sucesos y los objetos que intercambian sucesos se pueden mostrar ambos en un escenario mejorado que se denomina de seguimiento traza de sucesos. El tiempo aumenta desde arriba hacia abajo, aunque en el seguimiento de sucesos no importa la temporización exacta, sino que importa la secuencia de los procesos.

- Estados. Un estado es una abstracción de los valores de los atributos y de los enlaces de un objeto. Los conjuntos de valores se agrupan dentro del estado de acuerdo con aquellas propiedades que afecten al comportamiento del objeto. Un estado especifica la respuesta del objeto a los sucesos entrantes. La respuesta a un suceso recibido por un objeto puede variar cuantitativamente, dependiendo de los valores exactos de sus atributos, pero cualitativamente la respuesta es la misma para todos los valores dentro del mismo estado, y puede ser distinta para valores de distintos estados. La respuesta de un objeto a un suceso puede incluir una acción o un cambio de estado por parte del objeto. Un estado corresponde al intervalo entre dos sucesos recibidos por un objeto. Los sucesos representan puntos temporales; los estados representan intervalos de tiempo. Los estados suelen estar asociados con que el valor de un objeto satisfaga alguna condición. Al definir estados, ignoramos aquellos atributos que no afectan al comportamiento del objeto, y agrupamos en un único estado todas las combinaciones de valores de atributos y de enlaces que tienen una misma respuesta a los sucesos.

Por supuesto, todo atributo tiene algún efecto sobre el comportamiento, o no tendría sentido, pero es frecuente que algunos atributos no afecten a la trama de control, y que se pueda pensar en ellos como valores de parámetros dentro de un estado. Tanto los sucesos como los estados dependen del nivel de abstracción utilizado. Los estados se pueden caracterizar de diferentes maneras, pero normalmente cada estado tiene un nombre y una descripción en la que se indica en la situación en la que se encuentra el sistema.



Diagrama de Estados

Un diagrama de estados relaciona sucesos y estados. Cuando se recibe un suceso, el estado siguiente depende del actual, así como del suceso; un cambio de estado causado por un suceso es lo que se llama una transición. Un diagrama de estados es un grafo cuyos nodos son estados, y cuyos arcos dirigidos son transiciones rotuladas con nombres de sucesos.

El diagrama de estados especifica la secuencia de estados que causa una cierta secuencia de sucesos. Si un objeto se encuentra en un cierto estado y se produce un suceso cuyo nombre corresponda al de una de sus transiciones, entonces el objeto pasa al estado que se encuentra en el extremo de destino de la transición. Se dice que la transición se dispara. Si hay más de una transición que sale de un estado, entonces el primer suceso que se produzca dará lugar a que se dispare la transición correspondiente.

Si se produce un suceso que no tiene ninguna transición que salga del estado actual, entonces el suceso se ignora. Una secuencia de sucesos se corresponde con un camino a través del grafo.

Un diagrama de estados describe el comportamiento de una sola clase de objetos. Dado que todas las instancias de una clase tienen el mismo comportamiento, todas ellas comparten el mismo diagrama de estados, por cuanto todas ellas comparten las mismas características de clase. Pero dado que todo objeto posee sus propios

valores de atributos, cada objeto posee su propio estado, que es el resultado de la especial secuencia de sucesos que haya recibido. Todo objeto es independiente de los demás objetos, y procede a su paso.

Los diagramas de estados pueden representar ciclos vitales únicos o bien bucles continuos. Los diagramas de un solo uso representan objetos de duración finita y tienen estados iniciales y finales. Se entra en el estado inicial al crear el objeto; al entrar en el estado final estamos implicando la destrucción del objeto. Los diagramas de un solo uso son una "subrutina" del diagrama de estados, a la cual se puede hacer alusión desde distintos lugares en un diagrama de alto nivel.

El modelo dinámico es una colección de diagramas de estados que interactúan entre sí a través de sucesos compartidos.

Condiciones

Una condición es una función Booleana lógica que tiene a objetos como valores. Las condiciones se pueden utilizar como protecciones en las transiciones. Una transición con protección se dispara cuando se produce su suceso, pero sólo si la condición de protección es verdadera.

Operaciones

Los diagramas de estados tendrían muy poca utilidad si solamente describiesen tramas de sucesos. Una descripción de un objeto debe especificar lo que hace el objeto como respuesta a los sucesos.

Una actividad es una operación cuya realización requiere un cierto tiempo. Toda actividad está asociada a un estado. Entre las actividades se cuentan las operaciones continuas, tales como mostrar una imagen en una pantalla, así como las operaciones secuenciales que terminan por sí mismas después de un cierto intervalo de tiempo. Un estado puede controlar una actividad continua o una actividad secuencial que va avanzando hasta que termina o hasta que se produce un suceso que la hace finalizar prematuramente.

La anotación "hacer: X" indica que la actividad secuencial X comienza al entrar en ese estado, y se detiene cuando ha finalizado. Si un suceso da lugar a una transición que sale de ese estado antes de que haya finalizado la actividad, entonces, la actividad finaliza de forma prematura.

Una acción es una operación instantánea que va asociada a un suceso. Una acción representa a una operación cuya duración es insignificante en comparación con la resolución del diagrama de estados. Realmente, no hay operaciones que sean instantáneas, pero se modelan de esta manera aquellas operaciones de las que no nos importa su estructura interna.

Las acciones también pueden representar operaciones internas de control, tales como dar valores a atributos o generar otros sucesos. Estas acciones no tienen contrapartidas en el mundo real, sino que son mecanismos para estructurar el control dentro de una implementación.

Diagramas de Estados Anidados

Los diagramas de estados se pueden estructurar, para hacer posibles unas descripciones concisas de sistemas complejos. Las formas de estructurar máquinas de estados son similares a las formas de estructurar los objetos: la generalización y la agregación. La generalización es el equivalente a expandir las actividades anidadas. Permite describir una actividad empleando un nivel alto, y expandirla después en un nivel más bajo añadiendo detalles, de forma similar a las llamadas a procedimientos anidados. Además, la generalización permite que los estados y los sucesos se dispongan en jerarquías de generalización con herencia de estructuras y comportamientos comunes, de forma similar a la herencia de atributos y de operaciones en las clases. La agregación permite que el estado se descomponga en componentes ortogonales, con una interacción limitada entre ellos, de forma similar a una jerarquía de agregación de objetos. La agregación es el equivalente a la concurrencia de estados.

Los estados concurrentes suelen corresponderse con agregaciones de objetos, posiblemente de todo un sistema, que tengan partes que interactúen.

- Anidamiento de diagramas de estados. Una actividad de un estado se puede expandir en forma de diagrama de estados de nivel inferior, en el cual cada uno representará un paso de la actividad. Las actividades anidadas son diagramas de estados de un solo uso, con transiciones de entrada y de salida, parecidas a subrutinas.
- Generalización de estados. Un diagrama de estados anidados es en realidad una forma de generalización de estados. Un objeto que se encuentre en un estado del diagrama de alto nivel tiene que estar precisamente en uno de los estados del diagrama anidado. Los estados del diagrama anidado son todos ellos refinamientos del estado del diagrama de alto nivel. Los estados pueden poseer subestados que hereden las transiciones de sus superestados, del mismo modo que las clases poseen subclasses que heredan los atributos y operaciones de sus superclases.

Toda transición o acción que sea aplicable a un estado es aplicable también a todos sus subestados, a no ser que sea invalidada por una transición equivalente del subestado. Las transiciones de un superestado son heredadas por todos sus subestados.

Concurrencia

- Concurrencia de Agregación. Un modelo dinámico describe un conjunto de objetos concurrentes, cada cual con su propio estado y con su propio diagrama de estados. Los objetos de todo sistema son inherentemente concurrentes, y pueden cambiar de estado independientemente. El estado de todo el sistema no se puede representar mediante un solo estado de un único objeto; es el producto de los estados de todos los objetos que lo componen. En muchos sistemas además, el número de objetos puede cambiar dinámicamente. Un diagrama de estados para un subsistema es una colección de los mismos, uno por cada componente. La agregación implica concurrencia. El estado agregado corresponde a los estados combinados de todos los diagramas que componen el subsistema.

El estado agregado es un estado del primer diagrama y de todos los demás diagramas. En algunos casos los estados componentes interactúan. Las transiciones protegidas para un objeto pueden depender de que otro objeto se encuentre en un cierto estado. Esto permite la interacción entre diagramas de estados, manteniendo al mismo tiempo la modularidad.

- Concurrencia dentro de un objeto. La concurrencia dentro del estado de un único objeto surge cuando se puede descomponer el objeto en subconjuntos de atributos o de enlaces, cada uno de los cuales posee su propio subdiagrama.



Desarrollo de un modelo dinámico

- Preparar escenarios para las secuencias de interacción típicas.
- Identificar eventos entre objetos y preparar trazos de eventos para cada escenario.
- Preparar un diagrama de flujo de eventos para el sistema.
- Desarrollar un diagrama de estados para cada clase que tenga un comportamiento dinámico importante.
- Verificar que los eventos compartidos entre diagramas de estado sean consistentes y correctos.



Notaciones

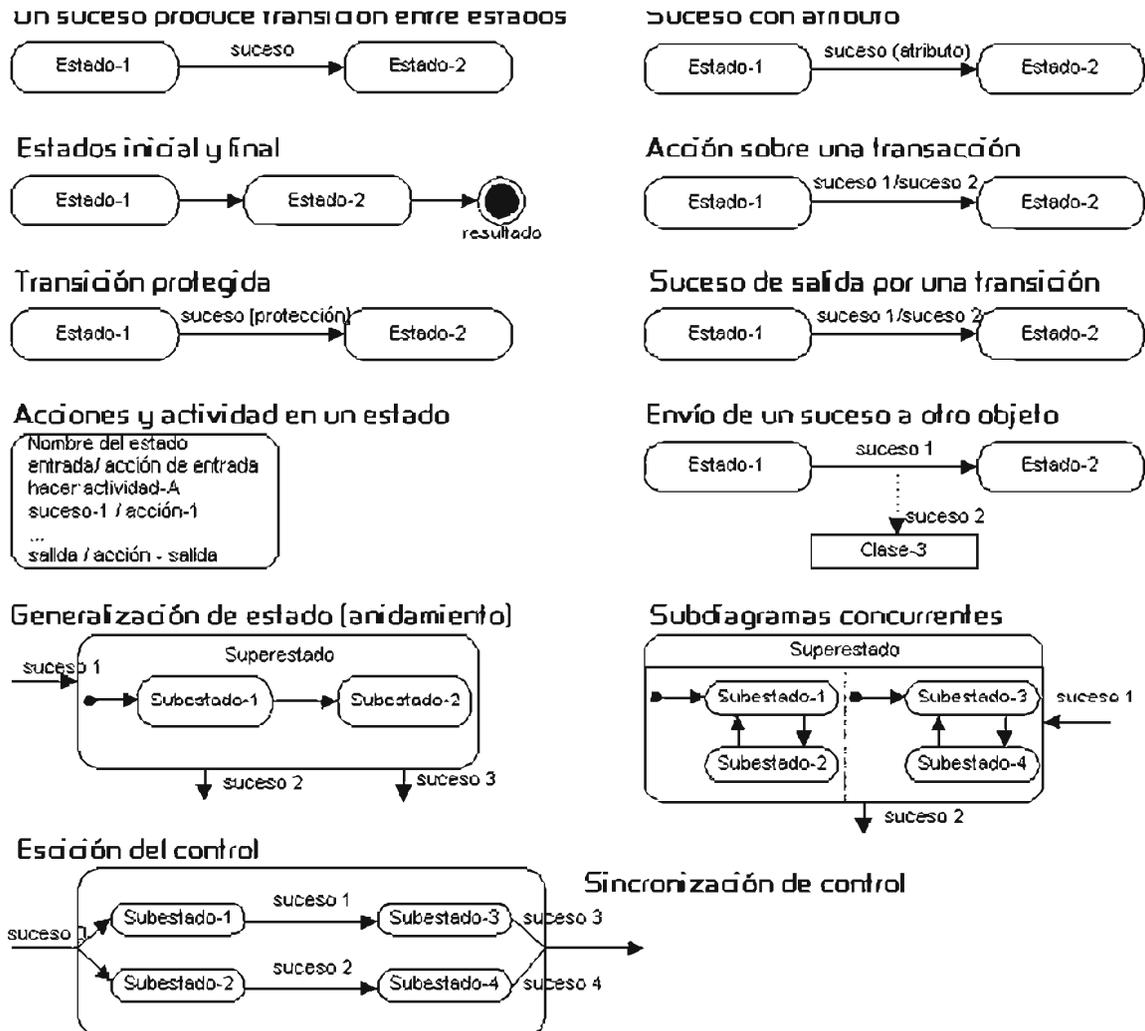


Figura 6: Notaciones del modelo dinámico

2.4.2.3. Modelo Funcional

El modelo funcional describe los cálculos existentes dentro del sistema siendo la tercera parte del modelado. Dentro del modelado del sistema, el modelo funcional especifica lo que sucede, el modelo dinámico cuándo sucede, y el modelo de objetos especifica a qué le sucede.

El modelo funcional muestra la forma en que se derivan los valores producidos en un cálculo a partir de los valores introducidos, sin tener en cuenta el orden en el cual se calculan los valores. Consta de múltiples diagramas de flujo de datos, que muestran el flujo de valores desde las entradas externas, a través de las operaciones y almacenes internos de datos hasta las salidas externas. También incluyen restricciones entre valores dentro del modelo de objetos. Los diagramas de flujo de datos no muestran el control ni tampoco información acerca de la estructura de los objetos; todo esto pertenece a los modelos dinámico y de objetos.

Diagramas de flujo de datos

El modelo funcional consta de múltiples diagramas de flujo de datos, que especifican el significado de las operaciones y de las restricciones. Un diagrama de flujo de datos (DFD) muestra las relaciones funcionales entre los valores calculados por un sistema, incluyendo los valores introducidos, los obtenidos, y los almacenes internos de datos. Un diagrama de flujo de datos es un grafo que muestra el flujo de valores de datos desde sus fuentes en los objetos mediante procesos que los transforman, hasta sus destinos en otros objetos. Un diagrama de flujo de datos no muestra información de control como puede ser el momento en que se ejecutan los procesos o se toman decisiones entre vías de datos alternativas; esta información pertenece al modelo dinámico. Un diagrama de flujo de datos no muestra la organización de los valores en objetos; esta información pertenece al modelo de objetos.

Un diagrama de flujo de datos contiene procesos que transforman datos, flujos de datos que los trasladan, objetos actores que producen y consumen datos, y de almacenes de datos que los almacenan de forma pasiva.

- **Procesos.** Un proceso transforma valores de datos. Los procesos del más bajo nivel son funciones puras, sin efectos laterales. Un grafo completo de flujo de datos es un proceso de alto nivel. Los procesos pueden tener efectos laterales si contienen componentes no funcionales, tales como almacenes de datos u objetos externos.

El modelo funcional no especifica de forma única el resultado de un proceso que tenga efectos laterales, solamente indica las posibles vías funcionales; no muestra la que realmente se recorrerá. Los resultados de estos procesos dependen del comportamiento del sistema, según se especifica en el modelo dinámico. El diagrama muestra solamente el patrón de entradas y salidas. El cálculo de valores de salida a partir de los de entrada también debe ser especificado. Un proceso de alto nivel se puede expandir en todo un diagrama de flujo de datos, de forma muy parecida a la manera en que se puede expandir una subrutina en otra subrutina de nivel inferior. Eventualmente, la recursividad finaliza, y los procesos atómicos deben describirse directamente, en lenguaje natural, mediante ecuaciones matemáticas, o por algún otro medio. Los procesos se implementan como métodos de operaciones que se aplican a clases de objetos. El objeto destino suele ser uno de los flujos de entrada, sobre todo si esa misma clase de objeto es también un flujo de salida.

- Flujo de datos. Un flujo de datos conecta la salida de un objeto o proceso con la entrada de otro objeto o proceso. Representa un valor de datos intermedio dentro de un cálculo que no es modificado por el flujo de datos. El mismo valor se puede enviar a varios lugares. En algunas ocasiones un valor de datos agregado se descompone en sus componentes, cada uno de los cuales va a un proceso diferente. La combinación de varios componentes en un valor agregado es justamente lo contrario. Cada flujo de datos representa un valor en algún momento del cálculo. Los flujos de datos internos al diagrama representan valores intermedios dentro de un cálculo, y o tienen necesariamente ningún significado en el mundo real. Los flujos en la frontera de un diagrama de flujo de datos son sus entradas y salidas. Estos pueden ser inconexos (si el diagrama es un fragmento de un sistema completo), o bien pueden estar conectados con objetos.
- Actores. Un actor es un objeto activo que controla el grafo de flujo de datos produciendo o consumiendo valores. Los actores están asociados a las entradas y salidas del grafo de flujo de datos.

En cierto sentido, los actores yacen en la frontera del grafo, pero hacen que concluya el flujo de datos como consumidores y productores de datos, así que en algunos casos se llaman sumideros. Las flechas entre el actor y el diagrama son las entradas y salidas del diagrama.

- Almacenes de datos. Un almacén de datos es un objeto pasivo dentro de un diagrama de flujo de datos que almacena datos para su posterior utilización, se limita a responder a solicitudes de almacenamiento y acceso a datos. Los almacenes de datos permiten acceder a los valores por un orden distinto a aquel en que fueron generados. Tanto los actores como los almacenes de datos son objetos. Se diferencian en que su comportamiento y utilización suelen ser distintos, aún cuando en un lenguaje orientado a objetos ambos pudieran ser implementados como objetos. Por otra parte, un almacén de datos se podría implementar como un fichero, y un actor en un dispositivo externo. Algunos flujos de datos son también objetos, aún cuando en muchos casos son valores puros, tales como enteros, que carecen de una identidad individual.
- Diagramas de flujo de datos anidados. Un diagrama de flujo de datos resulta muy útil para mostrar la funcionalidad de alto nivel de un sistema, y su descomposición en unidades funcionales más pequeñas. Un proceso se puede expandir en otro diagrama de flujo de datos. Todas las entradas y salidas del proceso lo serán también del nuevo diagrama, también puede tener almacenes de datos que no se muestren en el diagrama de nivel superior. Los diagramas se pueden anidar hasta una profundidad arbitraria, y todo el conjunto de diagramas anidados forman un árbol. Esto permite que cada nivel sea coherente y comprensible, y con toda la funcionalidad global que puede ser arbitrariamente compleja. Un diagrama que hace referencias a si mismo repetidamente representa un cálculo recursivo. El anidamiento de diagramas concluye con funciones sencillas que deben ser especificadas como operaciones.

- Flujos de control. Un diagrama de flujo de control muestra todas las posibles vías de computación para los valores. No muestra cuales son las vías que se ejecutan ni en qué orden. Las decisiones y la secuenciación son problemas de control, que forman parte del modelo dinámico. Una decisión afecta a si una o más funciones llagan incluso a ejecutarse, en lugar de proporcionarnos un valor. Aún cuando las funciones no poseen valores de entrada procedentes de estas funciones de decisión, a veces resulta útil incluirlas en el modelo funcional, para mostrar su dependencia con respecto a los datos. Esto se hace incluyendo flujos de control en el diagrama de flujo de datos.

Un flujo de control es un valor booleano que afecta a si un proceso es o no ejecutado. El flujo de control no es un valor de entrada al proceso en sí. Los flujos de control pueden ser útiles en ocasiones, pero duplican información del modelo dinámico.

Especificación de Operaciones

Los procesos de los diagramas de flujo deben ser implementados eventualmente como operaciones que se aplican a objetos. Todo proceso atómico del más bajo nivel es una operación. Los procesos de nivel superior también se pueden considerar operaciones, aún cuando una implementación pueda estar organizada de forma distinta del diagrama de datos que representa como consecuencia de la optimización. Toda operación se podrá especificar de diferentes maneras, entre las que están:

- Funciones matemáticas,
- Tablas de valores de entrada y salida,
- Ecuaciones que especifican la salida dependiendo de la entrada, condiciones previas y posteriores,
- Tablas de decisión,
- Pseudocódigo,
- Lenguaje natural.

La especificación de una operación se compone de dos partes. La primera de ellas es la que indica la interfaz de la operación: los argumentos que requiere y los valores que proporciona. La segunda, es la que explica la transformación de los valores de entrada para producir los valores de salida.

La especificación externa de una operación describe solamente cambios visibles fuera de ella. Durante la implementación de una operación, se pueden crear valores internos por conveniencia o por optimización. Algunos pueden incluso formar parte del estado interno de un objeto. El propósito de la especificación es indicar lo que debe hacer una operación lógicamente, y no como debe ser implementada. Por tanto, el estado del objeto en sí debe dividirse en información visible externamente e información privada, interna. Los cambios del estado interno de un objeto que no sean externamente visibles no modificarán su valor.

Las operaciones de acceso son operaciones que leen o escriben atributos o enlaces de un objeto. No es necesario enumerarlos o especificarlos durante el análisis. Durante el diseño, es necesario observar cuáles de las operaciones de acceso van a ser públicas, y cuáles privadas para esa clase de objetos. La razón para restringir el acceso no es una razón de corrección lógica, sino más bien para encapsular las clases con el objetivo de protegerlas contra errores y para permitir modificaciones en la implementación en un futuro. Las operaciones de acceso se derivan directamente de los atributos y asociaciones de la clase dentro del modelo de objetos.

Las operaciones no triviales se pueden desglosar en tres categorías: consultas, acciones y actividades. Una consulta es una operación que carece de efectos laterales en el estado visible externamente de cualquier objeto; es una función pura. Una consulta sin parámetros recibe el nombre de atributo derivado porque tiene la forma de un atributo.

Una acción es una transformación que posee efectos laterales sobre el objeto destino, o sobre otros objetos del sistema que resulten alcanzables desde él.

Las acciones no tienen una duración temporal: son instantáneas. Dado que el estado de un objeto queda definido por sus atributos y enlaces, todas las acciones deben de ser definibles en términos de actualizaciones de los atributos y enlaces básicos. Se puede definir una acción en términos del estado del sistema antes y después de la acción, no es necesario un componente de control.

Las acciones se pueden describir de diferentes maneras, incluyendo las ecuaciones matemáticas, árboles, tablas de decisión, enumeración de todas las posibles entradas, cálculo de predicados y lenguaje natural. Es importante que la especificación sea clara y no ambigua. Es necesaria una especificación formal. Hay varios elementos en la especificación: el nombre de la función, las entradas y salidas, las transformaciones de valores y las restricciones.

Una actividad es una operación hecha por o sobre un objeto que tiene una cierta duración temporal, por oposición a las consultas y acciones, que se consideran instantáneas. Una actividad tiene efectos colaterales como consecuencia de su duración temporal. Las actividades sólo tienen sentido para actores y objetos que generen operaciones propias, porque los pasivos son solamente depósitos de datos. Los detalles de una actividad son especificados por el modelo dinámico, así como por el modelo funciones, y no se pueden considerar tan sólo una transformación. En la mayoría de los casos, una actividad corresponde a un diagrama de estados del modelo dinámico.

Restricciones

Una restricción muestra la relación entre dos objetos al mismo tiempo o bien entre distintos valores del mismo objeto en instantes diferentes. Las restricciones se pueden expresar como una función total (un valor que es especificado completamente por otro) o como una función parcial (un valor que está restringido, pero no completamente especificado por otro valor). Las restricciones pueden aparecer en todas las clases del modelo.

Las restricciones de objetos especifican que algunos objetos dependen entera o parcialmente de otros objetos. Las restricciones dinámicas especifican relaciones entre los estados o sucesos de distintos objetos. Las restricciones funcionales especifican limitaciones aplicables a operaciones.

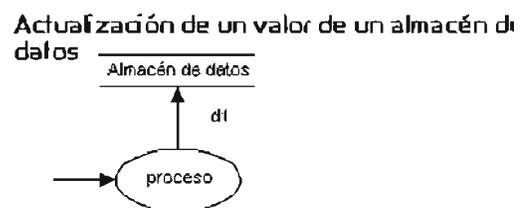
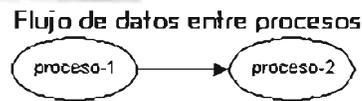
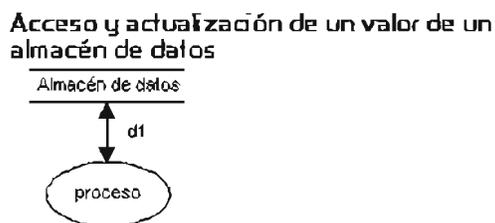
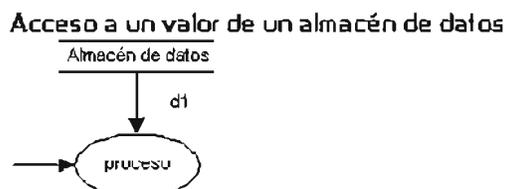
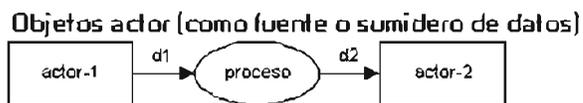
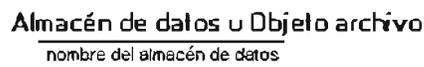
Una restricción entre valores de un objeto a lo largo del tiempo es lo que suele denominarse un invariante.

Construcción de un modelo funcional

- Identificar valores de entrada y salida.
- Usar diagramas de flujo de datos para mostrar dependencias funcionales.
- Describir las funciones.
- Identificar restricciones.
- Especificar criterios de optimización.

Notación

Figura 7: Notaciones del modelo funcional



2.4.3. Ventajas y Desventajas de la Metodología OMT

2.4.3.1. Ventajas

1. Proporciona una serie de pasos perfectamente definidos al desarrollador.
2. Tratamiento especial de la herencia.
3. Facilita el mantenimiento dada la gran cantidad de información que se genera en el análisis.
4. Es fuerte en el análisis

2.4.3.2. Desventajas

1. Hay pocos métodos para encontrar inconsistencias en los modelos.
2. Interacción de objetos no soportada explícitamente en ninguna herramienta gráfica.
3. Al ser un análisis iterativo es difícil de saber cuando comenzar con el diseño.
4. Es débil en el diseño

2.4.4. Aplicaciones de la Metodología OMT

Esta Tecnología puede ser aplicada en varios aspectos de implementación incluyendo:

- Archivos.
- Base de datos relacionales.
- Base de datos orientadas a objetos.
- Estructura de datos.
- Multimedia.
- Interactivas.
- Web.
- Cliente/servidor.
- Distribuidas.

Y en general, en prácticamente cualquier actividad de ingeniería que requiera hacer un análisis de un problema para poder resolver un problema.

2.5. TECNOLOGIA

La tecnología es un concepto amplio que abarca un conjunto de técnicas, conocimientos y procesos, que sirven para el diseño y construcción de objetos para satisfacer necesidades humanas.

En la sociedad, la tecnología es consecuencia de la ciencia y la ingeniería, aunque muchos avances tecnológicos sean posteriores a estos dos conceptos.

La palabra tecnología proviene del griego tekne (técnica, oficio) y logos (ciencia, conocimiento).

2.5.1. Red

Una red de computadoras es una interconexión de computadoras para compartir información, recursos y servicios. Esta interconexión puede ser a través de un enlace físico (alambrado) o inalámbrico.

Algunos expertos creen que una verdadera red de computadoras comienza cuando son tres o más los dispositivos y/o computadoras conectadas.

Para comunicarse entre sí en una red el sistema de red utiliza protocolos de red. Por extensión las redes pueden ser:

- Área de red local (LAN)
- Área de red metropolitana (MAN)
- Área de red amplia (WAN)
- Área de red personal (PAN)

Por relación funcional se clasifican en:

- Cliente/Servidor
- igual-a-igual (P2P)

Por topología:

- Red alambrada
- Red de anillo
- Red de bus
- Red de bus-estrella
- Red de estrella
- Red Mesh

Por estructura:

Red OSI

Red TCP/IP

Puede ser una intranet o extranet.

2.5.2. Internet

Internet fue desarrollado originariamente para los militares de Estados Unidos, y después se utilizó para el gobierno, investigación académica y comercial y para comunicaciones.

Es una combinación de hardware (ordenadores interconectados por vía telefónica o digital) y software (protocolos y lenguajes que hacen que todo funcione). Es una infraestructura de redes a escala mundial (grandes redes principales (tales como MILNET, NSFNET, y CREN), y redes más pequeñas que conectan con ellas) que conecta a la vez a todos los tipos de ordenadores.

Hay unos seis millones de ordenadores que utilizan Internet en todo el mundo y que utilizan varios formatos y protocolos internet:

- Internet Protocol (IP): protocolo que se utiliza para dirigir un paquete de datos desde su fuente a su destino a través de Internet.

- Transport Control Protocol (TCP): protocolo de control de transmisión, que se utiliza para administrar accesos.
- User Datagram Protocol (UDP): protocolo del datagrama del usuario, que permite enviar un mensaje desde un ordenador a una aplicación que se ejecuta en otro ordenador.

Internet tiene varios cuerpos administrativos:

- Internet Architecture Board, que supervisa tecnología y estándares.
- Internet Assigned Numbers Authority, que asigna los números para los accesos, etc.
- InterNIC, que asigna direcciones de Internet.

2.5.3. Sitio Web

En inglés website o web site, un sitio web es un sitio (localización) en la World Wide Web que contiene documentos (páginas web) organizados jerárquicamente. Cada documento (página web) contiene texto y o gráficos que aparecen como información digital en la pantalla de un ordenador. Un sitio puede contener una combinación de gráficos, texto, audio, vídeo, y otros materiales dinámicos o estáticos.

Cada sitio web tiene una página de inicio (en inglés Home Page), que es el primer documento que ve el usuario cuando entra en el sitio web poniendo el nombre del dominio de ese sitio web en un navegador. El sitio normalmente tiene otros documentos (páginas web) adicionales. Cada sitio pertenece y es gestionado y por un individuo, una compañía o una organización.

Como medio, los sitios web son similares a las películas, a la televisión o a las revistas, en que también crean y manipulan imágenes digitales y texto, pero un sitio web es también un medio de comunicación.

La diferencia principal entre un sitio web y los medios tradicionales es que un sitio web está en una red de ordenadores (Internet) y está codificado de manera que permite que los usuarios interactúen con él. Una vez en un sitio web, puedes realizar compras, búsquedas, enviar mensajes, y otras actividades interactivas.

2.6. HERRAMIENTAS

2.6.1. PHP (Hypertext Pre-processor)

Lenguaje de programación usado generalmente en la creación de contenidos para sitios web. Es un lenguaje interpretado especialmente usado para crear contenido dinámico web y aplicaciones para servidores, aunque también es posible crear aplicaciones gráficas utilizando la biblioteca GTK+.

Generalmente los scripts en PHP se embeben en otros códigos como HTML, ampliando las posibilidades del diseñador de páginas web enormemente. La interpretación y ejecución de los scripts PHP se hacen en el servidor, el cliente (un navegador que pide una página web) sólo recibe el resultado de la ejecución y jamás ve el código PHP.

Permite la conexión a todo tipo de servidores de base de datos como MySQL, Postgres, Oracle, ODBC, DB2, Microsoft SQL Server, Firebird y SQLite. PHP es una alternativa a otros sistemas como el ASP.NET/C#/VB.NET de Microsoft o a ColdFusion de Macromedia, a JSP/Java de Sun Microsystems, y a CGI/Perl. La ventaja con los de Microsoft o Macromedia es que es totalmente gratuito, no hay que pagar licencias.

2.6.2. MySQL

MySQL es un sistema de gestión de bases de datos (SGBD) multiusuario, multiplataforma y de código abierto.

MySQL pertenece a la compañía sueca MySQL AB, a la que le pertenece casi todos los derechos del código fuente.

La compañía desarrolla y mantiene el sistema, vendiendo soporte y servicios, como también las licencias para usar MySQL.

Uso de MySQL

MySQL es muy popular en aplicaciones web, y es componente de las plataformas LAMP, MAMP, WAMP, entre otras. MySQL suele combinarse con el popular lenguaje PHP.

Características de MySQL

- * MySQL está escrito en C y C++
- * Emplea el lenguaje SQL para consultas a la base de datos.
- * MySQL Server está disponible como freeware bajo licencia GPL.
- * MySQL Enterprise es la versión por suscripción para empresas, con soporte las 24 horas.
- * Trabaja en las siguientes plataformas: AIX, BSDi, FreeBSD, HP-UX, GNU/Linux, Mac OS X, NetBSD, Novell NetWare, OpenBSD, OS/2 Warp, QNX, SGI IRIX, Solaris, SunOS, SCO OpenServer, SCO UnixWare, Tru64, Microsoft Windows (95, 98, ME, NT, 2000, XP y Vista).

2.6.3. Arcview3.2



ArcView es la herramienta SIG más extendida en todo el mundo dadas sus avanzadas capacidades de visualización, consulta y análisis de información geográfica, además de las numerosas herramientas de integración de datos desde todo tipo de fuentes y herramientas de edición.

Figura 8.1.: Arcview

Por sí solo, ArcView permite la explotación de toda la información tanto en sistemas monousuario como en sistemas departamentales, pero es al integrarse en la arquitectura ArcGIS donde se consigue una solución global en el manejo de información geográfica y escalable según las necesidades del usuario.

Las tres aplicaciones de ArcView permiten acceder a una gran variedad de funcionalidad que abarca todos los campos de trabajo y procesamiento necesarios en un SIG:

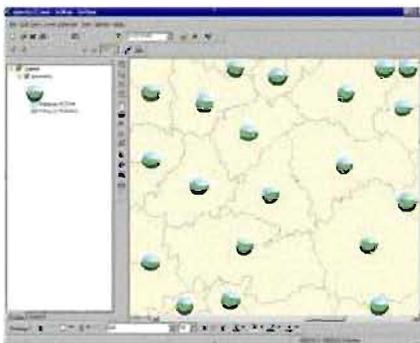
ArcMap: Permite visualizar, consultar, editar y realizar análisis sobre nuestros datos.

ArcCatalog: Constituye un avanzado explorador de datos geográficos y alfanuméricos, pensado para la visualización, administración y documentación de la información.

ArcToolbox: Es la herramienta que permite la realización de conversiones entre formatos, cambios de proyección y ajuste espacial.

La funcionalidad de ArcView se encuentra distribuida entre las tres aplicaciones que lo constituyen (ArcMap, ArcCatalog y ArcToolbox) y es el uso combinado de las tres aplicaciones lo que permite realizar, de manera muy intuitiva, tareas que van desde consultas hasta complejos análisis, incluyendo la gestión y edición de los datos, así como funciones de geoprocésamiento.

En particular ArcView permite realizar las siguientes funciones:



Explorar y administrar la información geográfica y alfanumérica en múltiples formatos.

Visualizar y consultar la información geográfica y alfanumérica.

Crear y mantener los metadatos de la información catalogada.

Figura 8.2.: Arcview

Crear el modelo de datos apropiado a las necesidades de cada usuario.

Acceder de manera inmediata a servicios de Internet a través del servidor de aplicaciones ArcIMS.

Realizar todo tipo de edición gráfica y alfanumérica gracias a su potente y avanzado entorno de edición.

Realizar tareas de análisis avanzado tanto de los datos geográficos como los alfanuméricos.

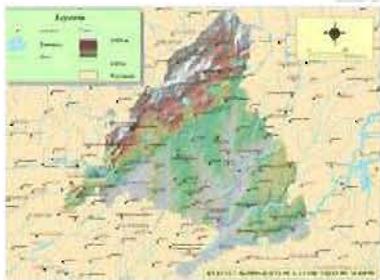
Producir cartografía de muy alta calidad.

Permitir una personalización rápida y sencilla del entorno de trabajo.

Acceder a funciones de geoprocésamiento mediante distintos entornos (ModelBuilder, Cuadros de diálogo, Línea de comandos y Entorno de scripts).

Gestionar las propiedades de etiquetado de un mapa desde un entorno centralizado.

Funcionalidad de ArcMap



Permite la visualización y consulta de varias capas de forma simultánea, gracias a herramientas como la ventana de aumento, la ventana de situación o los marcadores espaciales, así como la posibilidad de aplicar porcentajes de transparencia a las capas tanto vectoriales como raster.

Figura 8.3.: Arcview

ArcMap incorpora numerosas herramientas de edición de Geodatabases monousuario y ficheros Shapefile. Con estas herramientas se asegura la creación y el mantenimiento de la integridad de la información geográfica de forma rápida y sencilla.

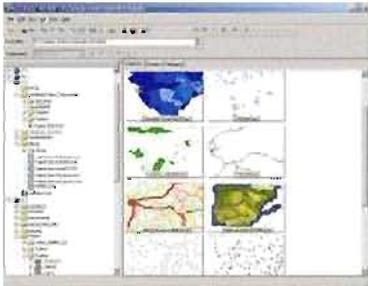
Mediante la topología implícita o topología de mapa se controlan las relaciones espaciales existentes entre los elementos elegidos, las cuales se mantienen durante el proceso de edición.

Junto con las operaciones de generación de zonas de influencia y geoprocésamiento, ArcMap incorpora innumerables funciones para el análisis SIG.

La multitud de librerías de simbología especializada, herramientas de etiquetado y plantillas hacen de ArcMap la aplicación ideal para la producción cartográfica de alta calidad.

Funcionalidad de ArcCatalog

Figura 8.4.: Arcview



Permite administrar, organizar, crear y previsualizar tanto datos geográficos como alfanuméricos.

Incorpora una potente herramienta para la creación y mantenimiento de metadatos, que sigue los estándares FGDC (Federal Geographic Data Committee) e ISO (International Organization for

Standardization), si bien estos estándares pueden ser ampliados mediante personalizaciones realizadas directamente por el usuario.

Funcionalidad de ArcToolbox

Permite el acceso a numerosas herramientas para conversión de datos a otros formatos, cambio de proyecciones y ajuste espacial.

Estas herramientas, organizadas temáticamente y mediante el empleo de intuitivos asistentes, permiten realizar dichas funciones de forma sencilla e inmediata.

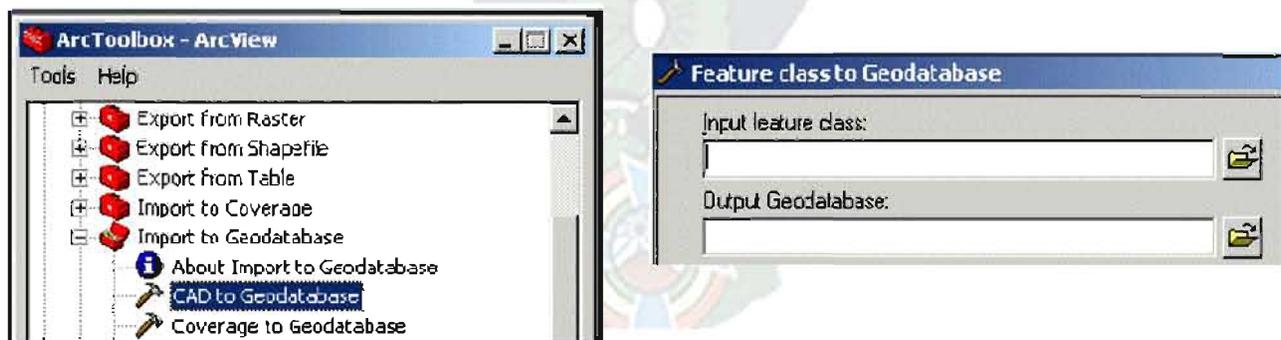


Figura 8.5.: Arcview

2.7. SEGURIDAD

Antes de entrar a fondo con el tema de seguridad, hay que tomar en cuenta que existen diferentes organizaciones, con diferentes tipos de información y por lo tanto con distintos requerimientos de seguridad.

En este caso veremos una parte del Libro Naranja para la seguridad de nuestro sistema.

El Libro Naranja es consecuencia de la creciente conciencia de la seguridad por parte del gobierno de los Estados Unidos y de la industria, ambos con la creciente necesidad de estandarizar el propósito y el uso de las computadoras por el gobierno federal.

El Libro Naranja define cuatro extensas divisiones jerárquicas de seguridad para la protección de la información. En orden creciente de confiabilidad se tienen:

D Protección Mínima

C Protección Discrecional

B Protección Obligatoria

A Protección Controlada

Cada división consiste en una o más clases numeradas, entre más grande sea el número se indica un mayor grado de seguridad.

2.7.1. Requisitos Fundamentales de la Seguridad en Cómputo

Cualquier discusión sobre seguridad en cómputo necesariamente empieza con una definición de sus requisitos básicos, es decir, realmente qué significa el llamar a un sistema informático "seguro". En general, un sistema seguro controlará, a través del uso de características específicas de seguridad, el acceso a la información, de forma tal, que solamente los individuos autorizados correctamente, o los procesos que obtienen los permisos adecuados, tendrán acceso para leer, escribir, crear, modificar o eliminar la información.

Se tienen seis requisitos fundamentales, los cuales se derivan de esta declaración básica; cuatro de ellos parten de la necesidad de proporcionar un control de acceso a la información y los dos restantes de cómo puede obtenerse una seguridad demostrable, logrando así un sistema informático confiable.

2.7.1.1. Política de Seguridad

Debe existir una política de seguridad explícita y bien definida reforzada por el sistema. Identificados los eventos y los objetos, debe haber un conjunto de reglas que son utilizadas por el sistema para determinar si un evento dado se puede permitir para acceder a un objeto específico.

Los sistemas informáticos de interés deben hacer cumplir una política obligatoria de seguridad, en la cual puedan implementarse eficientemente reglas del acceso para manejo de información sensible (p.e. clasificaciones) estas reglas deben de incluir requisitos tales como: Ninguna persona que carezca de los permisos apropiados obtendrá el acceso a la información clasificada. Además, los controles de seguridad discrecional se requieren para asegurar que solamente los usuarios o los grupos seleccionados de usuarios puedan obtener el acceso a los datos.(p.e., basarse en una necesidad de conocimientos específicos).

2.7.1.2. Marcas

El control de acceso por etiquetas debe de estar asociado a los objetos. Para controlar el acceso a la información almacenada en una computadora, según las reglas de una política obligada de seguridad, debe de ser posible el marcar cada objeto con uno una etiqueta que identifique confiablemente el nivel de la sensibilidad del objeto (p.e., clasificación), y/o los modos de obtener acceso y acordar quien puede tener acceso potencial al objeto.

2.7.1.3. Identificación

Los eventos individuales deben de ser identificados. Cada acceso a la información debe ser registrado teniendo como base quién está teniendo acceso a la información y qué autorización posee para ocupar cierta clase de información. La información de la identificación y la autorización debe ser administrada con seguridad por el sistema informático y asociar cierta seguridad a cada elemento activo que realice una cierta acción relevante en el sistema.

2.7.1.4. Responsabilidad

Las auditorias de la información deben ser selectivamente guardadas y protegidas de las acciones que puedan afectar la seguridad y de esta forma poder rastrear al responsable. Un sistema confiable debe tener la capacidad de registrar la ocurrencia de acontecimientos relevantes sobre seguridad en una bitácora auditable. Además de poseer la capacidad de seleccionar los eventos a auditar para ser registrados, es necesario para reducir al mínimo el costo de la revisión y permitir un análisis eficiente. Este tipo de registros o bitácoras, deben de estar protegidos contra la modificación y la destrucción no autorizada, y deben permitir la detección y la investigación posterior de las violaciones de seguridad.

2.7.1.5. Aseguramiento

El sistema informático debe contener los mecanismos de hardware/software que puedan ser evaluados independientemente para proporcionar una seguridad suficiente que el sistema haga cumplir los requisitos 1 a 4 mencionados anteriormente.

Para asegurar que los requisitos de política de seguridad, marcas, identificación, y responsabilidad de la seguridad son hechos cumplir por un sistema de cómputo, deben ser identificados como una colección unificada de hardware y software que controle y ejecute esas funciones.

Estos mecanismos son típicamente incluidos en el sistema operativo y se diseñan para realizar las tareas asignadas de una manera segura. La base para confiar en tales mecanismos del sistema operativo, radica en su configuración operacional, la cual debe ser claramente documentada a fin de hacer posible el examinar independientemente los eventos para su evaluación.

2.7.1.6. Protección Continúa

Los mecanismos de seguridad que hacen cumplir estos requisitos básicos, se deben de proteger continuamente contra cambios no autorizados o modificaciones que traten de alterarlos. Ningún sistema de cómputo puede ser considerado verdaderamente seguro si los mecanismos que hacen cumplir las Políticas de seguridad, están sujetos a modificaciones no autorizadas. El requisito de protección continua tiene implicaciones directas a través del ciclo de vida de los sistemas.

2.7.2. Rutas Seguras

Una ruta segura proporciona un medio libre de errores, por el cual un usuario (típicamente una Terminal o un a estación de trabajo) puede comunicarse directamente con un TCB (Computadora Confiable Base) sin interactuar con el sistema a través de aplicaciones inseguras (y posiblemente poco fiables) y capas del sistema operativo.

2.7.3. Auditoria

La auditoria es el registro, examen y revisión de las actividades relacionadas con la seguridad en un sistema confiable. Una actividad relacionada con la seguridad es cualquier acción relacionada con el acceso de usuarios, o acceso a objetos. En términos de auditoria, algunas actividades son llamadas frecuentemente eventos, y una auditoria interna se llama algunas veces eventos logging.

Los eventos típicos incluyen

- Logins (exitosos o fallidos)
- Logouts
- Accesos a sistemas remotos
- Operaciones de archivos, apertura, renombrar, eliminación.
- Cambios en los privilegios y atributos de seguridad (cambiar en un archivo la etiqueta sensitiva o el nivel del pase de un usuario).

2.7.4. Recuperación Confiable

La recuperación confiable asegura que la seguridad no ha sido violada cuando se cae un sistema o cuando cualquier otra falla del sistema ocurre. La recuperación confiable actualmente involucra dos actividades: prepararse ante una falla del sistema y recuperar el sistema. La principal responsabilidad en preparación es respaldar todos los archivos del sistema crítico con una base regular. El procedimiento de recuperación puede ser con mucho, esforzarse por restaurar solo un día o dos de procesamiento de información.

2.7.5. Diseño de Documentación

Es un requerimiento formidable para todos los desarrolladores de sistemas. La idea de diseñar documentación es documentar internamente el sistema (o lo más básico del TCB) hardware, firmware y software. El objetivo del diseño de documentación es que “la filosofía del fabricante sobre protección y cómo esta filosofía es trasladada dentro del TCB” Una tarea clave que define los límites del sistema y distingue claramente entre cuales porciones del sistema son relevantemente seguras y cuales no.

Las dos mayores metas del diseño de documentación son: el probar al equipo de evaluación que el sistema cumple con el criterio de evaluación y el auxiliar al equipo de diseño y desarrollo para ayudar a definir las políticas del sistema de seguridad y como hacer que las políticas se lleven a cabo durante la implementación.



CAPITULO III

MARCO APLICATIVO

MARCO APLICATIVO

En este capítulo se desarrollará todas las técnicas descritas en el anterior capítulo, utilizando la metodología OMT como base de desarrollo.

3.1. Fase de Análisis

3.1.1. Enunciado del Problema

El problema general sería que no cuenta con un sistema independiente y necesario para realizar sus reportes estadísticos, control de rabia a nivel nacional y sus publicaciones al público en general.

3.1.2. Modelo de Objetos

El objetivo del modelo de objetos es capturar aquellos conceptos del mundo real que sean importantes para la aplicación, representándose mediante diagramas de objetos.

3.1.2.1. Identificación de Objetos y Clases

Identificamos las siguientes clases:

Figura 9: Identificación de objetos y clases



RABIA
Aquí se almacena todos los datos de los informes epidemiológicos de la rabia.
ATRIBUTOS
COD_LAB NOM_LAB SEM_EPI FECHA DEPTO PROVINCIA MUNICIPIO LOCALIDAD GER_RED DIRECCION ENTREGADO ESPECIE MORDIDOS VACUNA PER_AGREDIDA LU_MORDEDURA EDAD SEXO RESULTADO

Tabla 1: Objeto Rabia

MEDICAMENTOS
Aquí se almacena todos los datos de las vacunas y sueros que se envían a toda Bolivia.
ATRIBUTOS
COD_SEDES SEDES A_NOM TIPO CANTIDAD FECHA

Tabla 2: Objeto Medicamentos

CONTROL_MEDICAMENTOS Aquí se almacena los informes estadísticos de las medicinas.
ATRIBUTOS
COD_CONTME NOM_PROM PRO_GRAL PRO_ANUAL PRO_MEN PRO_SEM PRO_SEM

Tabla 3: Objeto Control medicamentos

DEPARTAMENTO Aquí se almacena los datos necesarios del departamento.
ATRIBUTOS
COD_DEP NOM_DEP POBLACION

Tabla 4: Objeto departamento

PROVINCIAS Aquí se almacena los datos necesarios de la provincia.
ATRIBUTOS
COD_PRO NOM_PRO NOM_DEP POBLACION

Tabla 5: Objeto provincias

MUNICIPIOS Aquí se almacena los datos necesarios de los municipios.
ATRIBUTOS
COD_MUNI NOM_MUNI NOM_PRO NOM_DEP POBLACION

Tabla 6: Objeto municipios

LOCALIDADES Aquí se almacena los datos necesarios de las localidades.
ATRIBUTOS
COD_LOC NOM_LOC NOM_COM POBLACION

Tabla 7: Objeto localidades

USUARIOS Aquí se almacena todas las personas que harán uso de la herramienta grafica.
ATRIBUTOS
COD_US NOM_US AP_US AM_US CI COD_TIPUS

Tabla 8: Objeto usuarios

TIPOS_USUARIOS Aquí se almacena todos los tipos de usuarios que puedan existir, de acuerdo al tipo de acceso que tenga.
ATRIBUTOS
COD_TIPUS DESCRIP_TIPUS

Tabla 9: Objetos tipos_usuarios

CONTROL_USUARIOS Aquí se registra al usuario y contraseña de cada usuario.
ATRIBUTOS
COD_US NOM_US CONTRA_US

Tabla 10: Objeto control_usuarios

Tabla 11: Objeto control_usuarios_accesos

CONTROL_USUARIOS_ACCESOS Aquí se registra las ventanas a las cuales tiene acceso un usuario.
ATRIBUTOS
COD_US COD_VEN COD_ESTVEN

3.1.2.2. Diccionario de datos

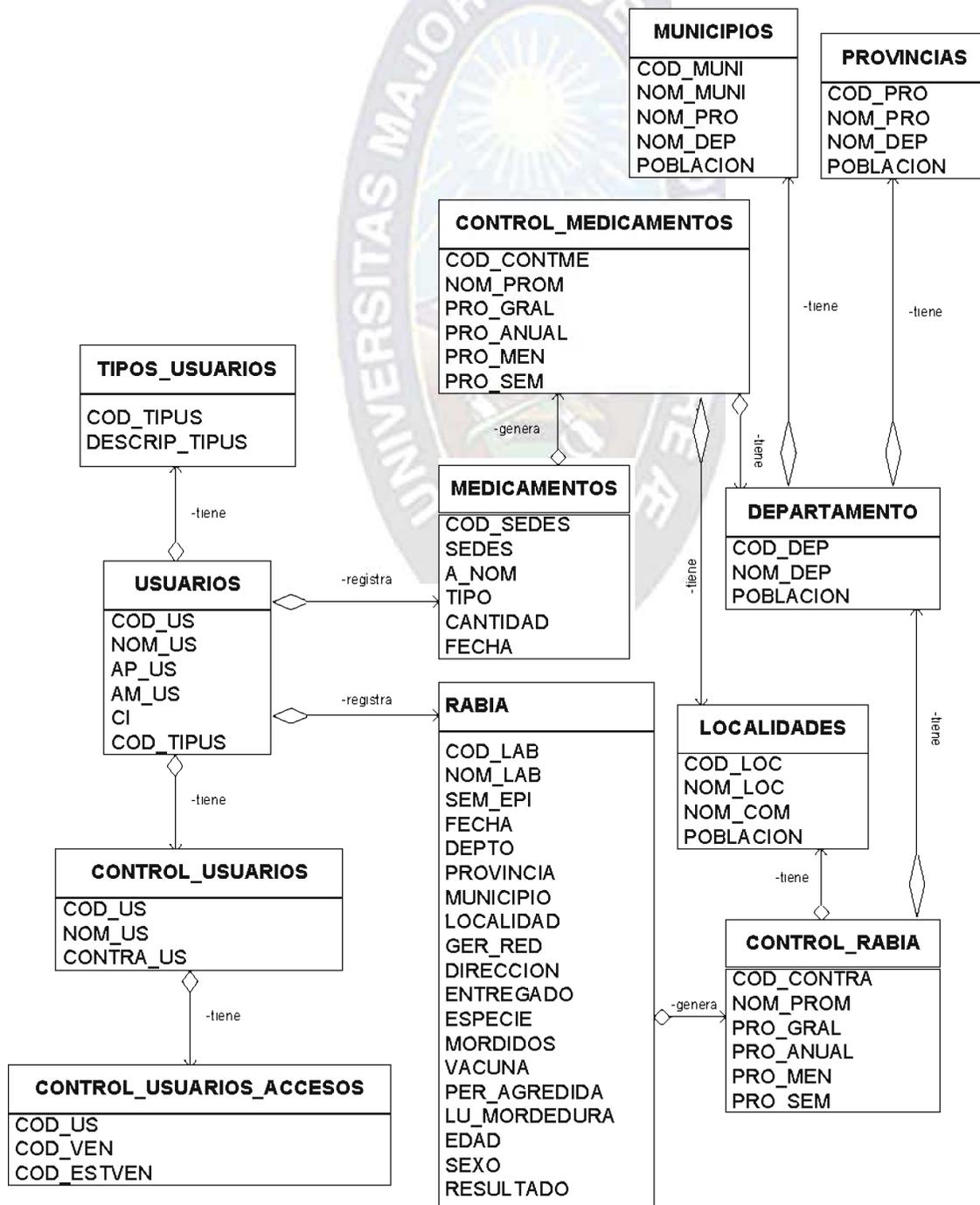
A_NOM	El nombre completo de la persona a la cual se manda las medicinas
AM_US	Apellido materno del usuario
AP_US	Apellido paterno del usuario
CANTIDAD	La cantidad de medicinas que se manda a cada SEDES
CI	La cedula de identificación del usuario
COD_CONTME	Código del control de las medicinas
COD_CONTRA	Código del control de la rabia
COD_DEP	Código de departamento
COD_ESTVEN	Código del estado de la ventana
COD_LAB	Código de laboratorio
COD_LOC	Código de localidad
COD_MUNI	Código de municipio
COD_PRO	Código de provincia
COD_SEDES	Código de las SEDES
COD_TIPUS	Código del tipo de usuario
COD_US	Código del usuario
COD_VEN	Código de la ventana
CONTRA_US	Contraseña del usuario
DEPTO	Nombre de departamento
DESCRIP_TIPUS	Descripción del tipo de usuario
DIRECCION	Dirección donde ocurrió la mordedura
EDAD	Edad de la persona que sufrió la mordedura
ENTREGADO	Nombre completo de la persona que entrego el formulario
ESPECIE	Especie del animal agresor
FECHA_R	Fecha del llenado del Formulario
FECHA_M	Fecha del llenado del Formulario de medicamentos
GER_RED	Gerencia de red de salud
LOCALIDAD	Nombre de localidad
LU_MORDEDURA	Lugar en el cuerpo de la persona que fue mordida
MORDIDOS	Especie del ser que fue mordido
MUNICIPIO	Nombre de municipio
NOM_COM	Nombre de la comunidad
NOM_DEP	Nombre de departamento
NOM_LAB	Nombre de laboratorio
NOM_LOC	Nombre de localidad
NOM_MUNI	Nombre de municipio
NOM_PRO	Nombre de provincia
NOM_PROM	Nombre del promedio que quiera sacar
NOM_US	Nombre del usuario
PER_AGREDIDA	Nombre completo de la persona agredida
POBLACION	Población de cierto sector
PRO_ANUAL	Promedio anual
PRO_GRAL	Promedio general
PRO_MEN	Promedio mensual
PRO_SEM	Promedio semanal
PROVINCIA	Nombre de provincia
RESULTADO	Resultado de análisis de rabia (positivo/negativo)

SEDES	Servicios Departamentales de Salud
SEM_EPI	Numero de la semana epidemiológica
SEXO	El sexo de la persona agredida
TIPO	Tipo de medicamento enviado
VACUNA	Nombre del medicamento enviado

3.1.2.3. Construcción del diagrama de clases básico

Construimos una versión básica del diagrama de clases que muestra el comportamiento que va a tener el sistema.

Figura 10: Diagrama de clases básico



3.1.3. Modelo Dinámico

El objetivo del modelo dinámico, es capturar el control de un sistema que describe las secuencias de operaciones que se producen sin tener en cuenta lo que hagan las operaciones.

3.1.3.1. Preparar escenarios para las secuencias de iteración típicas

A continuación veremos los escenarios entre el usuario y el sistema.

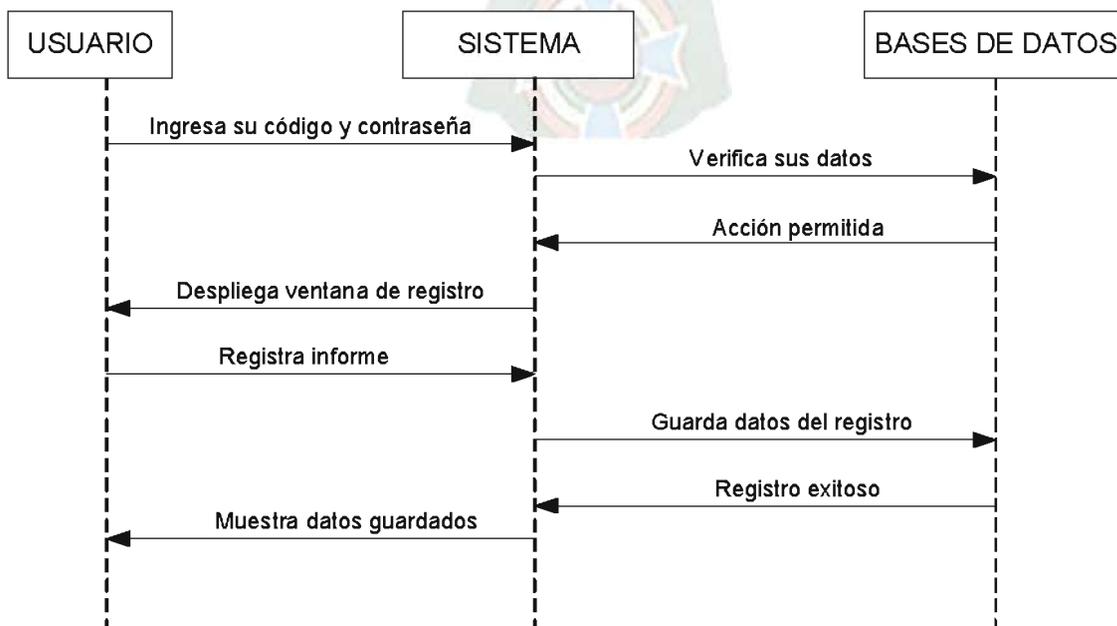
Escenario de secuencias de interacción para:
Registro del Informe Epidemiológico de Diagnostico de Rabia
<ul style="list-style-type: none">• El usuario ingresa al sistema con su código y contraseña, el sistema verifica sus datos para que pueda ingresar.• El usuario registra el informe que realizan semanalmente, el sistema verifica que los datos sean llenados correctamente.• El sistema despliega los datos llenados.
Escenario de secuencias de interacción para:
Momificación del Informe Epidemiológico de Diagnostico de Rabia
<ul style="list-style-type: none">• El usuario ingresa al sistema con su código y contraseña, el sistema verifica sus datos para que pueda ingresar.• El usuario escoge cierto informe para modificar, el sistema envía sus datos para que el mismo pueda ser modificado.• El usuario modifica el informe, el sistema hace los cambios y los guarda.• El sistema despliega los nuevos datos modificados.
Escenario de secuencias de interacción para:
Control del Informe Epidemiológico de Diagnostico de Rabia
<ul style="list-style-type: none">• El usuario pide reportes estadísticos según la clase y tiempo que necesite, el sistema le devuelve esos reportes y los guarda.• El usuario pide datos de reportes estadísticos para actualizar mapas geográficos, el sistema exporta esos datos a Excel.

Escenario de secuencias de interacción para:
Registro de Vacunas y Sueros
<ul style="list-style-type: none"> • El usuario ingresa al sistema con su código y contraseña, el sistema verifica sus datos para que pueda ingresar. • El usuario registra las vacunas y sueros, el sistema verifica que los datos sean llenados correctamente. • El sistema despliega los datos llenados.
Escenario de secuencias de interacción para:
Modificación de Vacunas y Sueros
<ul style="list-style-type: none"> • El usuario ingresa al sistema con su código y contraseña, el sistema verifica sus datos para que pueda ingresar. • El usuario escoge cierto registro para modificar, el sistema envía sus datos para que el mismo pueda ser modificado. • El usuario modifica el informe, el sistema hace los cambios y los guarda. • El sistema despliega los nuevos datos modificados.

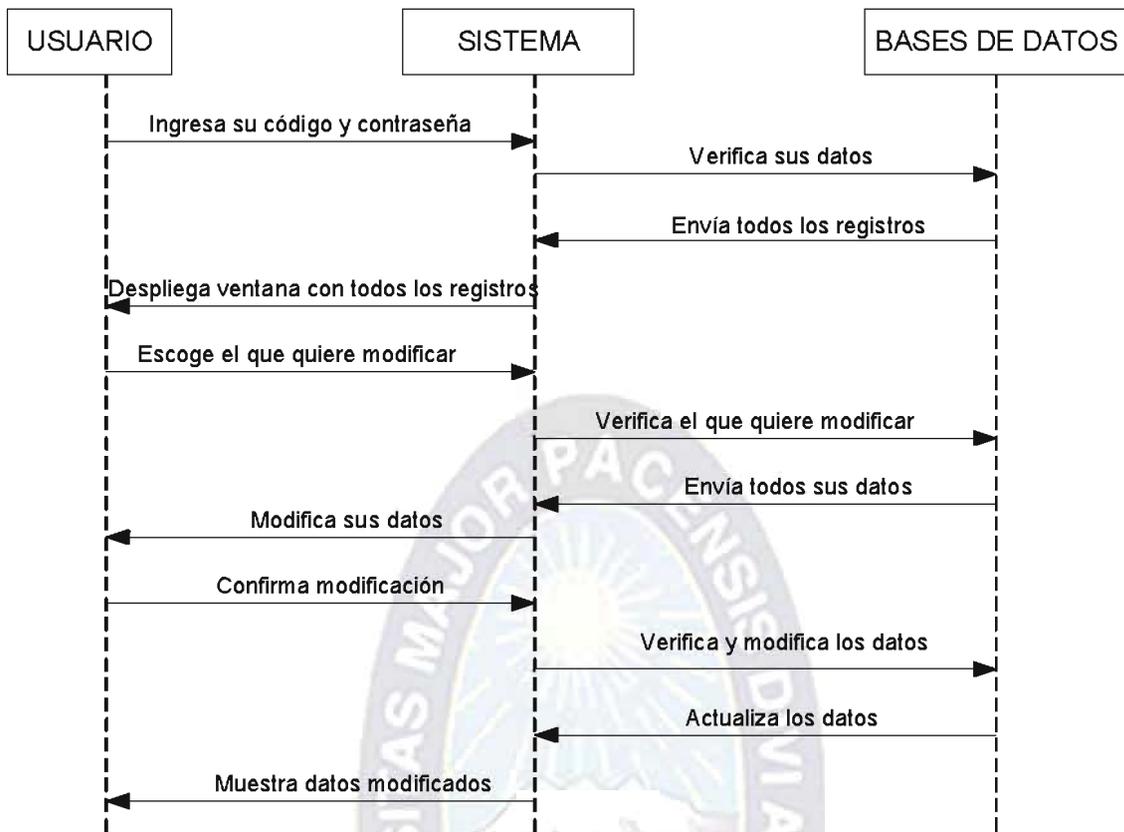
3.1.3.2. Preparar trazos de eventos para cada escenario

Mostraremos el comportamiento del sistema en cada escenario, los cuales tienen un comportamiento dinámico.

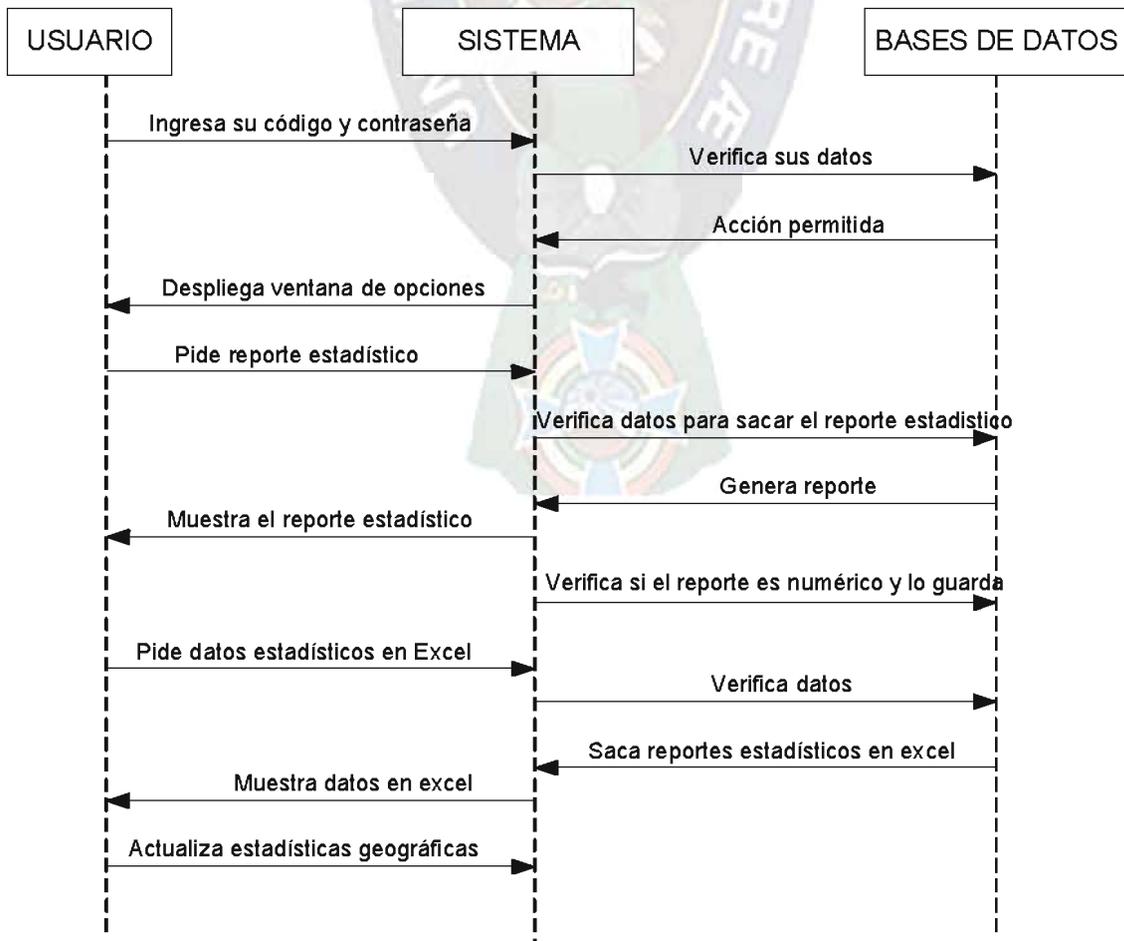
**Figura 11: Trazos de eventos
Registro del Informe Epidemiológico de Diagnostico de Rabia**



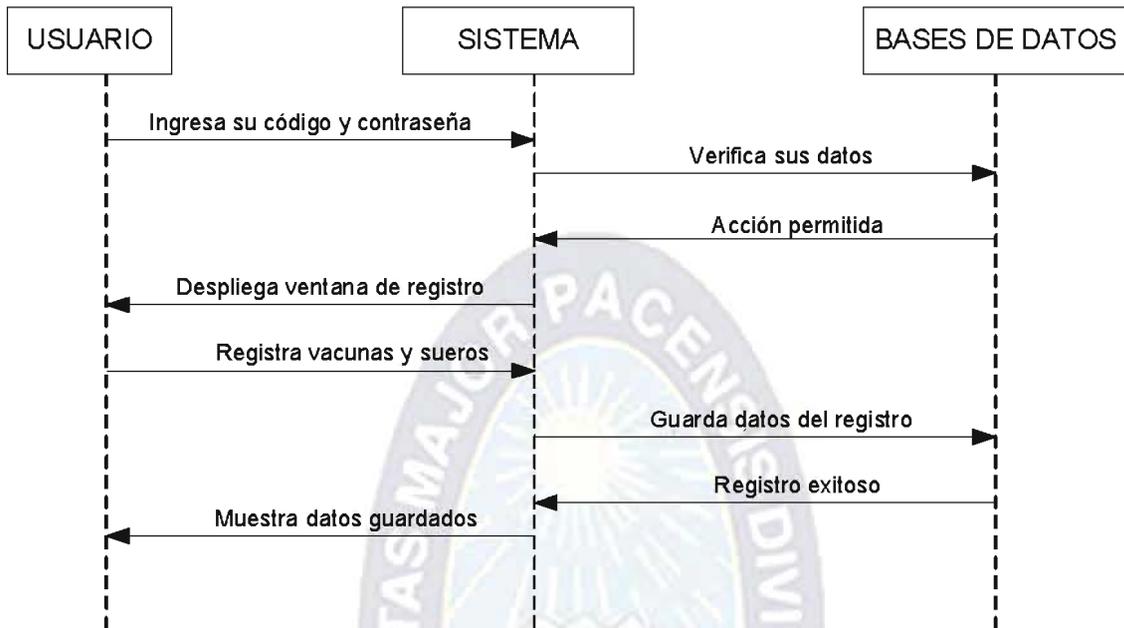
Momificación del Informe Epidemiológico de Diagnostico de Rabia



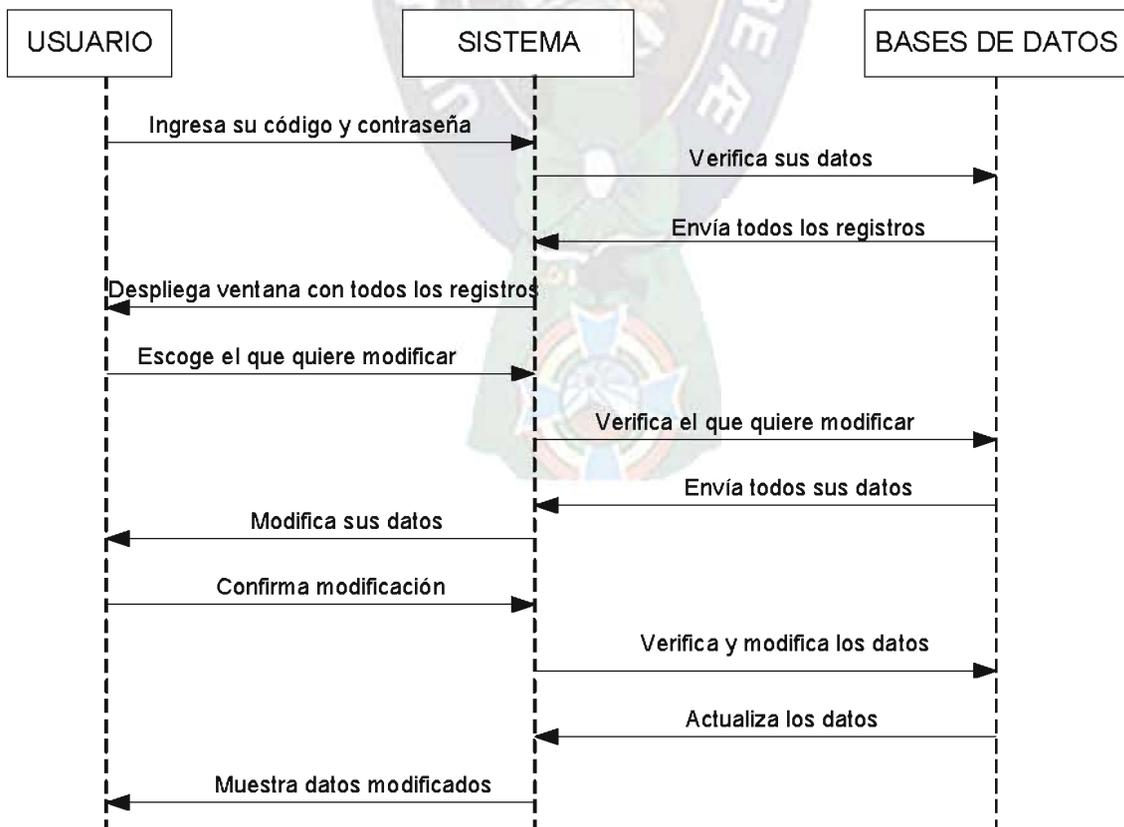
Control del Informe Epidemiológico de Diagnostico de Rabia



Registro de Vacunas y Sueros



Modificación de Vacunas y Sueros



3.1.3.3. Diagrama de Flujo de Eventos para el Sistema

Construimos el diagrama de eventos completo.

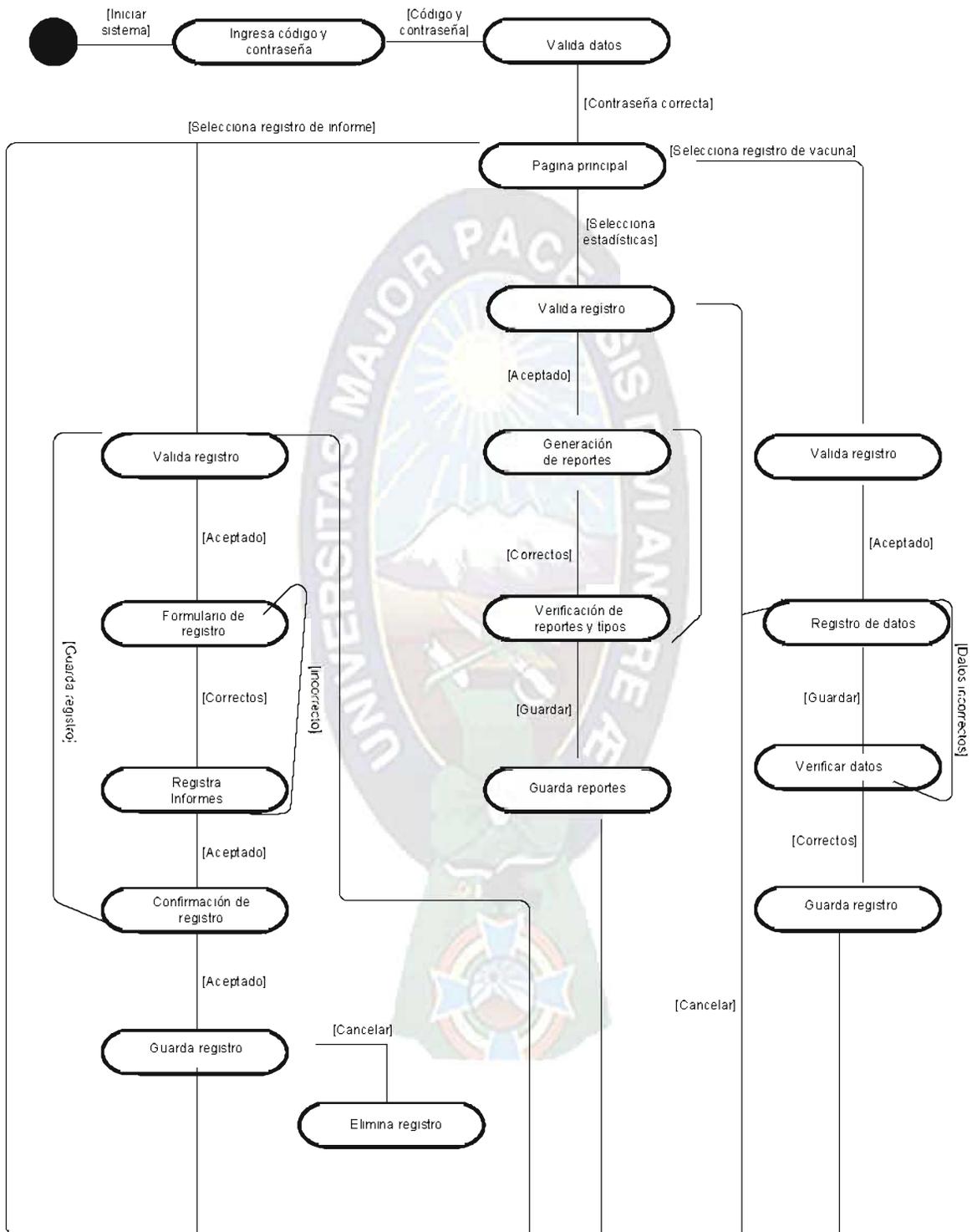
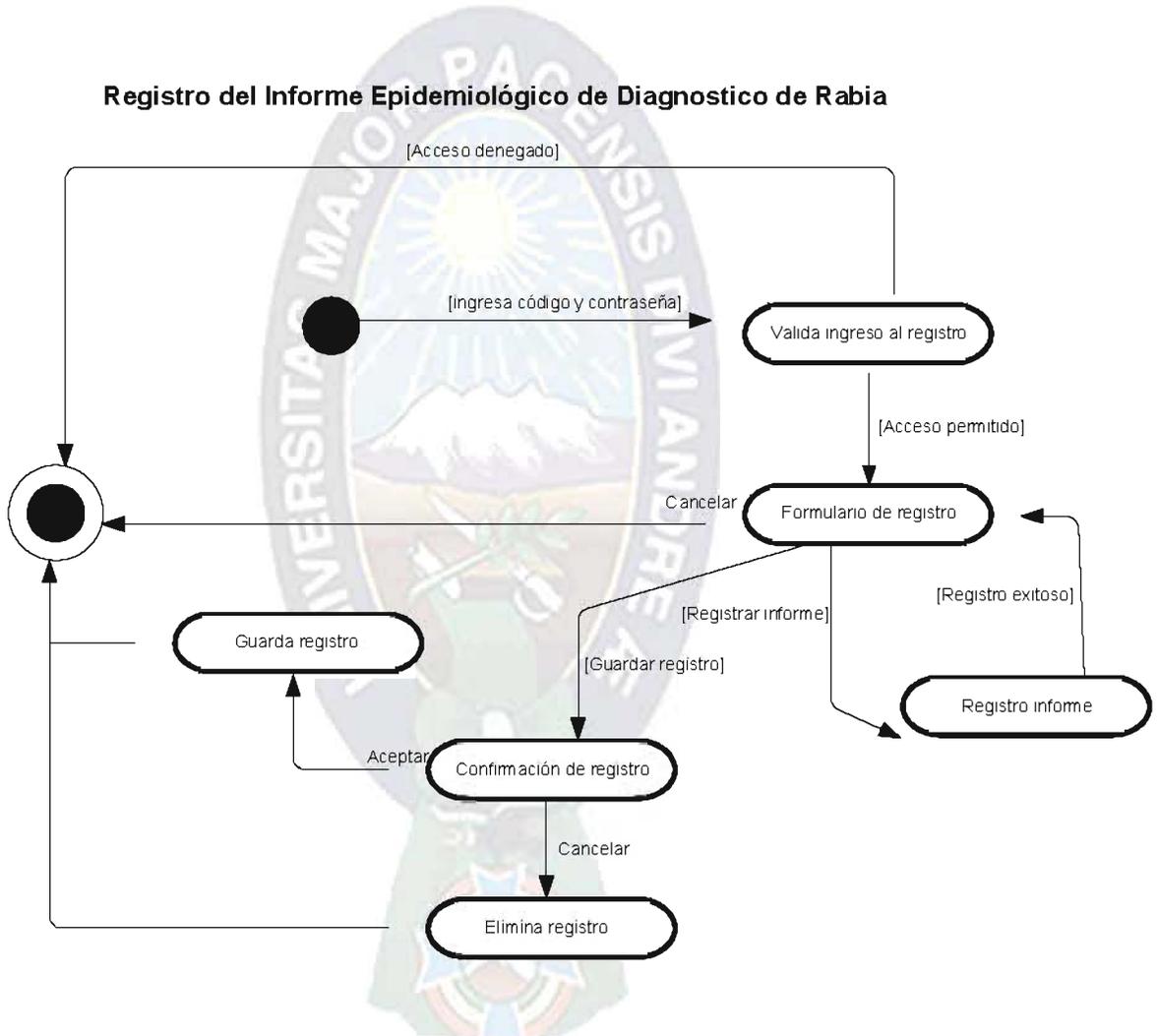


Figura 12: Diagrama de flujo de eventos

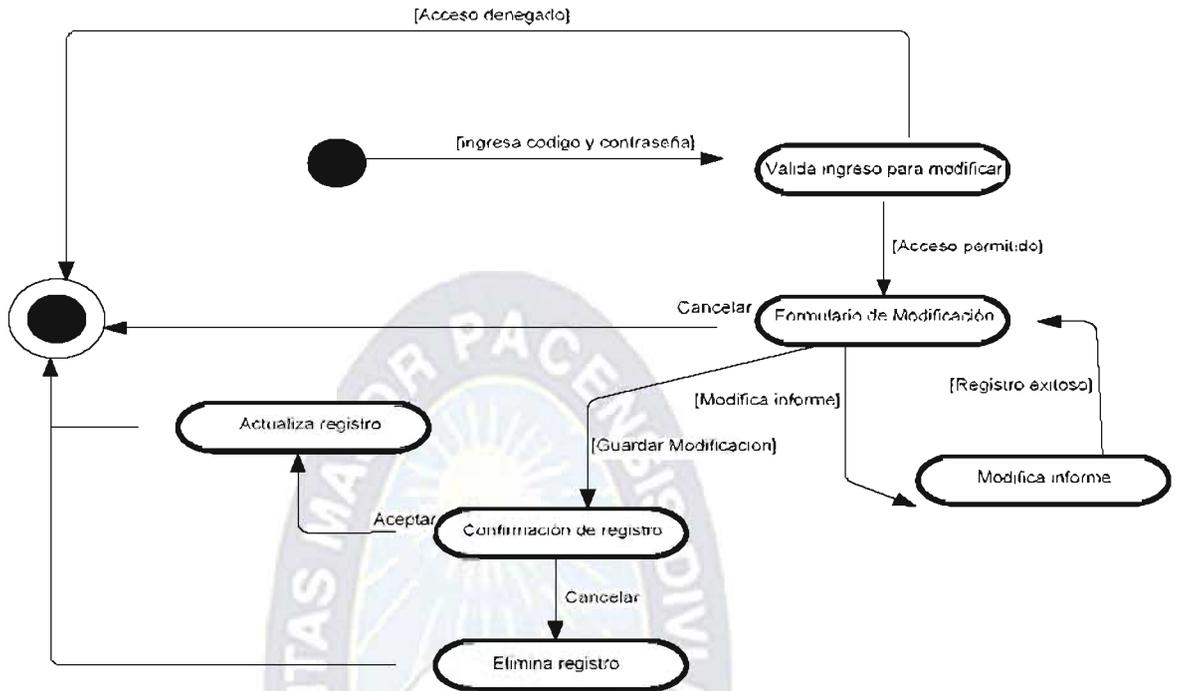
3.1.3.4. Diagrama de de estados para cada clase con comportamiento dinámico.

Para generar los diagramas de estados de una clase se toma en cuenta el seguimiento de sucesos afectando a la clase que es modelada.

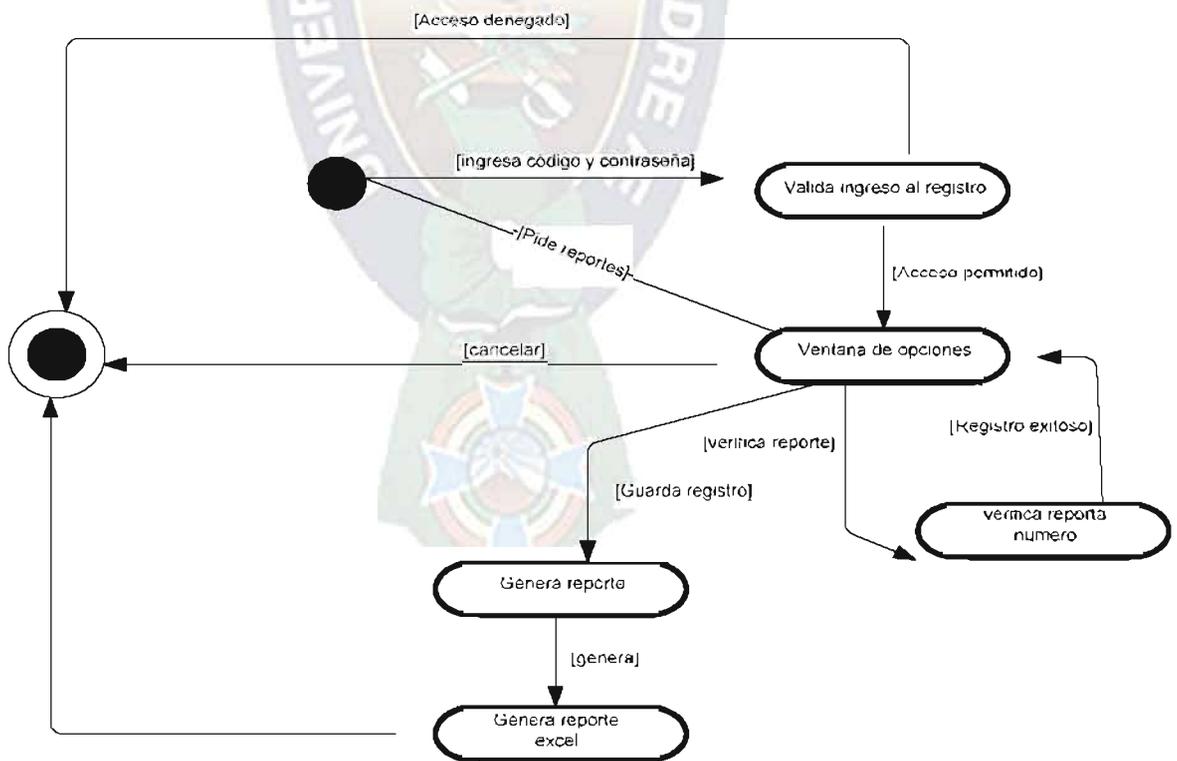
Figura 13: Diagramas de estados



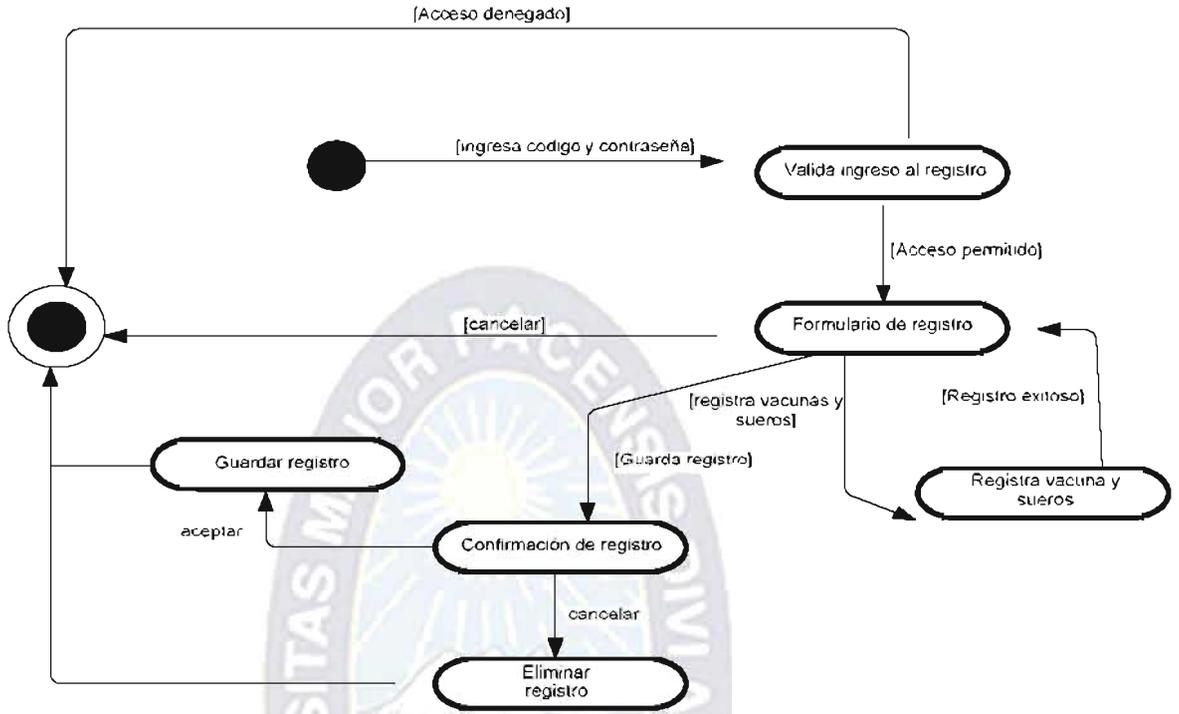
Modificación del Informe Epidemiológico de Diagnostico de Rabia



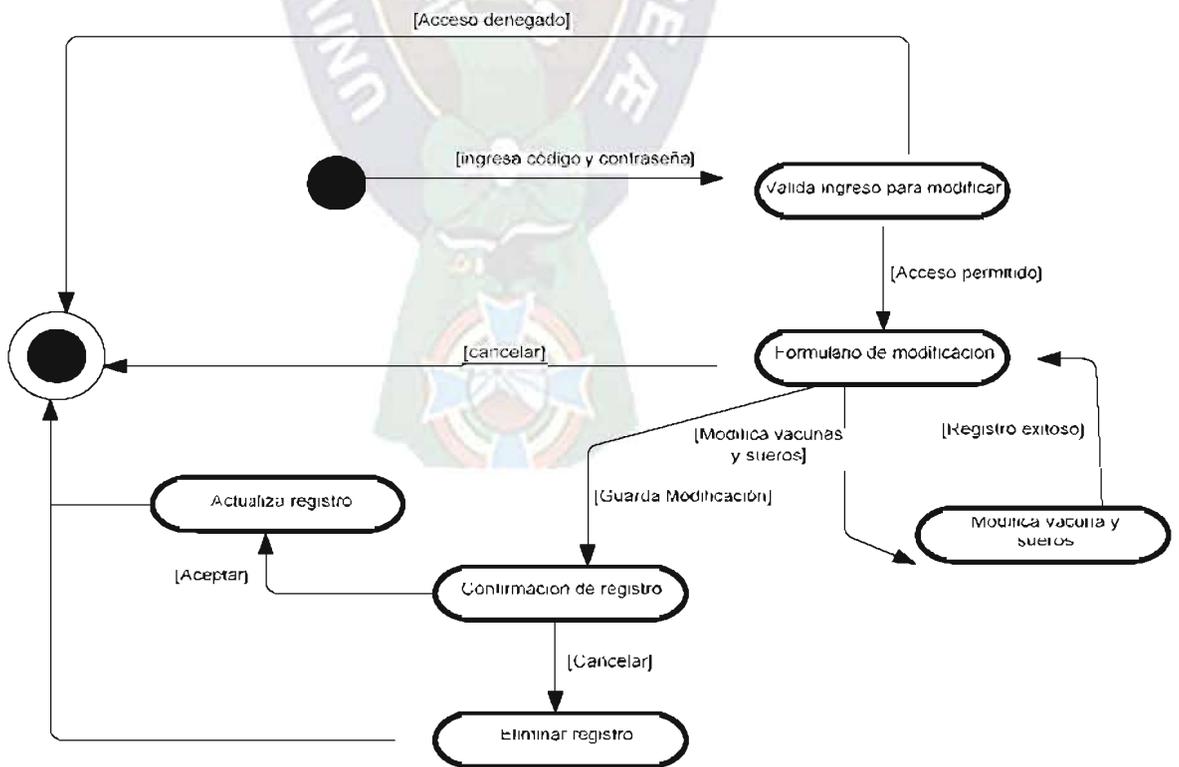
Control del Informe Epidemiológico de Diagnostico de Rabia



Registro de vacunas y sueros



Modificación de Vacunas y Sueros



3.1.4. Modelo Funcional

Captura lo que hace el sistema, independientemente de cuando se haga o de la forma en que se haga.

3.1.4.1. Diagrama de Contexto

El diagrama de contexto es la descripción general del sistema.



Figura 14: Diagrama de contexto

3.1.4.2. Diagrama de Flujo de Datos (Padre)

En el diagrama de flujo de datos (Padre) se hace una descripción detallada del sistema y su funcionalidad.

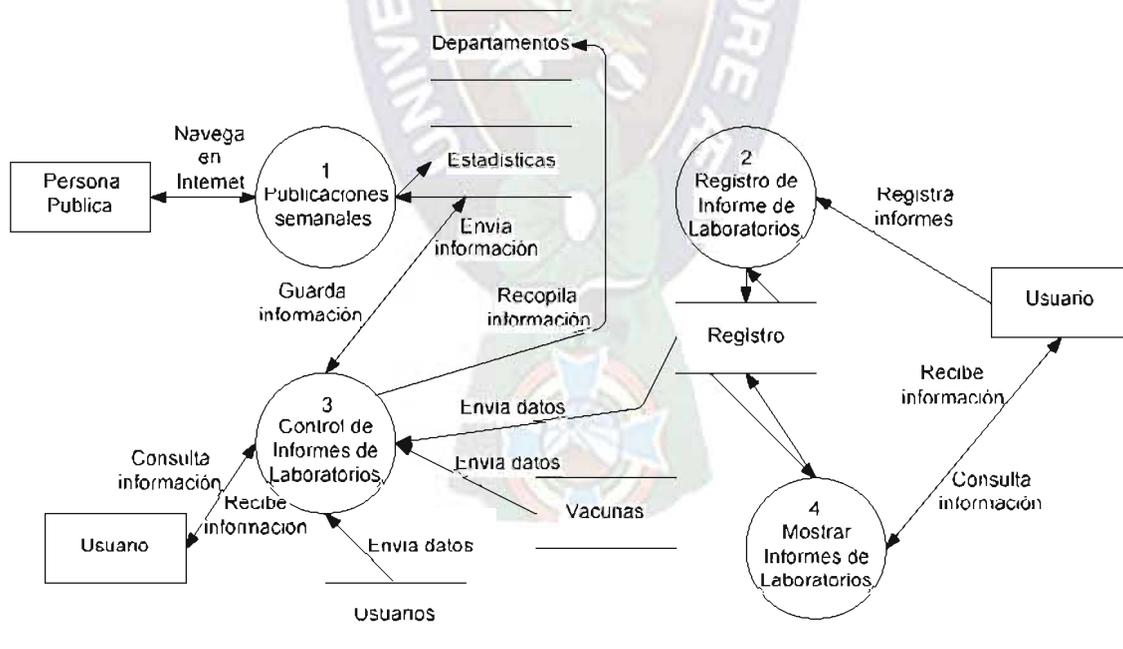
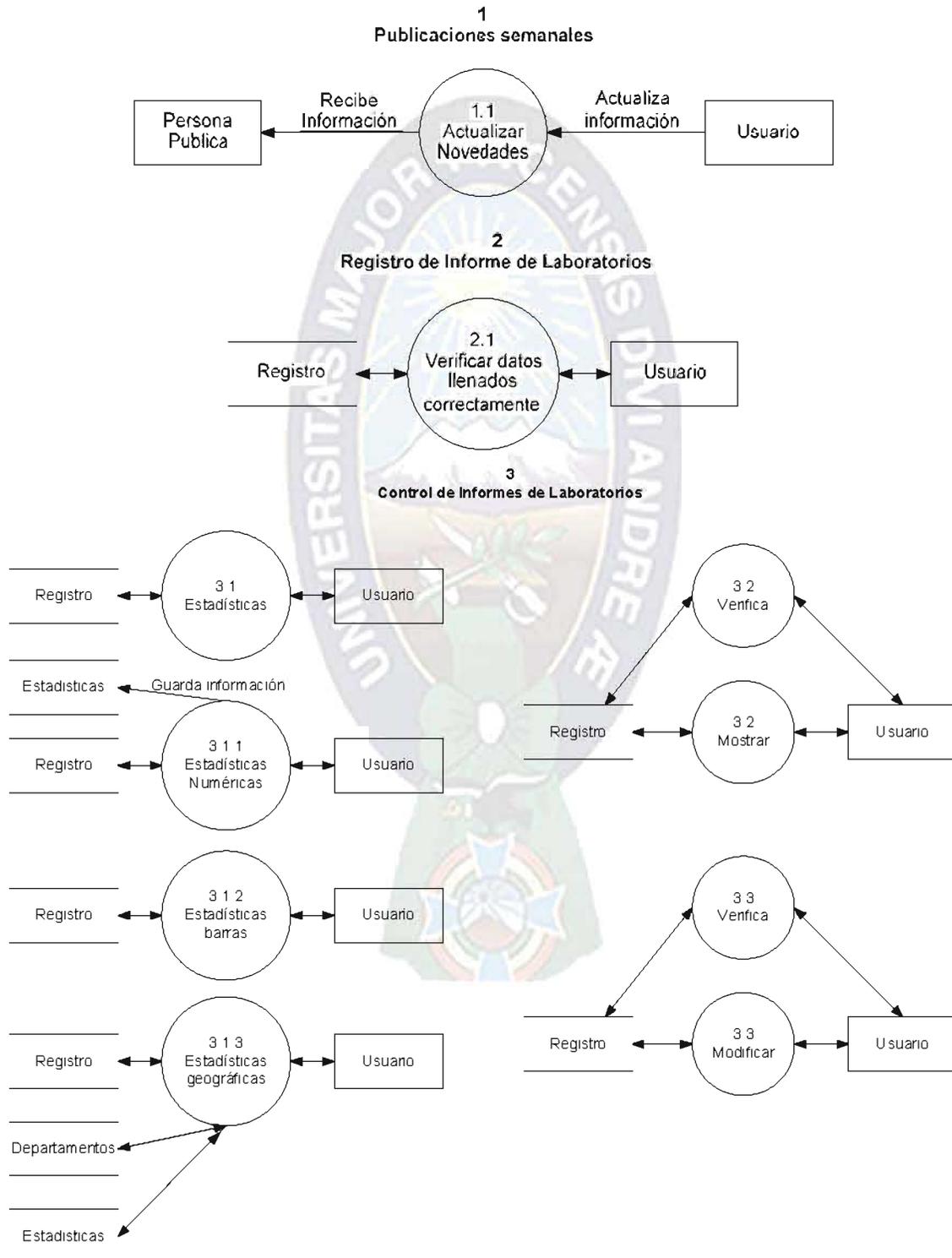


Figura 15: Diagrama de flujo de datos

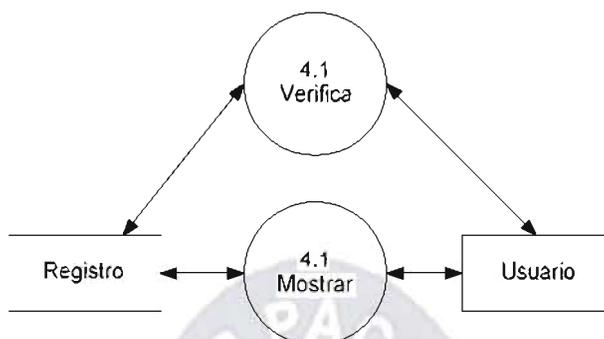
3.1.4.3. Diagrama de Flujo de Datos (Hijos)

En los diagramas de flujo de datos (Hijos) se desarrollan en detalle cada proceso del diagrama padre.

Figura 16: Diagramas de flujo de datos al detalle



Mostrar Informes de Laboratorios



3.2. Fase de Diseño de Sistemas

3.2.1. Estructura de la Arquitectura Básica del Sistema

Existen varias arquitecturas canónicas que pueden servir como un punto de inicio adecuado.

3.2.1.1. Organización del sistema en subsistemas

Dividiremos el sistema en subsistemas de acuerdo a aspectos que compartan alguna propiedad en común.

Subsistemas	Descripción
Inicio de Sistema	En este modulo se muestra toda la información educativa referente a la enfermedad de la rabia, se muestran enlaces relacionados con la misma y las ultimas estadísticas nacionales de la rabia.
Laboratorios	En este modulo se registra los formularios de informes de rabia, se muestran todos los formularios que deseen ver por fechas.
Administración	En este modulo se registra las vacunas y sueros enviados a toda Bolivia, se muestran datos según fecha, se sacan estadísticas numéricas, en barras y geográficas, se hacen modificaciones de tanto de los informes como de las vacunas y sueros.

Tabla 12: Organización del sistema en subsistemas

3.2.2. Dediciones de estrategias de alto nivel

Durante el diseño de sistemas, se selecciona la estructura de alto nivel del sistema.

3.2.2.1. Asignación de subsistemas a procesadores y tareas

En este caso estimaremos requisitos de hardware y software ya que no encontramos subsistemas concurrentes.

Windows XP

Es el sistema operativo que utilizarán los usuarios.

MySQL Database Version 4.0.20

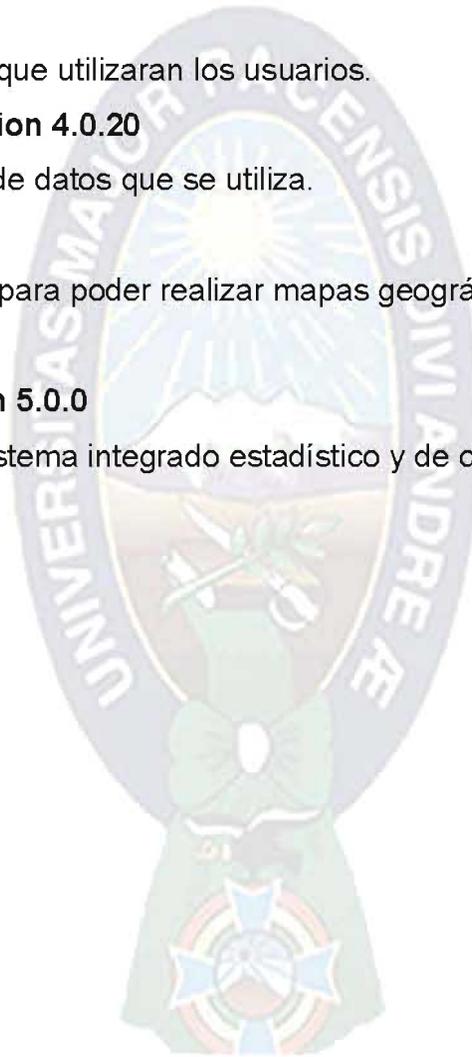
Es el gestor de la base de datos que se utiliza.

Arcview 3.2

Es nuestra herramienta para poder realizar mapas geográficos.

PHP Language Version 5.0.0

Es para desarrollar el sistema integrado estadístico y de control de la rabia en Bolivia.

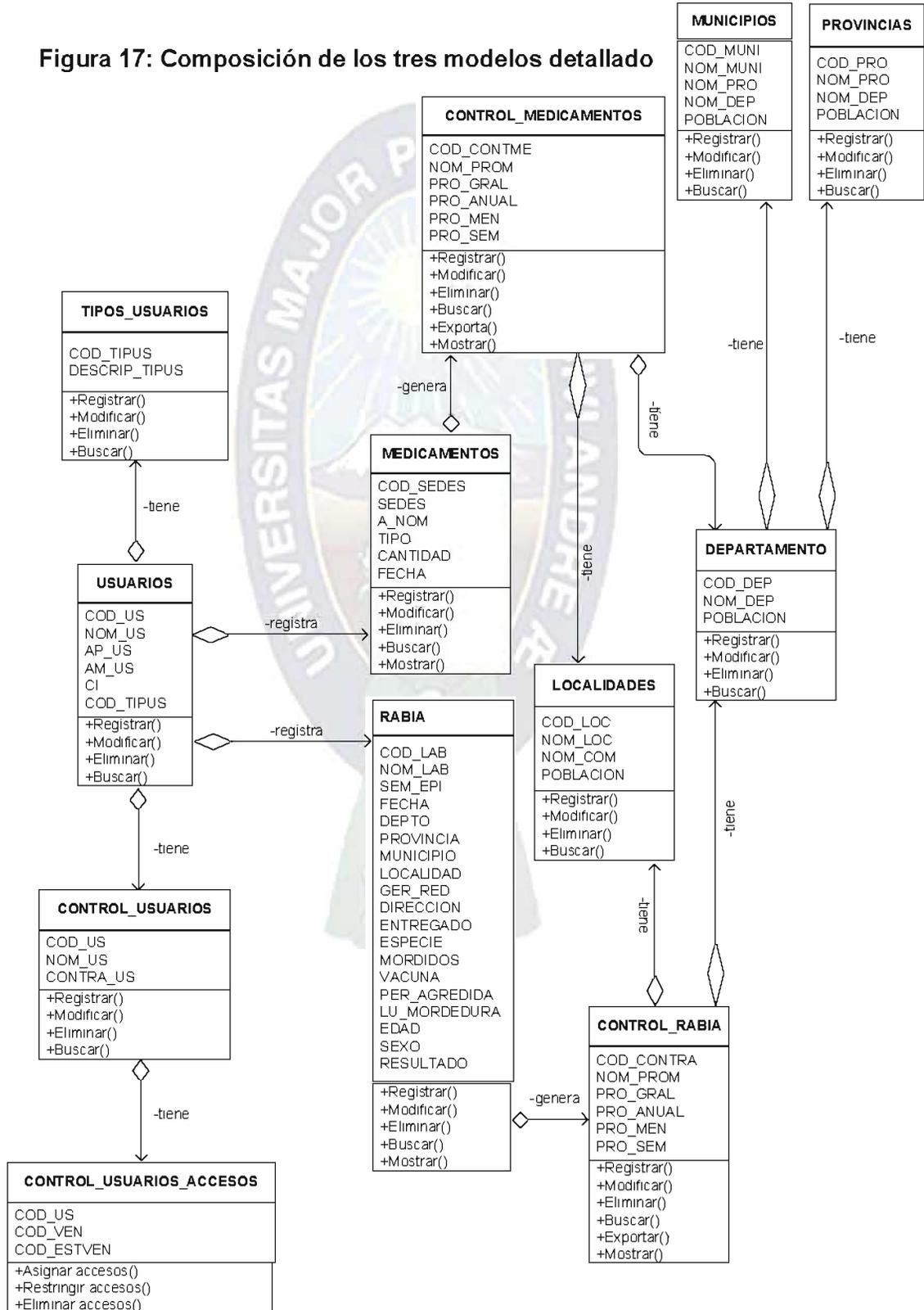


3.3. Fase de Diseño de Objetos

3.3.1. Composición de los tres modelos detallado

Luego de haber desarrollado los tres modelos en la fase de análisis mostraremos a continuación el producto final de los tres modelos.

Figura 17: Composición de los tres modelos detallado





CAPITULO IV

METRICAS DE CALIDAD

METRICAS DE CALIDAD

4.1. ISO 9126

ISO 9126 es un estándar internacional para la evaluación del Software. Está supervisado por el proyecto SQuaRE, ISO 25000:2005, el cuál sigue los mismos conceptos.

El estándar está dividido en cuatro partes las cuales dirigen, respectivamente, lo siguiente: modelo de calidad, métricas externas, métricas internas y calidad en las métricas de uso.

El modelo de calidad establecido en la primera parte del estándar, ISO 9126-1, clasifica la calidad del software en un conjunto estructurado de características y subcaracterísticas de la siguiente manera:

- Funcionalidad - Un conjunto de atributos que se relacionan con la existencia de un conjunto de funciones y sus propiedades específicas. Las funciones son aquellas que satisfacen lo indicado o implica necesidades.
 - Idoneidad
 - Exactitud
 - Interoperabilidad
 - Seguridad
 - Cumplimiento de normas.
- Fiabilidad - Un conjunto de atributos relacionados con la capacidad del software de mantener su nivel de prestación bajo condiciones establecidas durante un período de tiempo establecido.
 - Madurez
 - Recuperabilidad
 - Tolerancia a fallos
- Usabilidad - Un conjuntos de atributos relacionados con el esfuerzo necesitado para el uso, y en la valoración individual de tal uso, por un establecido o implicado conjunto de usuarios.

- Aprendizaje
- Comprensión
- Operatividad
- Eficiencia - Conjunto de atributos relacionados con la relación entre el nivel de desempeño del software y la cantidad de recursos necesitados bajo condiciones establecidas.
 - Comportamiento en el tiempo
 - Comportamiento de recursos

4.1.1. Funcionalidad

Para hallar la funcionalidad del sistema se consideran los siguientes aspectos:

Parámetros de medición	Cuenta	X	Simple	Mediano	Complejo	=	Resultado
Numero de entradas de usuario	5	X	3	4	6	=	30
Numero de salidas de usuario	10	X	4	5	7	=	70
Numero de peticiones de usuario	5	X	3	4	6	=	30
Numero de archivos	12	X	7	10	15	=	180
Numero de interfaces externas	1	X	5	7	10	=	10
TOTAL							320

Tabla 13: Métricas Punto Función

$$PF = \text{Cuenta Total} \times [0.65 + 0.01 \times (F_i)]$$

- Donde: PF: → Punto de Función
 Cuenta Total: → Suma de todas las entradas e la tabla
 Fi: → (i=1 a 14) valores de ajuste de la complejidad según respuesta a las preguntas

FACTOR	0	1	2	3	4	5
¿Requiere el sistema de copias de seguridad?						X
¿Se requiere comunicación entre datos?						X
¿Existen funciones de procesamiento distribuidas?						X
¿Es crítico el rendimiento?		X				
¿Se ejecutara el sistema en un entorno operativo existente y fuertemente utilizado?					X	
¿Requiere el sistema de entrada de datos interactiva?						X
¿Requiere la entrada de datos interactivos que las transacciones de entrada se lleven acabo sobre múltiples pantallas u operaciones?		X				
¿Se actualizan los archivos maestros de forma interactiva?					X	
¿Son complejos las entradas, las salidas, los archivos o las peticiones?					X	
¿Es complejo el procesamiento interno?						X
¿Se a diseñado el código para ser reutilizable?						X
¿Están incluidas en el diseño la conversión y la instalación?			X			
¿Se a diseñado el sistema para soportar múltiples instalaciones en diferentes organizaciones?						X
¿Se a diseñado la aplicación para facilitar los cambios y para ser fácilmente utilizado por el usuario?						X

Tabla 14: Factor Punto función

Por lo tanto la suma de los Fi es:

$$\sum Fi = 56$$



$$PF = \text{Cuenta Total} \times [0.65 + 0.01 \times (Fi)]$$

$$PF = 320 \times [0.65 + 0.01 \times (56)]$$

$$PF = 387.2$$

Si Fi se considera como el 100%



$$PF = 320 \times [0.65 + 0.01 \times 80]$$

$$PF = 464$$



$$387.2 / 464 = 0.83 = 83\% \text{ de funcionalidad del sistema.}$$

4.1.2. Fiabilidad

Definición	Criterio elemental	%
Representa los enlaces encontrados que conducen a destinos ausentes y como parámetro.	Variable normalizada, continua, absoluta	90
Si existe mala presentación del sitio Web a los navegadores Página en construcción También páginas muertas	Referencia de claridad directa	80

Tabla 15: Fiabilidad

4.1.3. Usabilidad

Definición	Criterio elemental	%
Con componentes gráficos que muestra la arquitectura global (jerárquica) del sitio Web	Criterio discreto y absoluto	100
Permite conocer al cliente características de la Web	Criterio discreto y absoluto	90
Indica el agrupamiento adecuado de enlaces	Criterio de referencia de calidad directa	95
Todo los sitios de enlace son de mismo color	Criterio de referencia de calidad directa	95
Todo la Web es uniforme estilo y fuente también en su totalidad	Criterio de referencia de calidad directa	100

Tabla 16: Usabilidad

4.1.4. Eficiencia

Definición	Criterio elemental	%
Mide el tamaño de todas las paginas de aplicación considerando todo sus componentes	Multi_variable, contenido	80
Se encuentra en la pagina versión de solo texto e imagen	Multi_nivel	80
Los frames o marcos organizan a una ventana en diferentes áreas su contenido	Multi_variable, contenido	100

Tabla 17: Eficiencia

Todos estos beneficios implican una reducción y optimización general de los costes de producción, así como un aumento en la productividad.

Resultados de las preferencia global

Atributo	Preferencia Global
Funcionalidad	83
Usabilidad	85
Eficiencia	96
Confiabilidad	86.6





CAPITULO V

CONCLUSIONES Y RECOMENDACIONES

CONCLUSIONES Y RECOMENDACIONES

5.1. Conclusiones

- Por medio del sistema se logra implementar el manejo del formato de ingreso de datos de los diferentes laboratorios de La Paz (INLASA), Cochabamba (LIDIVECO) y Santa Cruz (LIDIVET).
- Se logro manejar los datos de los laboratorios semanalmente cosa que no ocurría anteriormente.
- Se pueden obtener los reportes de diferentes formas tanto numéricos, en barras y geográficas a nivel nacional, como departamental, para mas detalle se logro también por localidades.
- Teniendo datos importantes nacionales, la seguridad de la misma es manipulado solo por usuarios autorizados.
- Por medio del sistema también logramos tener el control de las vacunas que son enviados para su uso a distintos departamentos.
- Se logra informar a la población de los sucesos importantes que pasa en el ministerio con relación a la rabia y se puede ver los datos estadísticos a nivel nacional.

5.2. Recomendaciones

Actualizar las novedades de la página Web constantemente para así poder educar más a las personas en el conocimiento de esta enfermedad mortal.

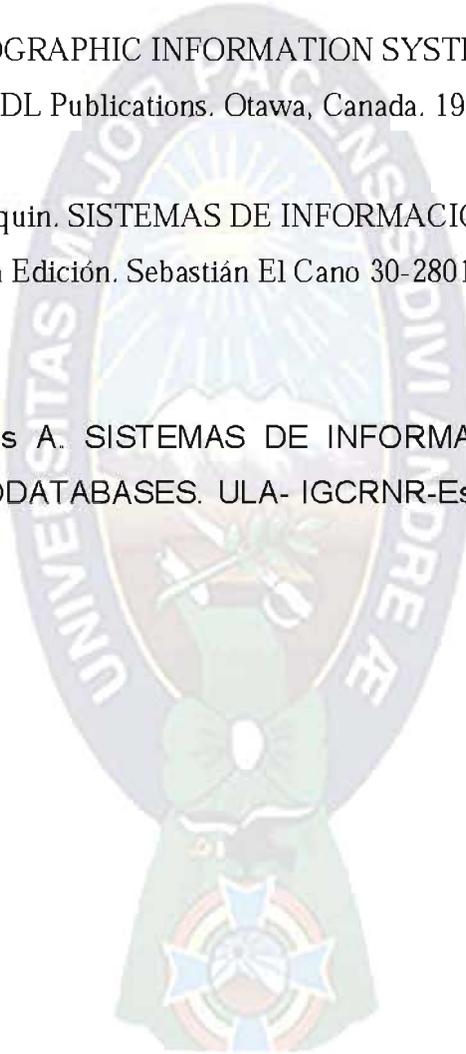
Sacar constantemente repotes estadísticos numéricos para poder actualizar los mapas geográficos.

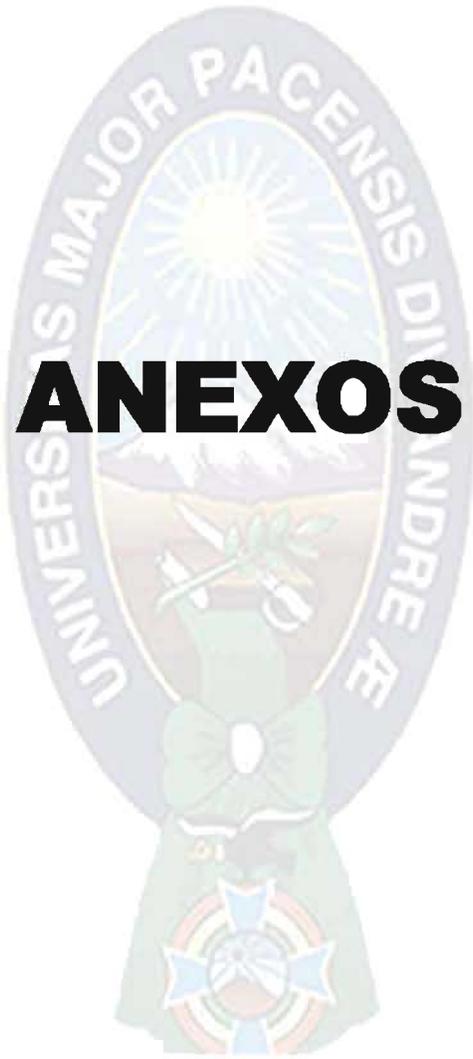
Tener siempre la computadora en buenas condiciones para que los mapas geográficos carguen con mayor velocidad.

Tener una persona a cargo con conocimientos de Arcview 3.2 para poder actualizar los mapas y solucionar cualquier problema que se presente.

BIBLIOGRAFÍA

1. Dr. Freddy Lizon F. Responsable Programa Nacional Zoonosis Ministerio de Salud y Deporte.
2. Internet de Wikipedia, la enciclopedia libre.
3. Aronoff, Stan. GEOGRAPHIC INFORMATION SYSTEMS: A MANAGEMENT PERSPECTIVE. WDL Publications. Ottawa, Canada. 1989. 294p.
4. Bosque Sendra, Joaquin. SISTEMAS DE INFORMACION GEOGRAFICOS. Ediciones Rialp. 1ra Edición. Sebastián El Cano 30-28012 Madrid, España. 1994. 451 p.
5. Montilva C, Jonás A. SISTEMAS DE INFORMACIÓN GEOGRÁFICA Y DISEÑO DE GEODATABASES. ULA- IGCRNR-Esc. Geografía. Manual de Curso. 1994.



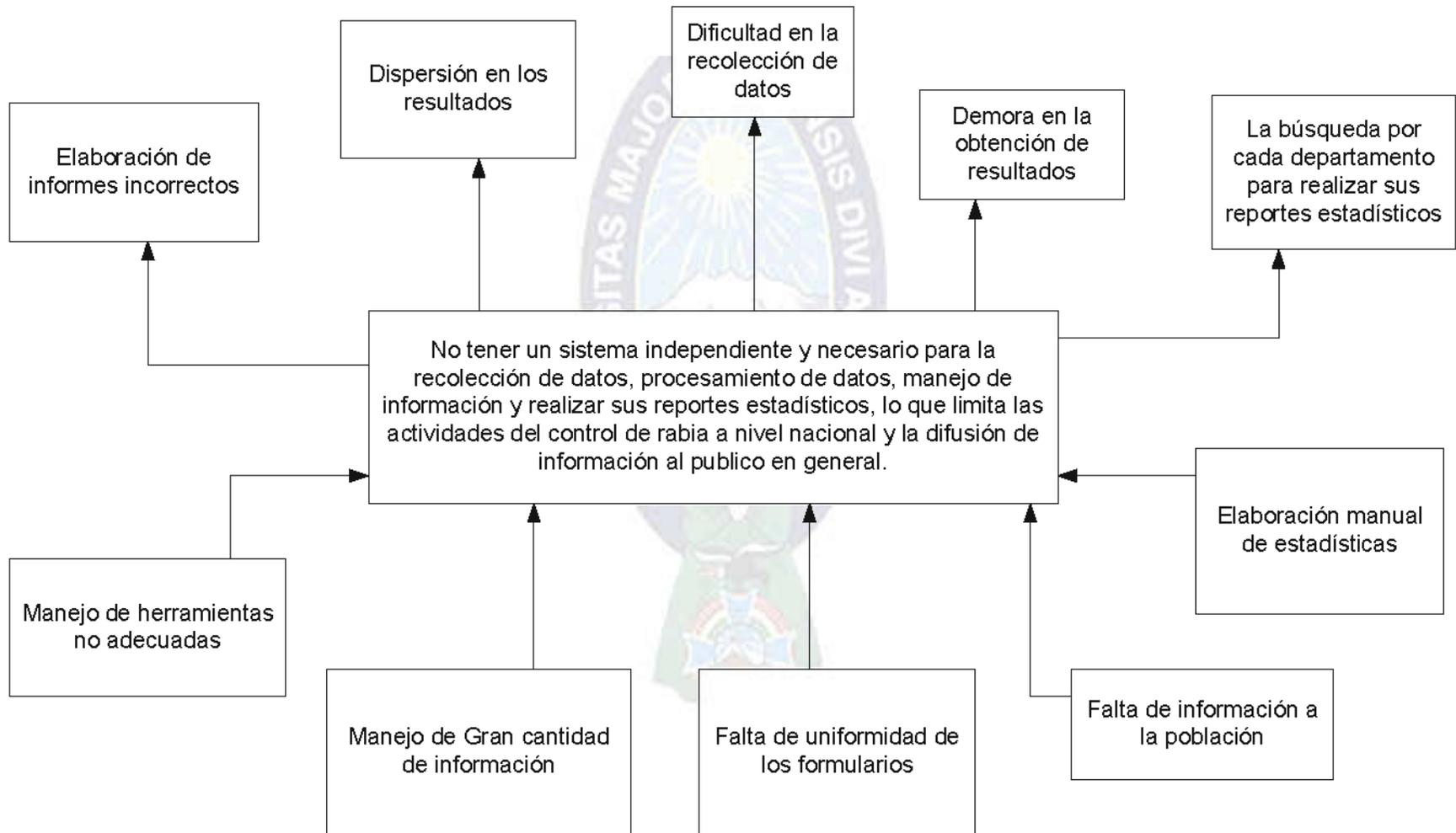


ANEXOS

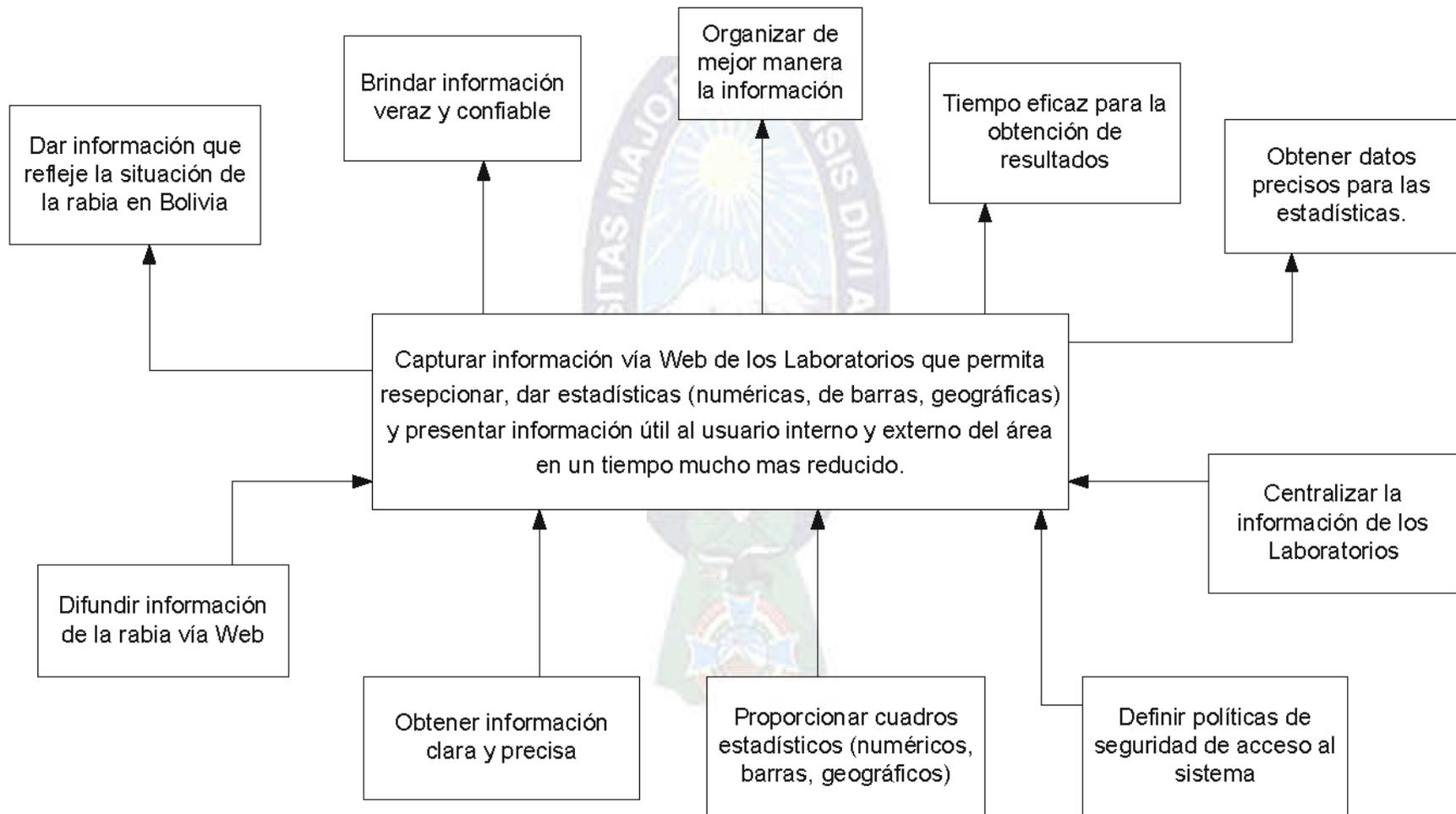
Anexo A: Análisis de Involucrados

Problema	Causa	Efecto	Involucrados
Los datos que mandan los Laboratorios de La Paz, Cochabamba y Santa Cruz a diferentes interventores no coinciden.	Falta de comunicación entre Laboratorios.	Los datos estadísticos no coinciden.	Jefatura nacional de Zoonosis, Laboratorios.
Envío de Datos en Word o Excel por correo electrónico.	Falta de un sistema automatizado.	Pérdida de información.	Jefatura nacional de Zoonosis, Laboratorios.
Las planillas de datos entre los Laboratorios de La Paz, Cochabamba y Santa Cruz son diferentes.	Falta de comunicación entre Laboratorios.	Diferencia de datos.	Jefatura nacional de Zoonosis, Laboratorios.
Trascripción de datos en forma manual de cada Laboratorio.	Falta de un sistema automatizado.	Pérdida de tiempo Horas/Hombre.	Jefatura nacional de Zoonosis.
Hacer una búsqueda por cada departamento para realizar sus reportes estadísticos.	Falta de un sistema automatizado.	Pérdida de tiempo Horas/Hombre.	Jefatura nacional de Zoonosis.
La no existencia de información a la población Nacional.	Falta de propagación de la información de la rabia en Bolivia.	Ignorancia respecto a la prevención de Dicha enfermedad.	Jefatura nacional de Zoonosis, población en general.

Anexo B: Árbol de Problemas



Anexo C: Árbol de Objetivos



Anexo D: Matriz Del Marco Lógico.

	Resumen Narrativo	Indicadores objetivamente verificables	Medios de verificación	Supuestos
Fin	Lograr la organización optima de la unidad de epidemiología.	El 90 % de operadores evitaran las tares de reajo y entrega de formularios.	Reportes con información correcta y adecuada para la toma de decisiones.	Seguridad de manejo de datos correctos.
Propósito	Implementar un sistema integrado estadístico y de control de la rabia en Bolivia, que permita consolidar y difundir información acerca del área.	Sistema informático concluido en agosto 2008.	Documentación del proyecto de grado, manual de usuario, manual de funciones, manual de contingencias.	La información que envían los laboratorios sea verídico.
Componentes	Obtención de datos, proporcionar información de manera grafica, proporcionar cuadros estadísticos, informes centralizados.	Todos los módulos estarán funcionado hasta Agosto	Informes de referencia de los cuadros de estadísticos, reportes impresos de los resultados obtenidos.	Contar con equipos solicitados que exista. Usuarios capacitados para el uso del sistema.

Actividades	Recolección y organización de datos. Análisis y diseño. Implementación. Pruebas. Capacitación para los usuarios.	Entrevista y documentación(facilitado por la institución). Computadoras (facilitados por la institución). Software (gratis) Capacitación a los usuarios.	Informes consecutivos para el análisis y diseño del sistema.	Contar con herramientas de software y el apoyo del personal del área.
--------------------	--	---	--	---



Anexo E: Marco Institucional.

El plan operativo nacional de control de la rabia (PONCRA) y las políticas de salud.

Este **Plan Operativo Nacional de Control de la Rabia (PONCRA)** es un instrumento de la nueva visión política y epidemiológica del Ministerio de Salud y Deportes; plantea, y adecua, las normas técnicas, los procedimientos y las estrategias de intervención, con el fin de mejorar el control y prevención de la rabia en nuestro país. De esta manera pretende contribuir a las Políticas Nacionales en la edificación de un país con equidad y justicia social, constituyéndose el control de esta zoonosis en un aporte a la lucha contra la pobreza.

El PONCRA define las líneas estratégicas de acción para garantizar la descentralización operativa del control de la rabia a nivel nacional. Fue elaborado en cada departamento con los directos responsables del control de rabia en los servicios departamentales de salud y municipios; estos deberán adecuar las actividades del plan para años posteriores de acuerdo a sus caracterizaciones epidemiológicas regionales y en coordinación con el Programa Nacional Zoonosis.

Objetivos

Objetivos Generales

Prevenir y controlar la rabia en el territorio de Bolivia.

Objetivos Específicos

- Prevenir la rabia humana.
- Controlar la rabia humana.
- Prevenir la rabia canina.
- Controlar la rabia canina.
- Controlar la aparición de focos epidémicos.
- Educar, informar y capacitar a la población para la prevención de esta enfermedad.
- Controlar la sobrepoblación animal.

- Difundir y ejecutar normas, manuales, ordenanzas municipales, resoluciones, y otros, sobre esta enfermedad y la tenencia responsable de animales.

Justificación

El PONCRA contempla una serie de acciones para este quinquenio (2007-2011) con el fin de tener un plan eficiente con capacidad de detección elevada de casos y el respectivo tratamiento del 100% de estos; para lo cual deberán contar con los suficientes recursos económicos aportados por parte del Ministerio de Salud y Deportes (MSyD), Prefecturas, y Municipios y tener una interacción oportuna entre estas instituciones y los actores sociales de nuestra comunidad

Estas acciones e interacciones contribuirán a disminuir las fuentes de infección, la transmisión del virus y mejoraran la tenencia responsable de animales, de este modo evitaremos fallecimientos a causa de esta enfermedad; así mismo; este plan comprometerá la participación activa de la sociedad civil organizada, toda la comunidad y de los actores con poder político y social que garanticen la sostenibilidad del PONCRA a largo plazo.

Metas

- Las 9 Prefecturas, los Municipios identificados de mayor riesgo y otras organizaciones que apoyan al control de la rabia han conformado los comités regionales de zoonosis; quienes coordinan con los SEDES el cumplimiento de los objetivos de los planes operativos de control de la rabia elaborados en cada municipio.
- El MSyD, SEDES y municipios cuentan con el 80% de RRHH capacitados en todo lo relacionado con la prevención, vigilancia y control de la rabia.
- El MSyD, SEDES y municipios orientan al 100% de la población del país para prevenir la rabia
- El 100% de los municipios cuentan con ordenanzas municipales y centros de zoonosis para el control de enfermedades zoonóticas.
- El MSyD asegura la dotación del 100% de inmunobiológicos para la profilaxis y tratamiento de la rabia, en las cantidades requeridas por cada departamento de acuerdo a sus programaciones.
- Los SEDES aseguran la participación del 100% de los RRHH en la ejecución del Plan

- Los Municipios dotan del 100% de los insumos requeridos para la ejecución del PNCR
- Los 9 SEDES proporcionan tratamiento y profilaxis adecuado y oportuno al 100% de las personas en riesgo y en accidentes rabícos.
- Los Municipios controlan al 100% de la población animal transmisora de esta enfermedad.
- El MSyD capacita y actualiza al 100% del personal de salud de los SEDES y Municipios en: tratamiento, profilaxis, gerencia, diagnóstico, normas y procedimientos; para el control y prevención de la rabia
- El MSyD, SEDES y Municipios realizan supervisiones, evaluaciones, ajustes y monitoreo en el 100% de los servicios de salud para la ejecución del PNCR

Estrategias

Las estrategias para la implementación de este plan están dadas por los siguientes componentes:

- Vacunación.
- Vigilancia Epidemiológica.
- Comunicación Social y Educación.
- Implementación Centros Municipales de Zoonosis.
- Control Sobrepoblación Animal.
- Participación de la Sociedad Civil, Actores Sociales, Compromiso y Coordinación Intersectorial e Interinstitucional.
- Coordinación del Apoyo Internacional.

Prototipos del Sistema



Ministerio de Salud y Deportes

AL DE SALUD PROGRAMAS NACIONALES ZOOZOSIS

INICIO LABORATORIOS ADMINISTRACION

INFORME EPIDEMIOLOGICO DE DIAGNOSTICO DE RABIA

Nombre del Laboratorio: NIASA

Semanas Epidemiológicas: 1

Fecha: 01/07/2006

Código de Laboratorio: 00108

Departamento: Chuquisaca

Provincia: Asnesu

Municipio: Chulupani

Lugar Habitado: 7.174F.97

Gerencia de Red:

Dirección:

REGISTRO

MOSTRAR

Ministerio de Salud y Deportes

UNIDAD

INICIO LABORATORIOS ADMINISTRACION

Nombre Laboratorio	Semanas Epidemiológicas	Día	Mes	Año	Código Laboratorio	Departamento	Provincia	Municipio	Lugar Habitado
NIASA	1	0	0	2006	00108	Chuquisaca	Asnesu	Chulupani	Porqueñi (Lugar Habitado)
ENLASA	1	0	0	2006	00208	La Paz	Mucilo	La Paz	C/ Rosendo Quiroga Z/ Saguro
ENLASA	1	0	0	2006	00308	La Paz	Mucilo	La Paz	C/ Cuba N° 1196 Z/ Mendoz...

REGISTRO

MOSTRAR

Ministerio de Salud y Deportes

UNIDAD

INICIO LABORATORIOS ADMINISTRACION

ADMINISTRACION

Programa Nacional de Zoonosis



0 de 300

Día	Ene	Feb	Mar	Abr	May	Jun	Jul
1	1	2	3	4	5	6	7
2	8	9	10	11	12	13	14
3	15	16	17	18	19	20	21
4	22	23	24	25	26	27	28
5	29	30	31				

LABORATORIOS

USUARIOS

NOTAS

Ministerio de Salud y Deportes

INICIO LABORATORIOS ADMINISTRACION

ESTADISTICAS MOSTRAR MODIFICAR

NUMERICAS

BARRAS

GEOGRAFICAS

LABORATORIOS

NOTAS

Ministerio de Salud y Deportes

DIRECCION GENERAL DE SALUD PROGRAMA NACIONAL ZOO

INICIO LABORATORIOS ADMINISTRACION

REGISTRAR MOSTRAR MODIFICAR

ESCOJA LA SEDES QUE QUIERE MODIFICAR

ID	SEDES						
#	de	ILIO	pero	25	3	Sede	2008

LABORATORIOS

NOTAS

Ministerio de Salud y Deportes

DIRECCION GENERAL DE EPIDEMIOLOGIA DIRECCION GENERAL DE SALUD

INICIO LABORATORIOS ADMINISTRACION

REGISTRAR MOSTRAR MODIFICAR

REGISTRO DE VACUNAS Y SUBROS

SEDES

A Nombre de

Tipo

Cantidad

Fecha

ACEPTAR

