

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



# PROYECTO DE GRADO

## **SISTEMA DE INFORMACIÓN GEOGRÁFICA PARA ZONOSIS**

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

**POSTULANTE: Johann Stanislao Casias Marquez**

**TUTOR: Lic. Efraín Silva Sánchez**

**REVISOR: Lic. José Luís Zeballos Abasto**

**La Paz - Bolivia  
2009**



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



**LA CARRERA DE INFORMÁTICA DE LA FACULTAD DE CIENCIAS PURAS Y NATURALES PERTENECIENTE A LA UNIVERSIDAD MAYOR DE SAN ANDRÉS AUTORIZA EL USO DE LA INFORMACIÓN CONTENIDA EN ESTE DOCUMENTO SI LOS PROPÓSITOS SON ESTRICTAMENTE ACADÉMICOS.**

**LICENCIA DE USO**

El usuario está autorizado a:

- a) visualizar el documento mediante el uso de un ordenador o dispositivo móvil.
- b) copiar, almacenar o imprimir si ha de ser de uso exclusivamente personal y privado.
- c) copiar textualmente parte(s) de su contenido mencionando la fuente y/o haciendo la referencia correspondiente respetando normas de redacción e investigación.

El usuario no puede publicar, distribuir o realizar emisión o exhibición alguna de este material, sin la autorización correspondiente.

**TODOS LOS DERECHOS RESERVADOS. EL USO NO AUTORIZADO DE LOS CONTENIDOS PUBLICADOS EN ESTE SITIO DERIVARA EN EL INICIO DE ACCIONES LEGALES CONTEMPLADOS EN LA LEY DE DERECHOS DE AUTOR.**

## **DEDICATORIA:**

*A mi Abuela María por sus consejos  
y sus enseñanzas*

*A mi Madre, a mis hermanos y  
familia por su apoyo incondicional*

## **AGRADECIMIENTOS**

Doy gracias a Dios nuestro padre, por fortalecerme, guiarme, y por darme aliento de vida.

A mi madre por el esfuerzo que hizo, de sacarnos adelante, para que sus hijos tengan mejores oportunidades, que ella no tuvo, y a mis hermanos por su ayuda que me brindaron.

A mis Tíos por su apoyo y los consejos que me dieron en los buenos y los malos momentos por los que atravesamos.

A mi familia, primos con quienes compartimos la infancia, la juventud que fueron momentos gratos los que pasamos.

Agradecer y sentirme orgulloso de pertenecer a esta casa de estudio, a la Universidad Mayor de San Andrés y a la Carrera de Informática por formarme como profesional, para ser un bien de la sociedad.

Agradecer a mi tutor la Lic. Nancy Orihuela Sequeiros por sus valiosos consejos y aportes y en especial a mi revisor Lic. José Luís Zeballos Abasto por la paciencia que me tuvo, sus aportes y críticas tan valiosas. Quienes hicieron que se realice este proyecto.

A la Unidad de Zoonosis por brindarme, su atención y amabilidad al proveer de información, sin cuya colaboración, hubiese sido imposible este trabajo.

**johanncasias@yahoo.es**

## ÍNDICE

**Pág.**

### **CAPÍTULO I.**

#### **PROBLEMÁTICA**

1.1. INTRODUCCIÓN .....	1
1.2. ANTECEDENTES .....	2
1.3. PLANTEAMIENTO DEL PROBLEMA.....	7
1.4. OBJETIVOS.....	8
1.4.1. Objetivo General.....	8
1.4.2. Objetivos Específicos.....	8
1.5 JUSTIFICACIÓN .....	9
1.5.1 Justificación Técnica.....	9
1.5.2 Justificación Socia.....	9
1.5.3 Justificación Económica.....	9
1.6 ALCANCES Y LÍMITES .....	10
1.6.1 Alcances .....	10
1.6.2 Limites.....	10
1.7 APORTES .....	10
1.8 MÉTODOS DE INVESTIGACIÓN Y METODOLOGÍA.....	11

### **CAPÍTULO II.**

#### **MARCO TEORICO**

2.1. SISTEMAS DE INFORMACIÓN GEOGRÁFICA SIG.....	12
2.1.1. Definición.....	12
2.1.2. Funciones de un Sistema de información Geográfica.....	12

2.1.3. Componentes de un SIG.....	13
2.1.4. Información que contiene un SIG.....	14
2.1.5. Arquitectura SIG.....	15
2.2. MODELADO DE LENGUAJE UNIFICADO UML.....	15
2.3. PROCESO UNIFICADO RUP .....	16
2.3.1. El Proceso Unificado Dirigido por Casos de Uso .....	16
2.3.2. Proceso Centrado en la Arquitectura.....	17
2.3.3. Proceso Iterativo Incremental .....	20
2.3.4. La Vida del Proceso Unificado.....	23
2.4. ARQUITECTURA CLIENTE/SERVIDOR .....	35
2.5. INGENIERÍA DE USABILIDAD.....	35
2.5.1. Modelos del Análisis y diseño de la Interfaz.....	36
2.5.2. El proceso de diseño de interfaz.....	37
2.6. MÉTRICAS DE CALIDAD .....	38
2.6.1. Funcionalidad.....	39
2.6.2. Confiabilidad .....	40
2.6.3. Facilidad de Uso .....	41
2.6.4. Facilidad de Mantenimiento .....	42
2.6.5 Portabilidad.....	42

## **CAPÍTULO III**

### **MARCO APLICATIVO**

3.1 Introducción.....	43
3.2 Fase Inicial .....	43
3.2.1 Requerimientos del Usuario.....	43
3.2.2 Requerimientos Tecnológicos.....	44
3.3 Fase de Elaboración .....	45
3.3.1 Modelo del Negocio .....	45
a. Casos de Uso Esencial.....	47

b. Casos de Uso Específico .....	49
3.3.2 Diagramas de Secuencia.....	49
3.3.3 Contratos .....	52
3.3.4 Diagramas de colaboración.....	52
3.3.5 Diagramas de estado.....	54
3.3.6 Diagramas de clases .....	57
3.4 Diseño de la Interfaz Gráfica del SIG .....	58
3.5 Fase de Construcción.....	58
3.5.1 Diagrama de Componentes.....	59
3.5.2 Diagrama de Despliegue.....	60
3.5.3 Diagrama Jerárquico de la Interfaz.....	60
3.5.4 Implementación .....	61
3.8 Fase de Transición .....	66

## **CAPÍTULO IV**

### **CALIDAD**

4.1 Introducción.....	67
4.2 Funcionalidad.....	67
4.3 Confiabilidad.....	71
4.4 Facilidad de Uso.....	73
4.5 Facilidad de Mantenimiento.....	74
4.6 Portabilidad .....	76
4.7 Análisis Costo/Beneficio .....	76
4.7.1 Análisis de costos .....	76
4.7.2 Análisis de beneficios.....	78

## **CAPÍTULO V**

### **CONCLUSIONES Y RECOMENDACIONES**

5.1 Conclusiones .....	79
5.2 Recomendaciones .....	80
BIBLIOGRAFIA.....	81

### **ANEXOS**

- ANEXO A.   Árbol de Problemas
- ANEXO B.   Árbol de Objetivos
- ANEXO C.   Casos de Uso Esenciales del SIGZ
- ANEXO D.   Contratos del Sistema SIGZ

## ÍNDICE DE FIGURAS

FIGURA 1.1 Organigrama de la Dirección de Salud .....	3
FIGURA 1.2 Actividad de zoonosis ante un caso de denuncia.....	4
FIGURA 2.1 Funciones de un SIG .....	13
FIGURA 2.2 Componentes de un SIG .....	14
FIGURA 2.3 Atributos de un SIG.....	15
FIGURA 2.4 Los Casos de Uso integran el trabajo .....	16
FIGURA 2.5 Trazabilidad a partir de los Casos de Uso.....	17
FIGURA 2.6 Los modelos se completan, la arquitectura no cambia drásticamente .....	19
FIGURA 2.7 Una iteración RUP.....	21
FIGURA 2.8 Esfuerzo en actividades según fase del proyecto .....	22
FIGURA 2.9 Ciclos, relación, baseline .....	23
FIGURA 2.10 Fases e hitos del Proceso Unificado (RUP) .....	24
FIGURA 2.11 Expresión gráfica del tiempo y esfuerzo dedicados a cada fase de RUP	24
FIGURA 2.12 Diagrama de Colaboración .....	31
FIGURA 2.13 Diagrama de Estados.....	32
FIGURA 2.14 Diagrama de componentes.....	33
FIGURA 2.15 Diagrama de despliegue .....	34
FIGURA 2.16 El proceso de diseño de interfaz .....	37
FIGURA 3.1 Modelo del Negocio SIGZ.....	46
FIGURA 3.2 Caso de uso: Registrar Denuncia.....	47
FIGURA 3.3 Caso de uso: Registrar Mascota .....	48
FIGURA 3.4 Caso de uso: Generar Reporte .....	48
FIGURA 3.5 Caso de uso: Realizar Consulta .....	49
FIGURA 3.6 Diagrama de Secuencia: Registrar Denuncia .....	50
FIGURA 3.7 Diagrama de Secuencia: Registrar Propietario y Mascota.....	50

FIGURA 3.8 Diagrama de Secuencia: Reportes.....	51
FIGURA 3.9 Diagrama de Secuencia: Consultas .....	51
FIGURA 3.10 Diagrama de Colaboración: Registrar Denuncia.....	53
FIGURA 3.11 Diagrama de Colaboración: Registrar Propietario y Mascota.....	53
FIGURA 3.12 Diagrama de Colaboración: Realizar consulta reporte .....	54
FIGURA 3.13 Diagrama de Colaboración: Realizar consulta .....	54
FIGURA 3.14 Diagrama de Estado: Registrar Denuncia.....	55
FIGURA 3.15 Diagrama de Estado: Registrar Propietario y Mascota.....	55
FIGURA 3.16 Diagrama de Estado: Consulta .....	56
FIGURA 3.17 Diagrama de Clases SIGZ.....	57
FIGURA 3.18 Ventana ArcView.....	58
FIGURA 3.19 Mapa catastral de El Alto.....	58
FIGURA 3.20 Diagrama de componentes SIGZ .....	59
FIGURA 3.21 Diagrama de componentes comunes.....	59
FIGURA 3.22 Diagrama de despliegue a nivel de instancia del Sistema .....	60
FIGURA 3.23 Diagrama jerárquico que muestra el prototipo inicial de la interfaz del Sistema de Información Geográfica para Zoonosis .....	61
FIGURA 3.24 Pantalla de Ingreso al Sistema.....	62
FIGURA 3.25 Interfaz del menú principal.....	62
FIGURA 3.26 Formulario de Denuncias.....	63
FIGURA 3.27 Consulta de búsqueda de Zonas .....	63
FIGURA 3.28 visualización del mapa.....	64
FIGURA 3.29 Formulario Propietario.....	65
FIGURA 3.30 Formulario de Matrícula.....	66
FIGURA 4.1 Diagrama de transferencia de módulos.....	71

## ÍNDICE DE TABLAS

TABLA 1.1 Proyectos de Grado que hacen uso del SIG .....	6
TABLA 1.2 Proyectos de Grado que hacen uso del SIG .....	7
TABLA 2.1 Duración y esfuerzo en cada fase a lo largo del proyecto .....	24
TABLA 2.2 Fase Inicio .....	25
TABLA 2.3 Fase de Elaboración .....	26
TABLA 2.4 Plantilla de un caso de uso expandido .....	28
TABLA 2.5 Plantilla de un Contrato .....	31
TABLA 2.6 Fase de Construcción .....	33
TABLA 2.7 Proyectos de Grado que hacen uso del SIG .....	35
TABLA 2.8 Calculo de Puntos Función .....	39
TABLA 2.9 Calculo de ajuste de complejidad .....	40
TABLA 2.10 Calculo de ajuste de complejidad .....	41
TABLA 3.1 Contrato: Formulario de denuncia .....	52
TABLA 4.1 Descripción de parámetros de medición .....	68
TABLA 4.2 Calculo de Puntos Función .....	69
TABLA 4.3 Calculo de los valores de ajuste de la complejidad .....	70
TABLA 4.4 Calculo de la confiabilidad de cada modulo .....	72
TABLA 4.5 Calculo de Puntos Función .....	73
TABLA 4.6 Métrica IMS .....	75
TABLA 4.7 Valores del IMS en cada versión .....	75
TABLA 4.8 Costo de instalación .....	77
TABLA 4.9 Costo de construcción .....	77
TABLA 4.10 Costo Total .....	77
TABLA 4.11 Cuadro de los Beneficios con el SIGZ .....	78

## RESUMEN

El presente Proyecto de Grado “Sistema de Información Geográfica para Zoonosis (SIGZ)”, surge de la necesidad de contar con mecanismos que ayuden a mejorar el control de **La Rabia** y los Animales de la Ciudad de El Alto; siendo importante identificar las zonas territoriales donde se propagan o se encuentren brotes de enfermedades zoonóticas.

Para ello se hará uso de la herramienta ArcView\_GIS para representar datos georreferenciados. ArcView es una de las herramientas de la familia de productos ArcGIS, está destinada a la explotación de la información geográfica. Esta herramienta permite explorar, visualizar la gestión de los datos, manipular metadatos, confeccionar mapas temáticos, consultas, y otros. Dispone también de herramientas para la creación y mantenimiento de mapas, con algunas limitaciones en lo que se refiere a datos espaciales y coberturas.

El objetivo es implementar un Sistema Información Geográfica, que mejore la toma de decisiones de esta institución de Zoonosis, para ello se utilizó la metodología RUP es una metodología sólida, con documentación, que apoya el ciclo de vida evolutivo incremental, además de orientarse al desarrollo de componentes apoyando el desarrollo orientado a objetos, que a la vez hace uso del Lenguaje Unificado de Modelado (Unified Modeling Language, UML); y la herramienta case Rational Rose, el que se utilizará para modelar los procesos que realiza el Sistema.

El proceso de software tiene tres aspectos fundamentales: dirigidos por casos de uso, está centrado en la arquitectura, y es iterativo incremental. El Proceso Unificado es más que un simple proceso; es un marco de trabajo genérico que se adapta a una gran variedad de sistemas de software, diferentes tipos de aplicación, diferentes tipos de organizaciones y distintos tipos de tamaño de software.

# CAPÍTULO I

## MARCO INTRODUCTORIO

### 1.1 INTRODUCCIÓN

Zoonosis es el área que se encarga de controlar las enfermedades o zoonosis propias de los animales vertebrados que a veces se transmiten al hombre. Tanto los animales como el hombre contraen la infección del suelo, el agua, animales invertebrados y plantas; los animales, como regla, no juegan un papel esencial en el ciclo vital del agente etiológico, pero pueden contribuir, a la distribución y transmisión de las infecciones.

El número de las infecciones encontradas en el Hombre y el Animal es de 200 enfermedades y por si no fuera poco el número de las zoonosis aumenta a medida que se incrementa los conocimientos que aportan las diferentes disciplinas medicobiológicas. Nuevas enfermedades zoonóticas aparecen continuamente, con la incorporación de la actividad humana en nuevos territorios que contienen focos naturales de infección. Las enfermedades zoonóticas más casuales a tratar por su importancia en nuestro medio son: la rabia, la fiebre aftosa, la gripe aviar y otros [AS97].

Es importante poder identificar las zonas territoriales donde se propagan o se encuentren brotes de enfermedades, para poder hacer frente a esta situación es necesario el uso de un sistema de información geográfica (SIG). El SIG es indispensable en procesos que ayudan a una mejor toma de decisiones, y así se podrá preservar la salud de las personas de esta ciudad de El Alto.

Un sistema de información geográfica es una herramienta computacional para el mapeo y el análisis de los elementos y eventos que ocurren en la tierra. La tecnología SIG integra operaciones comunes de bases de datos, tales como consultas y análisis estadísticos, con los beneficios propios de la visualización y el análisis geográfico que ofrecen las cartografías. Estas capacidades distinguen los SIG de otros sistemas de información y lo hacen

más valiosos para empresas públicas y privadas de diversa índole para explicar eventos, predecir salidas y en la planificación de estrategias.<sup>1</sup>

En el presente proyecto se aplicará el Sistema de Información Geográfica para Zoonosis (SIGZ), esto se realizará para mejorar la información de los distritos o zonas donde exista brotes de enfermedades zoonóticas, recopilación de datos de los animales que porten esta enfermedad que es la rabia el cual afecta a la población de la ciudad de El Alto, y así mejorar la calidad de vida de las personas.

## 1.2 ANTECEDENTES

### 1.2.1 La Unidad de Zoonosis

La Unidad de Zoonosis es creada en 1992 en una pequeña oficina, para el control de Canes vagabundos, con el tiempo el Gobierno Municipal implementa un refrigerador para conservar las vacunas biológicas y tenerlas a la mano durante todo el año además de termos de manejo de vacunas, cintas y carnets para las Diferentes Campañas.

La Unidad de Zoonosis de la Ciudad de El Alto realiza la orientación sobre la tenencia responsable de mascotas, sus cuidados, vacunas, recepción de mordeduras, vigilancia, eutanasia y control de mascotas peligrosas y conciliación de gastos de curación entre las partes afectadas.

Se han definido las zoonosis como ***“aquellas enfermedades que se transmiten de los animales vertebrados al hombre y viceversa”***.

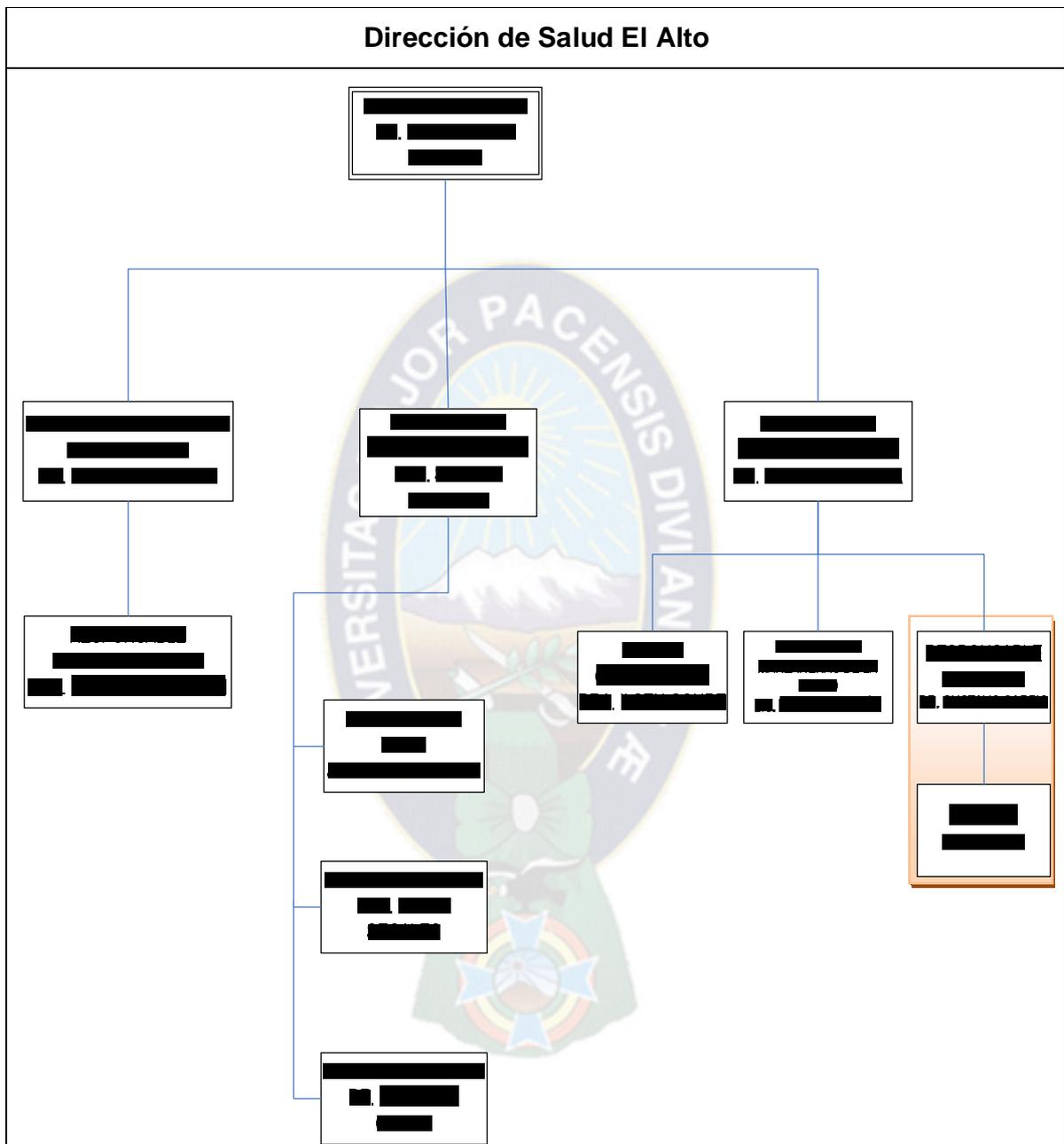
#### Y Servicios que presta zoonosis

Esta unidad a parte de los servicios ya mencionados, tiene el rol de orientar sobre la tenencia de mascotas, vacunación, recepción de denuncias de mordeduras, vigilancia, control de enfermedades zoonóticas y eutanasia de canes potencialmente peligrosos. El control de Alimentos de Origen Animal (teniasis, cisticercosis en carne de cerdo y de otros animales) o en mal estado en coordinación con OCDECO.

---

<sup>1</sup> Fuente: Environmental Systems Research Institute, Inc.

## Y Organización de la Institución

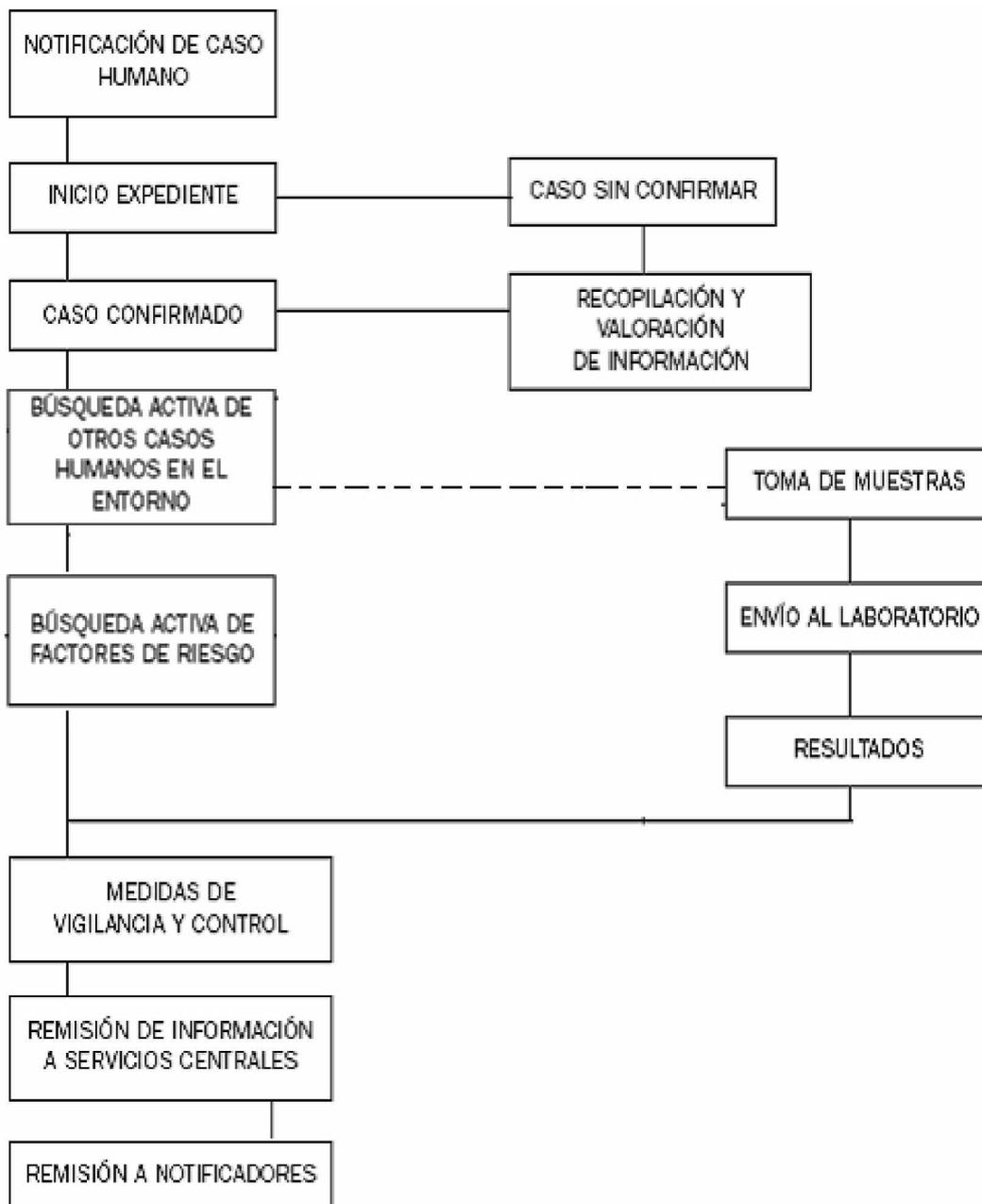


**Figura 1.1 Organigrama de la Dirección de Salud**

**Fuente: Dirección de Salud**

### Y Actividades que realiza zoonosis.

Actividades que Zoonosis realiza ante cualquier caso que se presente



**Figura 1.2 Actividad de zoonosis ante un caso de denuncia**  
**Fuente: Elaboración Propia**

## **1.2.2 Sistemas de Información Geográfica (SIG)**

A principios de los años 80's, los Sistemas de Información Geográfica (SIG) se habían convertido en un modelo plenamente operativo, a medida que la tecnología de cómputo se perfeccionaba, se hacía menos costosa y gozaba de una mayor aceptación. Actualmente se están instalando rápidamente estos sistemas en los organismos públicos, los laboratorios de investigación, las instituciones académicas, la industria privada y las instalaciones militares y públicas.

A continuación se describen brevemente algunas de sus aplicaciones principales:

### **-Infraestructura**

Algunos de los primeros sistemas SIG fueron utilizados por las empresas encargadas del desarrollo, mantenimiento y administración de redes de electricidad, gas, agua, teléfono, alcantarillado, etc.

### **-Medio ambiente**

Implementadas por instituciones de medio ambiente, que facilitan la evaluación del impacto ambiental en la ejecución de proyectos.

### **-Recursos mineros**

El diseño de estos SIG facilita el manejo de un gran volumen de información generada en varios años de explotación intensiva de un banco minero

### **-Banca**

Los bancos son buenos usuarios de los SIG debido a que requieren ubicar a sus clientes y planificar tanto sus campañas como la apertura de nuevas sucursales incluyendo información sobre las sucursales de la competencia.

## **1.2.1 Algunas Entidades en el Mundo que utilizan el SIG**

*UNESCO-RAPCA* para la evaluación de amenazas y riesgos de los desastres naturales. El Pentágono para controlar espacios aéreos sus bases militares, etc., Las Petroleras para realizar búsquedas de petróleo, gas natural y otros.

### 1.2.2 Trabajos Similares en los países de latino América y el mundo

<b>Título</b>	<b>AUTOR</b>	<b>DESCRIPCIÓN</b>
Aplicación de Los Sistemas de Información Geográfica A La Optimización de los Residuos Sólidos Urbanos	Jorge Rubén Gómez Cámara, Ángel Tajadura de la Torre, Ignacio Fontaneda Gonzáles.  “Universidad de Burgos”, España	El presente trabajo consiste en el estudio de mejora en la recogida de residuos sólidos urbanos (R. S. U.) implementado dentro de un Sistema de Información Geográfica.
El uso de los Sistemas de Información geográfica (sig) en el Análisis Demográfico de situaciones de Desastre	Angélica Reyna Universidad Autónoma del Estado de Hidalgo, México.	El presente documento tiene dos objetivos fundamentales: aportar algunas reflexiones metodológicas sobre la aplicación de los Sistemas de Información Geográfica (SIG) al análisis demográfico de situaciones de desastre y mostrar sus resultados prácticos en el análisis de las inundaciones ocurridas en México en el año 1999.
Aplicación de los SIG a la Previsión de la Demanda Eléctrica	Ignacio Zabalza Bribián, José Antonio Domínguez Navarro, José Luis Bernal Agustín, José María Yusta Loyo  Dpto. de Ingeniería Eléctrica Universidad de Zaragoza, España	Presenta la aplicación de un Sistema de Información Geográfica (SIG) vectorial como herramienta básica para la previsión de la demanda de energía eléctrica a largo plazo.
SIGMAPA - Sistema de Información Geográfica Corporativo del Ministerio de Agricultura, Pesca y Alimentación	María Soledad Gómez Andrés, España.	Desde hace casi dos décadas el Ministerio de Agricultura, Pesca y Alimentación viene utilizando tecnologías basadas en los Sistemas de Información Geográfica (SIG) para la gestión, explotación y planificación de los recursos agrícolas, pesqueros y ganaderos del territorio español.

**Tabla 1.1 Proyectos de Grado que hacen uso del SIG**  
Fuente de elaboración propia

### 1.2.3 Trabajos Similares en la carrera de Informática de la UMSA

TÍTULO	AUTOR	DESCRIPCION
Sistemas de Información Geográfica como apoyo en el Manejo de Desastres-	Catari Choquehuanca Marisol	El presente documento fue desarrollado para la trata de los desastres naturales.
Sistemas de Información Geográfica (SIG) para la Localización de Contaminantes Orgánicos Persistentes (COP's)	Aguilar Mollericona Juan Eduardo	El presente proyecto fue desarrollado para el seguimiento y controlar de este tipo de contaminantes que afectan al ecosistema.
Sistemas de Información Geográfica aplicado a seguridad ciudadana	Andrade Zurita Brigner Celi	La aplicación de este proyecto es para tener mejor información de los distritos y las zonas donde exista riesgo.
Sistemas de Información Geográfica de La Pobreza en Bolivia	Garflas P. Sandra, Pinto A Jorge	Fue implementado para tener datos georreferenciales, datos estadísticos en que departamentos existe mas índices de pobreza y tomar acciones inmediatas o posteriores
Sistemas de Información Geográfica de Control y Seguimiento de la Planificación y Gestión del Desarrollo Territorial Caso: Sub Alcaldía Cotahuma.	Silvestre Vila Dora Virginia	Fue desarrollado para administrar de manera eficiente la información generada en macro distritos de cotahuma.
Sistemas de Información Geográfica de las Redes de Servicio de Salud.	Jiménez Paco Freddy	La aplicación de este proyecto permite generar información de los servicios de salud donde se ubican, sobre la infraestructura, en que red esta localizado.

**Tabla 1.2 Proyectos de Grado que hacen uso del SIG**  
Fuente de elaboración propia

### 1.3 PLANTEAMIENTO DEL PROBLEMA

No se puede realizar un control de todos los animales, ya que es difícil abarcar en su totalidad La Ciudad de El Alto y peor aún erradicar estas enfermedades que ocasionan pérdidas en la economía de la Salud, impacto social y en algunos casos de mortalidad.

Lo que se hace actualmente para controlar las zoonosis es realizar un seguimiento de cada caso particular, observación de canes y/o animales involucrados en la incidencia de una

determinada enfermedad. También se controla con medicamentos, con vacunas, prevención de estas enfermedades con campañas anuales.

Hoy en día se sigue reportando este tipo de casos de rabia y otras enfermedades zoonóticas que afecta al Hombre, se pudo ver que en algunos casos los animales que están cerca de las personas son portadores de muchas enfermedades que afecta a la población.

Se pudo percibir los siguientes problemas.

- Y No existe un registro adecuado de los animales ni siquiera de los animales domésticos ya que se lo hace de manera manual en Zoonosis de la ciudad de El Alto.
- Y No existe información adecuada de las zonas donde se percibe que existe brote de enfermedades zoonóticas de los animales. Particularmente de la rabia.
- Y No se tiene información de los animales que portan este virus siendo peligroso para la población en su conjunto.
- Y No existe información adecuada de las zonas de donde se perciben que fueron agredidos por canes que tuvieran este virus.

## **1.4 OBJETIVOS**

### **1.4.1 Objetivo General**

Implementar el Sistema de Información Geográfica para Zoonosis (SIGZ) para mejorar la toma de decisiones, siendo útil para identificar las Zonas donde existe denuncias de casos de Rabia.

### **1.4.2 Objetivos Específicos**

- Y Almacenar la información requerida de las personas que fueron agredidos por canes, generando ventajas, reduciendo esfuerzos, manteniendo en orden los datos de los Denunciantes.
- Y Integrar la información geográfica de los mapas.
- Y Proporcionar información de las zonas donde existe enfermedades Zoonóticas.

- Y Guardar la información de los Propietarios y sus Mascotas.
- Y Mejorar la información que permita, registrar, actualizar y proporcionar información y consultas de los Animales.
- Y Efectuar el seguimiento de los Animales, enfermos.
- Y Características de la información:
  - a. Información oportuna
  - b. Información rápida
  - c. Información Segura
  - d. Información Integra

## **1.5 JUSTIFICACIÓN**

### **1.5.1 Justificación Técnica**

El uso y el aumento en la tecnología de un SIGZ que es indispensable para controlar las zonas que pueden ser afectados por la rabia canina u otras enfermedades de los animales transmisibles al hombre, y esta herramienta ayudará al momento de realizar informes, para una mejor toma de decisiones.

### **1.5.2 Justificación Social**

El SIGZ también reducirá el tiempo al momento de realizar sus principales tareas y a la toma de decisiones, en el caso de Zoonosis para mejorar el control de las enfermedades epidemiológicas que existe en los Animales, estas enfermedades son peligrosas y afectan ala población en su conjunto y así se podrá mejorar la calidad de vida.

### **1.5.3 Justificación Económica**

El proyecto permitirá que la institución de Zoonosis mejore sus principales tareas que realiza de manera manual, reduciendo las horas de trabajo del personal encargado de la unidad. Y así se puede lograr reducir el tiempo en averiguar que distritos o zonas donde se presume que se propaga enfermedades.

## **1.6 LÍMITES Y ALCANCES**

### **1.6.1 Alcances**

El Sistema de Información Geográfica para Zoonosis (SIGZ), comprenderá procesos de control de toda la información, el cual tendrá los siguientes alcances.

Lograr visualizar el material geográfico mostrando la información referente a los requerimientos del usuario.

La información obtenida será a nivel de Zonas de la ciudad de El Alto.

La obtención de la información la realizará mediante la autorización del personal autorizado.

En cuanto al control de las enfermedades Zoonóticas solo se realizara el control de lo que es la rabia en los animales domésticos como ser los canes y felinos (gatos).

### **1.6.2 Limites**

- Y No se podrá realizar el registro de todos los animales, y el registro de todos los casos de agresión a personas por mordedura de canes.
- Y El Sistema de Información Geográfica para Zoonosis es para el uso y la toma de decisiones de Zoonosis de la ciudad de El Alto.
- Y El sistema solo estará dirigida al control de las personas que fueron agredidos por Animales y al Control de los mismos.
- Y La actualización de datos la realizara solo el personal autorizado.

## **1.7 APORTES**

### **1.7.1 Aporte Social**

Los aportes que ofrecerá este Proyecto de Grado con el Sistema de Información Geográfica para Zoonosis es la de colaborar a la toma de decisiones según el momento específico, generando información confiable, mejorando la vigilancia de las enfermedades como es la rabia en los animales domésticos, latentes en esta ciudad.

## 1.8 MÉTODOS DE INVESTIGACIÓN Y METODOLOGÍA

En el presente perfil de proyecto de grado se ha utilizado la siguiente metodología:

- Y El Método de Marco Científico durante el proceso de investigación utilizando técnicas como:
  - a) Árbol de Problemas
  - b) Árbol de Objetivos
- Y Técnicas de Ingeniería de Software como ser:
  - a) Recopilación de información, mediante encuestas y entrevistas directas.
  - b) Recopilación de información y bibliografía acerca de su construcción y sus aplicaciones.
- Y La metodología RUP Y UML utilizada para la elaboración y el desarrollo de los módulos.
  - a) Diseño de base de datos confiable.
  - b) Para el diseño de la interfaz del sistema, se utilizara la ingeniería de usabilidad más concretamente la metodología iterativa o espiral.

## CAPÍTULO II

### MARCO TEÓRICO

#### 2.2 SISTEMAS DE INFORMACIÓN GEOGRÁFICA – SIG –

Los sistemas de información geográfica SIG se han convertido en una herramienta muy empleada en distintas áreas. Facilitando el análisis de la información que se posee con el propósito de optimizar la oportuna toma de decisiones, sobre planificación, medio ambiente, recursos naturales, transporte, instalaciones civiles, y otros.

##### 2.2.1 Definición.

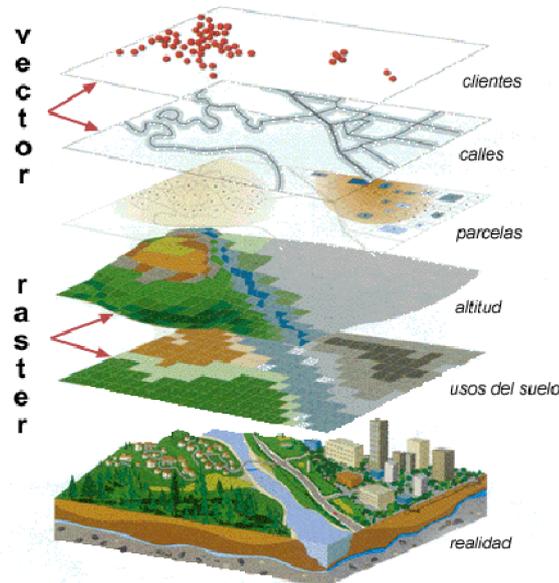
Un SIG se define como un conjunto de métodos, herramientas y datos; integrado para trabajar con información espacial, siendo un herramienta esencial para el análisis en muchas áreas vitales de una nación contribuyendo así en su desarrollo.

- i. **Información Geográfica.** Es una abstracción o representación de la realidad geográfica (paisaje).
- ii. **Geografía.** Descripción de los elementos que forman el ambiente.
  - Ø **Ambiente natural** - topografía, Hidrografía, formaciones geológicas, tipos de suelo, y otros.
  - Ø **Ambiente humano** – ciudades, edificaciones, vías de transporte, dotaciones (hospitales, escuelas, y otros).

##### 2.2.2 Funciones de un Sistema de información Geográfica

- Ø **Entrada de datos.** Proceso que realiza un sistema de información, utilizando los datos que requiere para posteriormente procesar dicha información.
- Ø **Almacenamiento de los datos.** Una de las capacidades mas importantes que poseen las computadoras, y que gracias a la misma se puede manipular la información guardada que se efectúa rutinariamente.

Ø **Proceso de los datos.** Es el componente mas importante de un SIG, porque utiliza la información disponible para generar una nueva información



**Figura 2.1 Funciones de un SIG**  
Fuente: [WEB03]

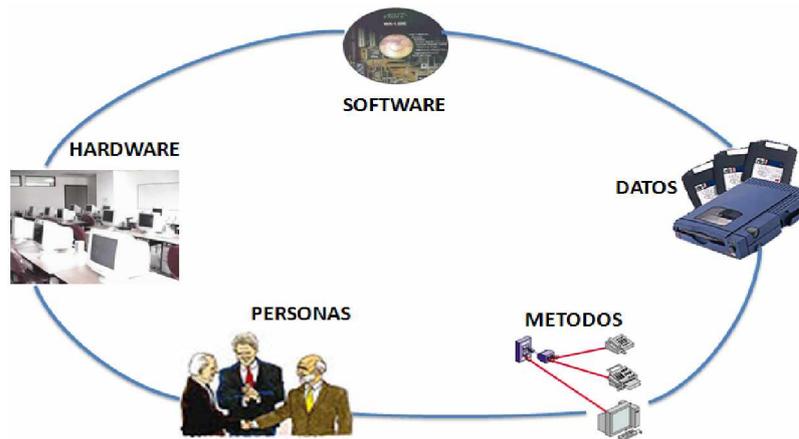
### 2.2.3 Componentes de un SIG

Los componentes de un SIG son: Equipos (Hardware), Programas (Software), datos, recurso humano y métodos.

- a) **Equipos (Hardware).** Son los recursos en el que opera un SIG. Los cuales se ejecutan en amplio rango de tipos de computadores, como son los Servidores, computadores portátiles, utilizados individualmente o en red [WEB03]
- b) **Programas (Software).** Los programas SIG provee las herramientas y las funciones necesarias para almacenar, analizar y desplegar la información geográfica. Los componentes principales del software son:
  - Y Un sistema de manejo de base de datos(DBMS)
  - Y Herramientas de soporte de consultas, análisis y visualización de datos Geográficos.

Y Una Interfaz gráfica para el usuario (GUI) para el fácil acceso a herramientas. [WEB03].

- c) **Datos.** Es importante tener una buena información para un SIG. Los datos geográficos y tabulares pueden ser adquiridos por quien implementa el sistema de información, así como terceros que tienen datos a disponibilidad [WEB03].



**Figura 2.2 Componentes de un SIG**  
Fuente: [WEB03]

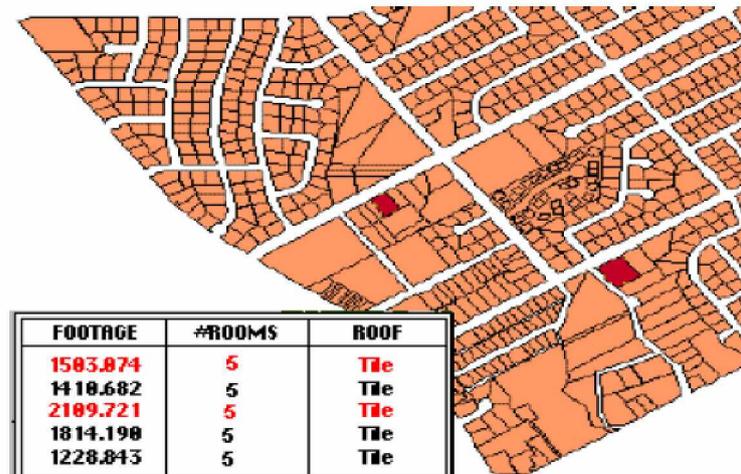
- d) **Recurso Humano.** La tecnología del SIG son de valor limitado si no se cuenta con el personal adecuado para administrar el sistema, los cuales establecen para aplicarlos en problemas del mundo real.

### 2.2.2 Información que contiene un SIG

Y **Referencia geográfica explícita.** Tal como una latitud y longitud, o una referencia implícita como domicilio, nombre de la calle, zona, avenida.

Y **Referencia implícita.** Pueden ser derivadas de las explícitas.

A todo objeto se asocian atributos



**Figura 2.3 Atributos de un SIG**  
Fuente: [WEB03]

### 2.2.3 Arquitectura SIG

Se fundamenta en una arquitectura genérica cliente servidor, que esta formado por la entrada de datos, administración de base de datos, análisis de los datos geográficos la representación y la salida de los datos.

### 2.3 MODELADO DE LENGUAJE UNIFICADO

Es una herramienta que haciendo uso de diagramas, permite representar estructuras de software estáticas, y dinámicas, para que sean fáciles de comprender y comunicarlas a otras personas. También sirve de ayuda al proceso de análisis de un problema, y a la toma de decisiones para construcción de un sistema [JBR00].

El lenguaje de modelado unificado diseñado para visualizar, especificar, construir y documentar partes de un sistema de software, prescribe un conjunto de notaciones y diagramas estándar para modelar sistemas orientados a objetos y describe la semántica esencial de lo que estos diagramas y símbolos significan. Por ello es un buen sistema de documentar partes del código que deban ser reutilizables por otros desarrolladores.

## 2.4 PROCESO UNIFICADO RUP

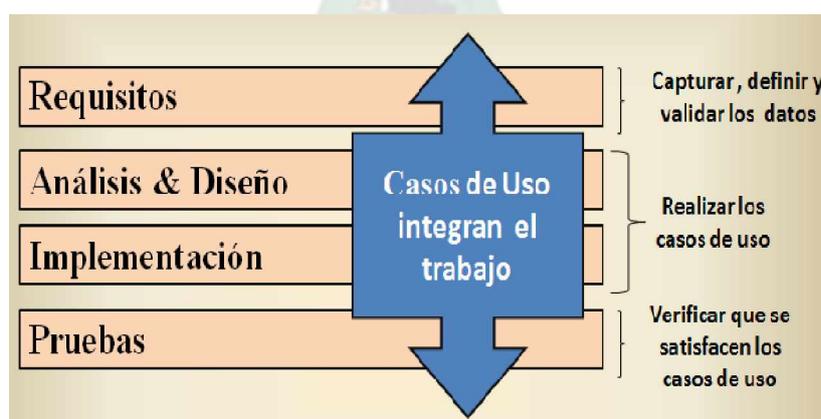
RUP es una metodología sólida, con documentación, que apoya el ciclo de vida evolutivo incremental, además de orientarse al desarrollo de componentes apoyando el desarrollo orientado a objetos.

RUP captura algunas de las mejores prácticas de la industria para el desarrollo de software las cuales son para desarrollar el software en iteraciones, administrar requerimientos, usar arquitecturas basadas en componentes, verificar la calidad del software, controlar los cambios al software.

### 2.4.1 El Proceso Unificado Dirigido por Casos de Uso

Los Casos de Uso son una técnica de captura de requisitos que fuerza a pensar en términos de importancia para el usuario y no sólo en términos de funciones que sería bueno contemplar. Se define un Caso de Uso como un fragmento de funcionalidad del sistema que proporciona al usuario un valor añadido. Los Casos de Uso representan los requisitos funcionales del sistema.

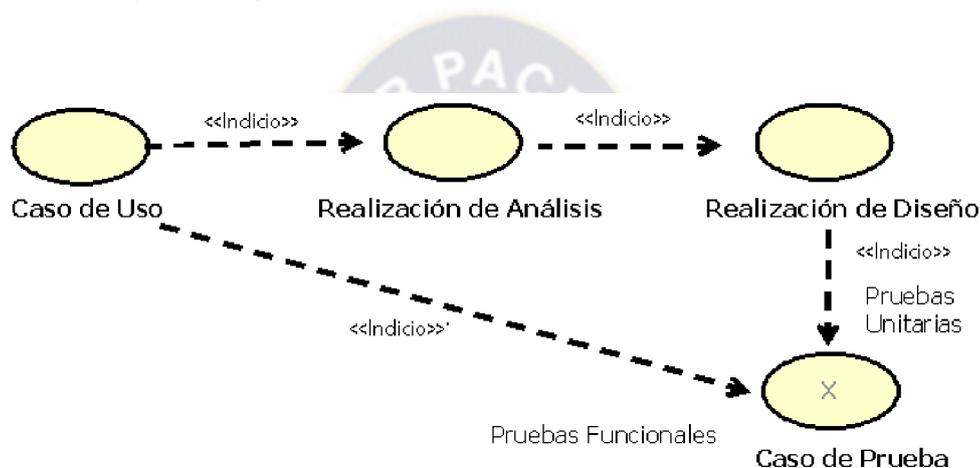
En RUP los Casos de Uso no son sólo una herramienta para especificar los requisitos del sistema. También guían su diseño, implementación y prueba. Los Casos de Uso constituyen un elemento integrador y una guía del trabajo como se muestra en la Figura 2.4



**Figura 2.4 Los Casos de Uso integran el trabajo**  
Fuente: [K96]

Los Casos de Uso no sólo inician el proceso de desarrollo sino que proporcionan un hilo conductor, permitiendo establecer trazabilidad entre los artefactos que son generados en las diferentes actividades del proceso de desarrollo.

Figura 2.5 Basándose en los Casos de Uso se crean los modelos de análisis y diseño, luego la implementación que los lleva a cabo, y se verifica que efectivamente el producto implemente adecuadamente cada Caso de Uso. Todos los modelos deben estar sincronizados con el modelo de Casos de Uso.



**Figura 2.5 Trazabilidad a partir de los Casos de Uso**  
**Fuente: (PG03)**

#### 2.4.2 Proceso Centrado en la Arquitectura

La arquitectura de un sistema es la organización o estructura de sus partes más relevantes, lo que permite tener una visión común entre todos los involucrados (desarrolladores y usuarios) y una perspectiva clara del sistema completo, necesaria para controlar el desarrollo.

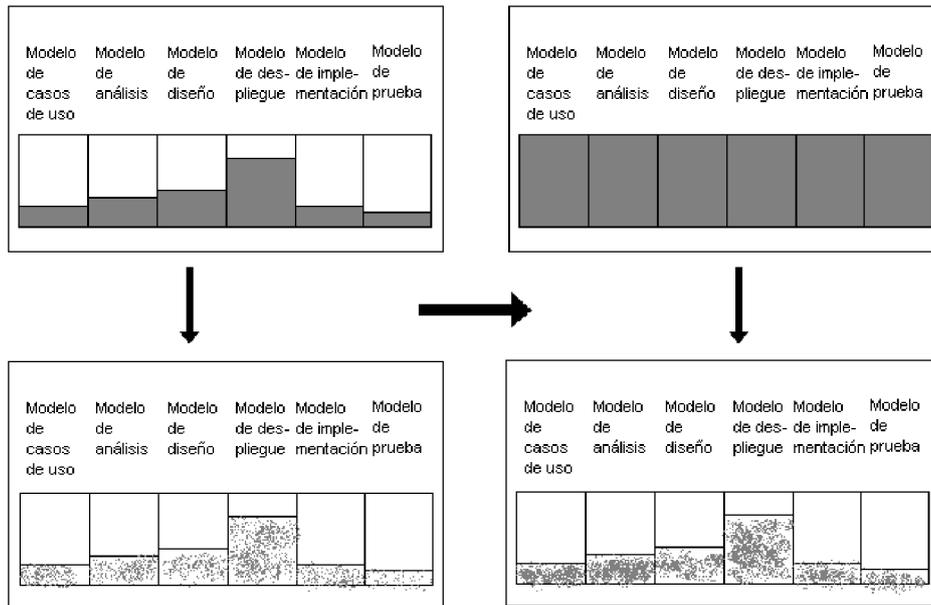
La arquitectura involucra los aspectos estáticos y dinámicos más significativos del sistema, está relacionada con la toma de decisiones que indican cómo tiene que ser construido el sistema y ayuda a determinar en qué orden. Además la definición de la arquitectura debe tomar en consideración elementos de calidad del sistema, rendimiento, reutilización y capacidad de evolución por lo que debe ser flexible durante todo el proceso de desarrollo. La

arquitectura se ve influenciada por la plataforma software, sistema operativo, gestor de bases de datos, protocolos, consideraciones de desarrollo como sistemas heredados. Muchas de estas restricciones constituyen requisitos no funcionales del sistema.

En el caso de RUP además de utilizar los Casos de Uso para guiar el proceso se presta especial atención al establecimiento temprano de una buena arquitectura que no se vea fuertemente impactada ante cambios posteriores durante la construcción y el mantenimiento.

Cada producto tiene tanto una función como una forma. La función corresponde a la funcionalidad reflejada en los Casos de Uso y la forma la proporciona la arquitectura. Existe una interacción entre los Casos de Uso y la arquitectura, los Casos de Uso deben encajar en la arquitectura cuando se llevan a cabo y la arquitectura debe permitir el desarrollo de todos los Casos de Uso requeridos, actualmente y en el futuro. Esto provoca que tanto arquitectura como Casos de Uso deban evolucionar en paralelo durante todo el proceso de desarrollo de software.

Es conveniente ver el sistema desde diferentes perspectivas para comprender mejor el diseño por lo que la arquitectura se representa mediante varias vistas que se centran en aspectos concretos del sistema, abstrayéndose de los demás. Para RUP, todas las vistas juntas forman el llamado modelo 4+1 de la arquitectura, el cual recibe este nombre porque lo forman las vistas lógica, de implementación, de proceso y de despliegue, más la de Casos de Uso que es la que da cohesión a todas.



**Figura 2.6 Los modelos se completan, la arquitectura no cambia drásticamente**  
**Fuente: [JBR00]**

Al final de la fase de elaboración se obtiene una *baseline*<sup>2</sup> de la arquitectura donde fueron seleccionados una serie de Casos de Uso arquitectónicamente relevantes (aquellos que ayudan a mitigar los riesgos más importantes, aquellos que son los más importantes para el usuario y aquellos que cubran las funcionalidades significativas)

Durante la construcción los diversos modelos van desarrollándose hasta completarse (según se muestra con las formas rellenas en la esquina superior derecha). La descripción de la arquitectura sin embargo, no debería cambiar significativamente (abajo a la derecha) debido a que la mayor parte de la arquitectura se decidió durante la elaboración. Se incorporan pocos cambios a la arquitectura (indicados con mayor densidad de puntos en la figura inferior derecha).

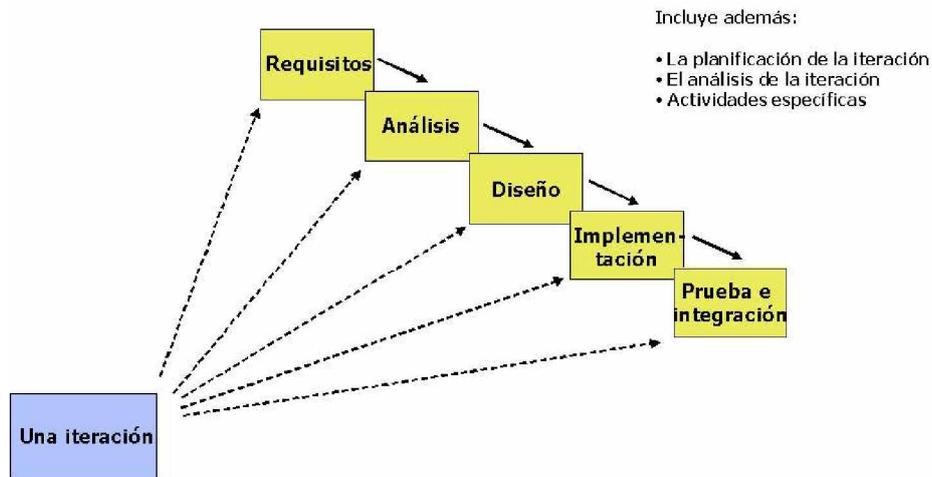
<sup>2</sup> Una baseline es una instantánea del estado de todos los artefactos del proyecto, registrada para efectos de gestión de configuración y control de cambios.

### 2.4.3 Proceso Iterativo Incremental

En primer lugar la iteración trata un grupo de casos de uso que juntos amplían la utilidad del producto desarrollado hasta ahora. En segundo lugar, la iteración trata los riesgos más importantes. Las iteraciones sucesivas se construyen sobre los artefactos de desarrollo tal como quedaron al final de la última iteración. Al ser miniproyectos, comienzan con los casos de uso y continúan a través del trabajo de desarrollo subsiguiente análisis, diseño, implementación y prueba. Especialmente en las primeras fases del ciclo de vida [I. Jacobson, 2000].

Para esto, la estrategia que se propone en RUP es tener un proceso iterativo e incremental en donde el trabajo se divide en partes más pequeñas o mini proyectos. Permitiendo que el equilibrio entre Casos de Uso y arquitectura se vaya logrando durante cada mini proyecto, así durante todo el proceso de desarrollo. Cada mini proyecto se puede ver como una iteración (un recorrido más o menos completo a lo largo de todos los flujos de trabajo fundamentales) del cual se obtiene un incremento que produce un crecimiento en el producto.

Una iteración puede realizarse por medio de una cascada como se muestra en la Figura 2.7 Se pasa por los flujos fundamentales (Requisitos, Análisis, Diseño, Implementación y Pruebas), también existe una planificación de la iteración, un análisis de la iteración y algunas actividades específicas de la iteración. Al finalizar se realiza una integración de los resultados con lo obtenido de las iteraciones anteriores.

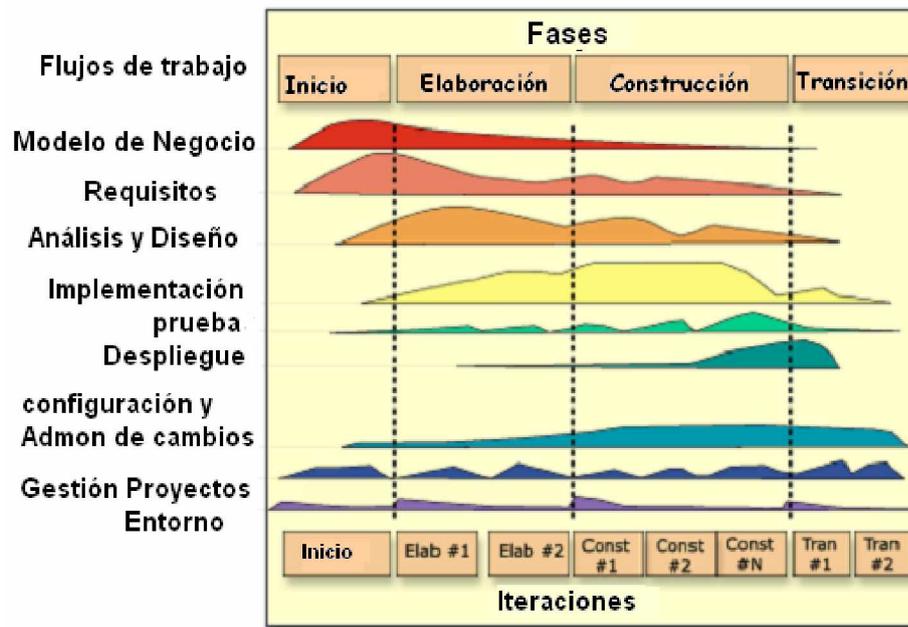


**Figura 2.7 Una iteración RUP**

**Fuente: (K96)**

El proceso iterativo e incremental consta de una secuencia de iteraciones. Cada iteración aborda una parte de la funcionalidad total, pasando por todos los flujos de trabajo relevantes y refinando la arquitectura. Cada iteración se analiza cuando termina. Se puede determinar si han aparecido nuevos requisitos o han cambiado los existentes, afectando a las iteraciones siguientes. Durante la planificación de los detalles de la siguiente iteración, el equipo también examina cómo afectarán los riesgos que aún quedan al trabajo en curso. Toda la retroalimentación de la iteración pasada permite reajustar los objetivos para las siguientes iteraciones. Se continúa con esta dinámica hasta que se haya finalizado por completo con la versión actual del producto.

RUP divide el proceso en cuatro fases, dentro de las cuales se realizan varias iteraciones en número variable según el proyecto y en las que se hace un mayor o menor hincapié en las distintas actividades. En la Figura 2.8 se muestra cómo varía el esfuerzo asociado a las disciplinas según la fase en la que se encuentre el proyecto RUP.



**Figura 2.8 Esfuerzo en actividades según fase del proyecto**

**Fuente: [JBR00]**

Las primeras iteraciones (en las fases de Inicio y Elaboración) se enfocan hacia la comprensión del problema y la tecnología, la delimitación del ámbito del proyecto, la eliminación de los riesgos críticos, y al establecimiento de una *baseline* de la arquitectura. Durante la fase de inicio las iteraciones hacen poner mayor énfasis en actividades modelado del negocio y de requisitos.

En la fase de elaboración, las iteraciones se orientan al desarrollo de la *baseline* de la arquitectura, abarcan más los flujos de trabajo de requerimientos, modelo de negocios (refinamiento), análisis, diseño y una parte de implementación orientado a la *baseline* de la arquitectura.

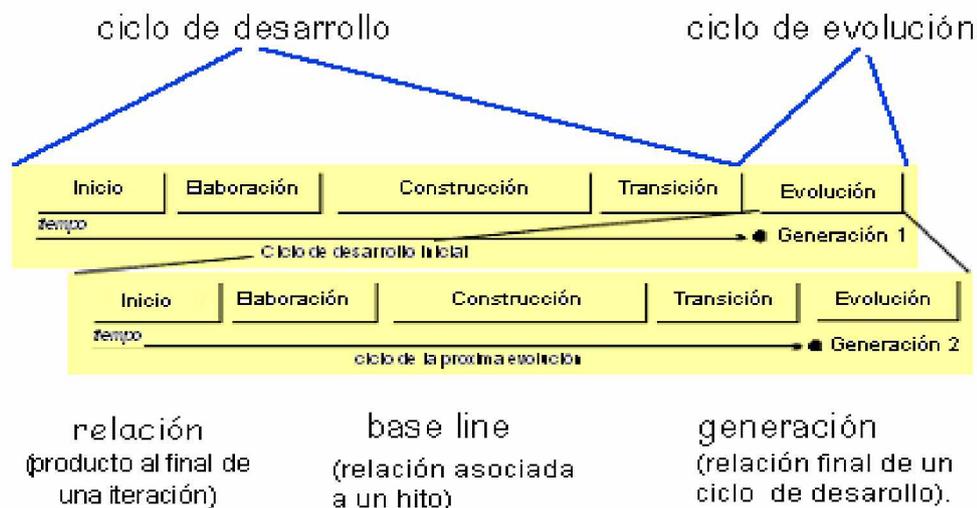
En la fase de construcción, se lleva a cabo la construcción del producto por medio de una serie de iteraciones. Para cada iteración se selecciona algunos Casos de Uso, se refina su análisis y diseño y se procede a su implementación y pruebas. Se realiza una pequeña cascada para cada ciclo. Se realizan tantas iteraciones hasta que se termine la implementación de la nueva versión del producto.

En la fase de transición se pretende garantizar que se tiene un producto preparado para su entrega a la comunidad de usuarios.

Como se puede observar en cada fase participan todas las disciplinas, pero que dependiendo de la fase el esfuerzo dedicado a una disciplina varía.

#### 2.4.4 La Vida del Proceso Unificado

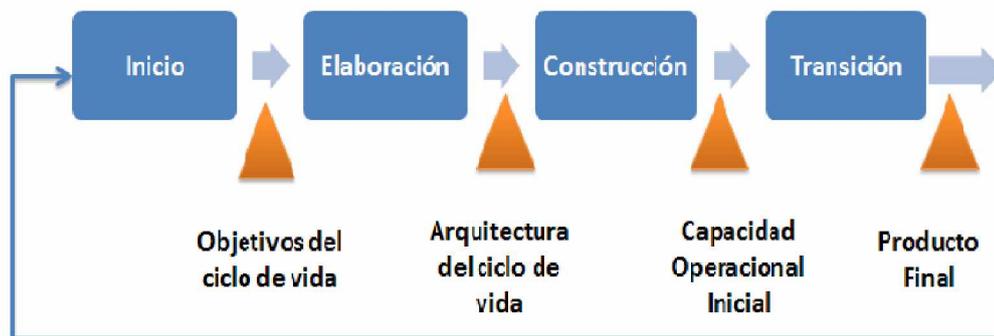
RUP se repite a lo largo de una serie de ciclos que constituyen la vida de un producto. Cada ciclo concluye con una generación del producto para los clientes. Cada ciclo consta de cuatro fases: Inicio, Elaboración, Construcción y Transición. Cada fase se subdivide a la vez en iteraciones, el número de iteraciones en cada fase es variable.



**Figura 2.9 Ciclos, relación, baseline**

**Fuente: (K96)**

Cada fase se concluye con un hito bien definido, un punto en el tiempo en el cual se deben tomar ciertas decisiones críticas y alcanzar las metas clave antes de pasar a la siguiente fase, ese hito principal de cada fase se compone de hitos menores que podrían ser los criterios aplicables a cada iteración. Los hitos para cada una de las fases son: Inicio *Objetivos del ciclo de Vida*, Elaboración *Arquitectura del ciclo de vida*, Construcción *Eficiencia Operacional inicial*, Transición *Relación con el Producto*. Las fases y sus respectivos hitos se ilustran en la Figura 2.10



**Figura 2.10 Fases e hitos del Proceso Unificado (RUP)**

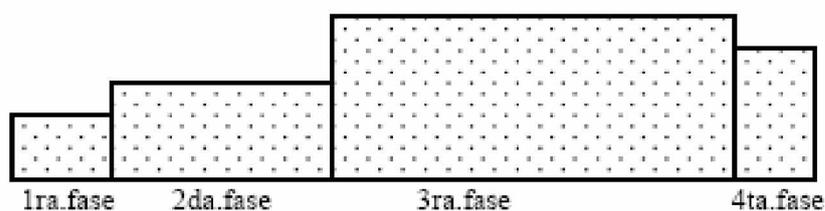
**Fuente: (K96)**

La duración y esfuerzo dedicado en cada fase es variable dependiendo de las características del proyecto. Sin embargo, la Tabla 2.1 ilustra porcentajes frecuentes al respecto. Consecuente con el esfuerzo señalado, la Figura 2.12 ilustra una distribución típica de recursos humanos necesarios a lo largo del proyecto.

	Inicio	Elaboración	Construcción	Transición
Esfuerzo	5 %	20 %	65 %	10%
Tiempo Dedicado	10 %	30 %	50 %	10%

**Tabla 2.1 Duración y esfuerzo en cada fase a lo largo del proyecto.**

**Fuente: (K96)**



**Figura 2.11 Expresión gráfica del tiempo y esfuerzo dedicados a cada fase de RUP**

**Fuente: (K96)**

### a) Fase de Inicio

Durante la fase de inicio se define el modelo del negocio y el alcance del proyecto. Se identifican todos los actores y Casos de Uso, se desarrolla, un plan de negocio para determinar que recursos deben ser asignados al proyecto.

Para la planificación de la fase de inicio se debe empezar a reunir información, cuando se disponga de suficiente información para presentar el plan detallado de la primera iteración. Los objetivos de esta fase nos permiten elaborar un documento apropiado de captura de requisitos.

MODELO RUP	ACTIVIDADES AGREGADAS A RUP
<ul style="list-style-type: none"> <li>Y Modelo inicial de Casos de Uso (10-20% completado).</li> <li>Y Plan del proyecto, mostrando fases e iteraciones.</li> </ul>	<ul style="list-style-type: none"> <li>Y Un análisis de las necesidades del entorno.</li> <li>Y Una lista de requerimientos pedagógicos relacionados con el contenido y la población a la que va dirigida el programa.</li> <li>Y Revisión de los objetivos y contenidos del material del programa.</li> <li>Y Establecer los límites de las áreas que se van a desarrollar.</li> </ul>

**Tabla 2.2 Fase Inicio**  
**Fuente: Elaboración Propia**

### b) Fase de Elaboración

El propósito de la fase de elaboración es analizar el dominio del problema, establecer los cimientos de la arquitectura, desarrollar el plan del proyecto y eliminar los mayores riesgos. Esta fase ayuda a planificar con gran precisión la fase de construcción.

Recibimos de la fase de inicio un plan para la fase de elaboración, un modelo de casos de uso parcialmente completo y una descripción de la arquitectura candidata.

Hacia el final de la fase de elaboración, se empieza a planificar en forma mas detallada la primera iteración de la fase de construcción, y esbozar en términos mas generales las iteraciones restantes.

En esta fase se construye un prototipo de la arquitectura, que debe evolucionar en iteraciones sucesivas hasta convertirse en el sistema final. Este prototipo debe contener los Casos de Uso críticos identificados en la fase de inicio. También debe demostrarse que se han evitado los riesgos más graves.

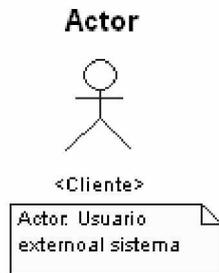
OBJETIVOS	RESULTADOS
<ul style="list-style-type: none"> <li>Y Definir, validar y cimentar la arquitectura.</li> <li>Y Crear un plan fiable para la fase de construcción.</li> </ul>	<ul style="list-style-type: none"> <li>Y Un modelo de Casos de Uso completa al menos hasta el 80%: todos los casos y actores identificados, la mayoría de los casos desarrollados.</li> <li>Y Requisitos adicionales que capturan los requisitos no funcionales y cualquier requisito no asociado con un Caso de Uso específico.</li> <li>Y Descripción de la arquitectura software.</li> <li>Y Plan de desarrollo para el proyecto.</li> <li>Y Un caso de desarrollo actualizado que especifica el proceso a seguir.</li> </ul>

**Tabla 2.3 Fase de Elaboración**  
**Fuente: Elaboración Propia**

En esta fase se debe tratar de abarcar todo el proyecto con la profundidad mínima. Sólo se profundiza en los puntos críticos de la arquitectura o riesgos importantes. En la fase de elaboración se actualizan todos los productos de la fase de inicio.

Y **Diagramas de los casos de uso.** Una vez identificado los casos de uso en la fase de inicio, se podrá realizar la representación grafica de los casos de uso. Los diagramas casos de uso muestran la relación que existe entre los actores y como opera con el sistema [WEB07].

## Y Elementos



Un **Actor** es un rol que un usuario juega con respecto al sistema. Es importante destacar el uso de la palabra **rol**, pues con esto se especifica que un Actor no necesariamente representa a una persona en particular, sino más bien la labor que realiza frente al sistema. Como ejemplo a la definición anterior, tenemos el caso de un sistema de ventas en que el rol de Vendedor con respecto al sistema puede ser realizado por un Vendedor o bien por el Jefe de Local [WEB07].

## Caso de Uso



Es una operación/tarea específica que se realiza tras una orden de algún agente externo, sea desde una petición de un actor o bien desde la invocación desde otro caso de uso.

### a. Relaciones:

∅ **Asociación** —————>

Es el tipo de relación más básica que indica la invocación desde un actor o caso de uso a otra operación (caso de uso). Dicha relación se denota con una flecha simple.

∅ **Dependencia o Instanciación** - - - - ->

Es una forma muy particular de relación entre clases, en la cual una clase depende de otra, es decir, se instancia (se crea). Dicha relación se denota con una flecha punteada.

### ∅ Generalización

Este tipo de relación es uno de los más utilizados, cumple una doble función dependiendo de su estereotipo, que puede ser de **Uso** (<<uses>>) o de **Herencia** (<<extends>>).

Este tipo de relación esta orientado exclusivamente para casos de uso (y no para actores).

**extends:** Se recomienda utilizar cuando un caso de uso es similar a otro (características).

**uses:** Se recomienda utilizar cuando se tiene un conjunto de características que son similares en más de un caso de uso y no se desea mantener copiada la descripción de la característica. De lo anterior cabe mencionar que tiene el mismo paradigma en diseño y modelamiento de clases, en donde esta la duda clásica de **usar** o **heredar**.

Y **Descripción de los casos de uso.** Este formato muestra la descripción narrativa de los procesos del dominio y que explica de una forma mas detallada la secuencia de pasos del actor, para ayudar a comprender los Casos de uso. También hace referencia a los requerimientos, con los cuales tiene relación. Para esta descripción se utilizará la siguiente plantilla (Tabla 2.3) [JBR00].

<b>Casos de uso:</b>	Nombre del caso de uso
<b>Actores:</b>	Que actores interactúan en el caso de uso
<b>Propósito:</b>	Objetivo del caso de uso
<b>Resumen:</b>	Resumen de los principales pasos del caso de uso
<b>Tipo:</b>	Que importancia es para el desenvolvimiento del sistema(prioridad)
<b>Referencia Cruzada:</b>	Que requerimientos y casos de uso mas utilizan este caso de uso
<b>Salida:</b>	No salidas de la interfaz del usuario; por ejemplo, mensajes o registros que se envían afuera del sistema
<b>Curso Normal de los Eventos</b>	
<b>Acción de los Actores:</b>	<b>Respuesta del Sistema</b>
Enumerado Causa	Enumerado Efecto

**Tabla 2.4 Plantilla de un caso de uso expandido.**

Y **Diagrama de clases.** Son de estructura estática al igual que los diagramas conceptuales representando un diseño estructural del sistema. Un diagrama de clases sirve para visualizar las relaciones que existe entre clases que representan entidades que involucran al sistema [WEB07].

Los diagramas de clases, también muestran un conjunto de clases, con sus atributos, operaciones, interfaces y sus relaciones.

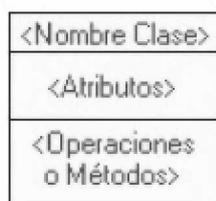
**a. Elementos**

• **Clase**

Es la unidad básica que encapsula toda la información de un Objeto (un objeto es una instancia de una clase). A través de ella podemos modelar el entorno en estudio (una Casa, un Auto, una Cuenta Corriente, etc.) [WEB07].

En UML, una clase es representada por un rectángulo que posee tres divisiones:

En donde:



Ø **Superior:** Contiene el nombre de la Clase.

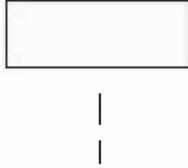
Ø **Intermedio:** Contiene los atributos (o variables de instancia) que caracterizan a la Clase (pueden ser private, protected o public).

Ø **Inferior:** Contiene los métodos u operaciones, los cuales son la forma como interactúa el objeto con su entorno (dependiendo de la visibilidad: private, protected o public).

Y **Diagrama de secuencia.** El Diagrama de Secuencia es uno de los diagramas más efectivos para modelar interacción entre objetos en un sistema. Un diagrama de secuencia se modela para cada caso de uso. Mientras que el diagrama de caso de uso permite el modelado de una vista 'negocio' del escenario, el diagrama de secuencia contiene detalles de implementación del escenario, incluyendo los objetos y clases que se usan para implementar el escenario, y mensajes pasados entre los objetos [WEB07].

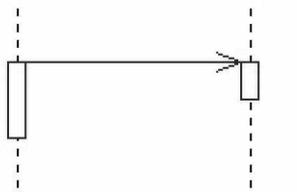
**a. Elementos**

Ø **Objeto/Actor:**



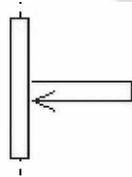
El rectángulo representa una instancia de un Objeto en particular, y la línea punteada representa las llamadas a métodos del objeto.

Ø **Mensaje a Otro Objeto:**



Se representa por una flecha entre un objeto y otro, representa la llamada de un método (operación) de un objeto en particular.

Ø **Mensaje al Mismo Objeto:**



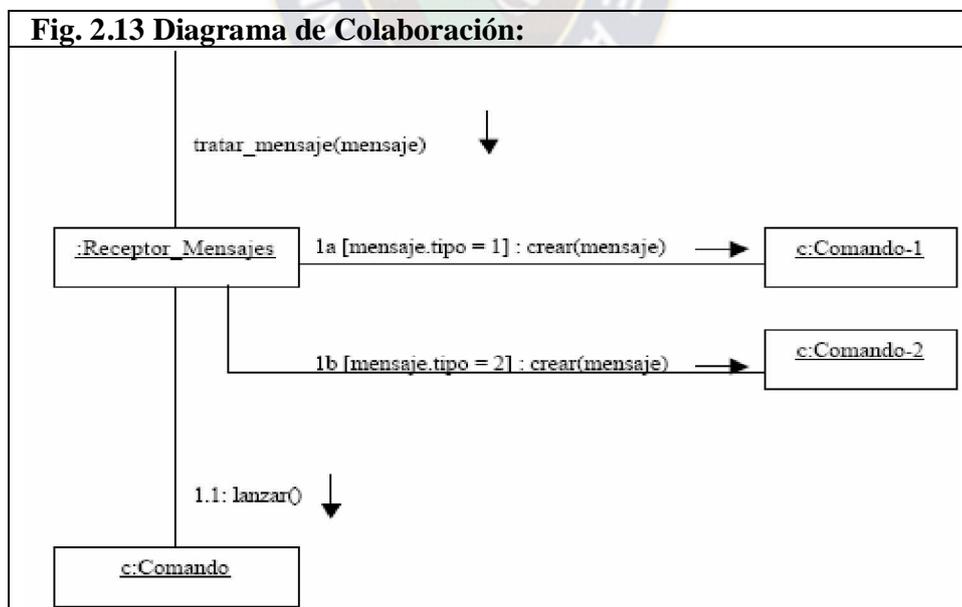
No solo llamadas a métodos de objetos externos pueden realizarse, también es posible visualizar llamadas a métodos desde el mismo objeto en estudio.

Ø **Contratos.** Los contratos contribuyen a definir el comportamiento del sistema; describen el efecto que sobre el tienen las operaciones. El lenguaje UML ofrece la mejor forma de representar estos contratos es mediante la siguiente plantilla

<b>Nombre:</b>	Nombre de la operación y parámetros
<b>Responsabilidades:</b>	Descripción informal de las responsabilidades de que debe cumplir la operación
<b>Tipo:</b>	Nombre del tipo (concepto, clase de software, interfaz)
<b>Referencias cruzadas:</b>	Número de referencias de las funciones del sistema, casos de uso, etc.
<b>Notas:</b>	Notas de diseño, algoritmos e información.
<b>Excepciones:</b>	Casos excepcionales
<b>Salida:</b>	No salidas de la interfaz del usuario; por ejemplo, mensajes o registros que se envían afuera del sistema
<b>Precondiciones:</b>	
<b>Poscondiciones:</b>	

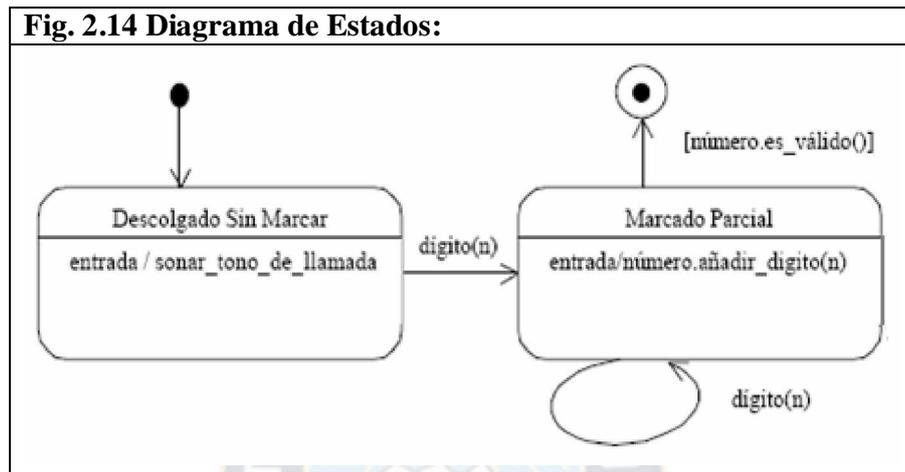
**Tabla 2.5 Plantilla de un Contrato**

Y **Diagrama de colaboración.** El Diagrama de Colaboración muestra una interacción organizada basándose en los objetos que toman parte en la interacción y los enlaces entre objetos en el sistema (en cuanto a la interacción se refiere).



**Fig. 2.12 Diagrama de Colaboración**  
Fuente: [JBR00].

Y **Diagrama de estados.** Un Diagrama de Estados muestra la secuencia de estados por los que pasa un caso de uso o un objeto a lo largo de su vida, indicando qué eventos hacen que se pase de un estado a otro y cuáles son las respuestas y acciones que genera.



**Fig. 2.13 Diagrama de Estados**

**Fuente: [JBR00].**

### c) Fase de Construcción

La finalidad principal de esta fase es alcanzar la capacidad operacional del producto de forma incremental a través de las sucesivas iteraciones. Durante esta fase todos los componentes, características y requisitos deben ser implementados, integrados y probados en su totalidad, obteniendo una versión aceptable del producto.

En esta fase se detalla los casos de uso y escenarios restantes, modifica la descripción de la arquitectura si es necesario.

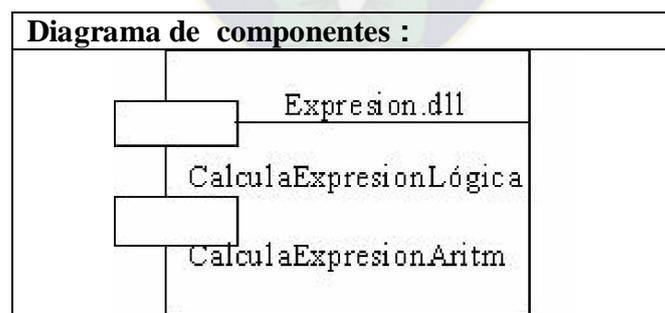
Los criterios de evaluación se basan en: material de usuario, material de cursos.

OBJETIVOS	RESULTADOS
<ul style="list-style-type: none"> <li>Y Minimizar los costes de desarrollo mediante la optimización de recursos y evitando el tener que rehacer un trabajo o incluso desecharlo.</li> <li>Y Conseguir una calidad adecuada tan rápido como sea práctico.</li> <li>Y Conseguir versiones funcionales (versiones de prueba) tan rápido como sea práctico.</li> </ul>	<ul style="list-style-type: none"> <li>Y Modelos Completos (Casos de Uso, Análisis, Diseño, Despliegue e Implementación)</li> <li>Y Arquitectura íntegra (mantenida y mínimamente actualizada)</li> <li>Y Riesgos Presentados Mitigados</li> <li>Y Plan del Proyecto para la fase de Transición.</li> <li>Y Manual Inicial de Usuario (con suficiente detalle)</li> <li>Y Prototipo Operacional</li> </ul>

**Tabla 2.6 Fase de Construcción**

**Fuente: elaboración propia**

Y **Diagrama de componentes.** Un diagrama de componentes muestra la organización y las dependencias entre un conjunto de componentes. Un diagrama de componentes consta de componentes, como archivos de código fuente, código binario, ejecutables o bibliotecas de vínculos dinámicos (DLL), conectados mediante dependencias.

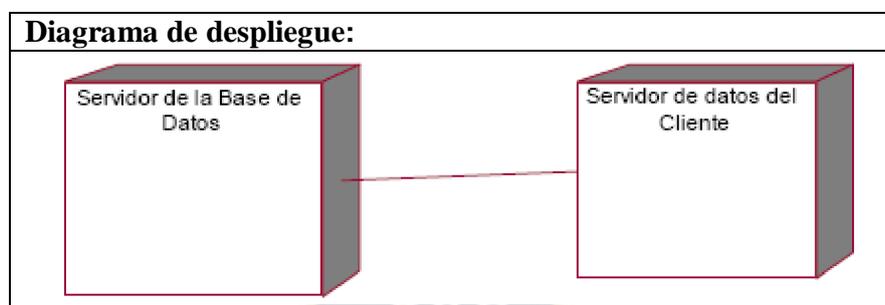


**Fig. 2.14 Diagrama de componentes**

**Fuente: [JBR00].**

Y **Diagrama de despliegue.** Los diagramas de despliegue son útiles para facilitar la comunicación entre los ingenieros de hardware y los de software. Los diagramas de despliegue muestran la estructura del sistema en tiempo de ejecución.

Los diagramas de despliegue constan de nodos, componentes y las relaciones entre ellos como se puede ver en la figura 2.15. Dependencia conectada a una interfaz.



**Fig. 2.15 Diagrama de despliegue**  
Fuente: [JBR00].

#### d) Fase de Transición

La finalidad de la fase de transición es poner el producto en manos de los usuarios finales, para lo que se requiere desarrollar nuevas versiones actualizadas del producto, completar la documentación, entrenar al usuario en el manejo del producto, y en general tareas relacionadas con el ajuste, configuración, instalación y facilidad de uso del producto.

El proyecto recibirá información para: determinar si el sistema hace lo que demandan los usuarios y el negocio, descubrir los riesgos inesperados, a notar problemas no resueltos, encontrar fallas, eliminar ambigüedades.

OBJETIVOS	RESULTADOS
<ul style="list-style-type: none"> <li>Y satisfaga suficientemente al usuario.</li> <li>Conseguir que el usuario se valga por si mismo.</li> <li>Y Un producto final que cumpla los requisitos esperados.</li> </ul>	<ul style="list-style-type: none"> <li>Y Documentos Legales.</li> <li>Y Caso del Negocio Completo.</li> <li>Y Línea de Base del Producto completa y corregida que incluye todos los modelos del sistema.</li> <li>Y Descripción de la Arquitectura completa y corregida.</li> <li>Y Las iteraciones de esta fase irán dirigidas normalmente a conseguir una nueva versión.</li> </ul>

**Tabla 2.7 Proyectos de Grado que hacen uso del SIG**  
Fuente de elaboración propia

## **2.4 ARQUITECTURA CLIENTE/SERVIDOR.**

Una arquitectura cliente servidor consta de un cliente inteligente que puede solicitar servicios de un servidor en red. En el lado del cliente de esta arquitectura encontramos una aplicación frontal bastante sencilla ejecutándose en una computadora personal bastante sencilla. En el lado del servidor encontramos un motor de servidor de base de datos inteligente. El servidor está diseñado para aceptar consultas SQL desde la aplicación frontal, generalmente en forma de llamadas a procedimientos almacenados que devuelven conjunto de resultados claramente definidos y de ámbito limitado [PR07].

## **2.5 INGENIERÍA DE USABILIDAD**

Es una metodología que está cobrando importancia en el desarrollo de software, la usabilidad se presenta como un atributo del software. La usabilidad de un sistema está ligada principalmente a la interacción del mismo, al modo en que se realizan las operaciones con el sistema.

Es tan importante que el usuario encuentre la interfaz de usuario atractivo, de fácil manejo y correcta, ya que si es todo lo contrario el usuario puede frustrarse al no alcanzar sus objetivos.

El diseño de la interfaz de usuario comienza con la identificación de los requisitos del usuario empieza con la identificación de los requisitos de éste, la tarea y el ambiente. Una vez identificadas las tareas del usuario, se crean y analizan los escenarios de éste para definir un conjunto de objetos y acciones para la interfaz. Esto constituye la base para la creación de formatos de pantalla que representan el diseño gráfico y la ubicación de los iconos; la definición del texto descriptivo en pantalla; la especificación y asignación de nombres a las ventanas, además de la especificación de los elementos principales y secundarios de los menús. Se recurre a herramientas para crear prototipos y finalmente implementar el modelo de diseño; por último se evalúa la calidad del resultado [PR07].

### **2.5.1 Modelos del Análisis y diseño de la Interfaz**

Existen cuatro tipos de modelos distintos cuando se analiza y diseña una interfaz:

Y **Modelo del usuario.** Establece el perfil de los usuarios finales del sistema. Para construir una interfaz de usuario efectiva,” todo diseño debe empezar por la comprensión de quiénes son los usuarios de destino, incluidos sus perfiles de edad, sexo, habilidades físicas, educación y antecedentes culturales o étnicos, motivaciones, objetivos y personalidad”. Además, es posible distribuir a los usuarios en las siguientes categorías [PR07]:

- I. **Principiantes.** No tienen conocimientos de la sintaxis del sistema y cuentan con escasos conocimientos de la semántica de la aplicación o del uso de la computadora en general [PR07].
- II. **Usuarios esporádicos y con conocimientos.** Tienen conocimientos razonables de semántica, pero muestran una retención relativamente baja de la información sobre sintaxis necesaria para utilizar la interfaz [PR07].
- III. **Usuarios frecuentes y con conocimientos.** Cuentan con conocimientos de sintaxis y semántica suficiente para llegar al “síndrome del usuario avanzado” es decir individuos que buscan combinaciones y métodos abreviados para interactuar [PR07].

Y **Modelo del diseño.** Incorpora datos, arquitectura, interfaz y representaciones procedimentales del software [PR07].

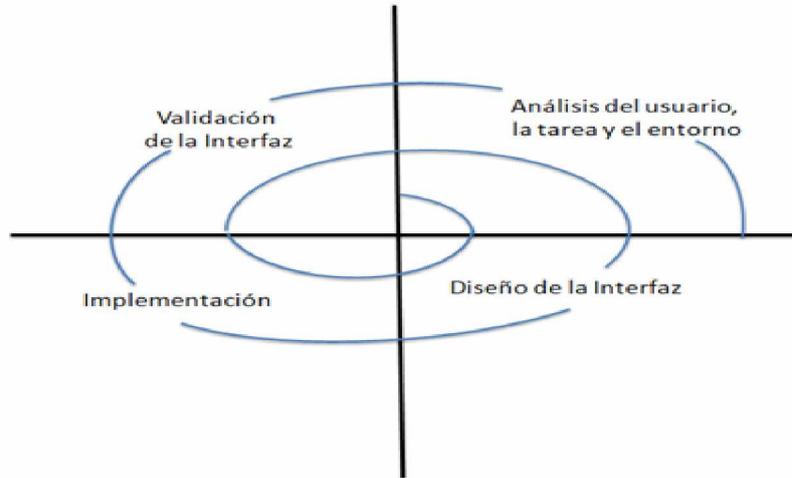
Y **Modelo mental del usuario o percepción del sistema.** Es la Imagen del sistema, que los usuarios finales llevan en la mente [PR07].

Una interfaz debe facilitar el proceso de crear un modelo mental. Un usuario que ha tenido la experiencia de manejar por lo menos una sola vez cualquier sistema.

Y **Modelo de la implementación.** Combina la manifestación externa del sistema (la apariencia de la interfaz) y toda la información de ayuda (libros, manuales, cinta de video, archivos de ayuda) que sirve para describir la sintaxis y semántica del sistema.

**2.5.2 El proceso de diseño de interfaz.** El proceso de análisis y diseño de las interfaces de usuario es iterativo y se representa con un modelo espiral ver fig. (a) 2.5.2, se puede

observar que el proceso de análisis y diseño de la interfaz de usuario abarca cuatro actividades distintas de marco de trabajo.



**Fig. 2.16 El proceso de diseño de interfaz**

**Fuente: [PR07].**

- Y **Análisis y modelado de usuarios, tareas y entornos.** Se fija la atención en el perfil de los usuarios que interactuarán con el sistema. Se registra el grado de habilidad, la comprensión del trabajo y la disposición en diferentes categorías. Describen y elaboran las tareas que el usuario realiza para alcanzar sus objetivos. Qué tipo de usuarios van a utilizar el programa, qué tareas van a realizar los usuarios y cómo las van a realizar, qué exigen los usuarios del programa, en qué entorno se desenvuelven los usuarios (físico, social, cultural).
- Y **Diseño de la interfaz.** El objetivo es definir un conjunto de objetos y acciones (y sus representaciones en pantalla) que permitan que el usuario realice todas las tareas definidas, de manera que cumplan todos los objetivos.
- Y **Construcción de la interfaz.** Se inicia al crear un prototipo previo, una primera versión que permita evaluar los escenarios de uso. A medida que continúa el proceso de diseño iterativo, se puede utilizar diferente tipo de herramientas de desarrollo de la interfaz de usuario.

Y **Validación de la interfaz.** Se deben realizar pruebas de usabilidad del producto, a ser posible con los propios usuarios finales del mismo, el cual se fija en:

- a. La capacidad de la interfaz para implementar correctamente todas las tareas del usuario, arreglar todas las variaciones de las tareas del usuario, cumpliendo con los requisitos del usuario.
- b. Facilidad en la utilización y el aprendizaje de la interfaz.
- c. La aceptación del usuario de que la interfaz es una herramienta útil para desempeñar su trabajo.

## 2.6 MÉTRICAS DE CALIDAD

Las métricas de calidad se fueron aplicando al desarrollo del software, para conocer mejor el diseño y la construcción del software. Y tener una mejor evaluación del producto de la ingeniería o de los sistemas que se construyen.

El estándar ISO 9126 se desarrolló para identificar los atributos de calidad de software. El estándar identifica atributos clave de calidad, también se hará uso del factor de calidad como es el de fiabilidad [PR07]:

1. Funcionalidad
2. Confiabilidad
3. Facilidad de uso
4. Facilidad de mantenimiento
5. Portabilidad

Del que solo se tomara en cuenta cuatro de los primeros atributos de calidad de este estándar.

**2.6.1 Funcionalidad.** Proveer funciones que cumplen las necesidades explícitas e implícitas cuando es utilizado en condiciones especificadas (adecuación, precisión, seguridad (datos), conformidad).

La funcionalidad, no se puede medir directamente, se debe derivar indirectamente mediante otras medidas directas como el punto función se calculan completando la siguiente tabla, determinando los cinco dominios de información: número de entradas de usuario, número de salidas de usuario, número de peticiones de usuario, número de archivos y número de interfaces externas.

Parámetros de medición	cuenta		Simple	Medio	Complejo		
Número de entradas de usuario	X		3	4	6	=	
Número de salidas de usuario	X		4	5	7	=	
Número de peticiones de usuario	X		3	4	6	=	
Numero de archivos	X		7	10	15	=	
Número de interfaces externas	X		5	7	10	=	

**Tabla 2.8 Calculo de Puntos Función**  
Fuente: [PR07].

$$PF = \text{Cuenta Total} * (0.65 + 0.01 * \sum(Fi))$$

Valor	Descripción
0	No influencia
1	Incidental
2	Moderado
3	Medio
4	Significativo
5	Esencial

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

**Tabla 2.9 Calculo de ajuste de complejidad**  
Fuente: [PR07].

**2.6.2 Confiabilidad.** Capacidad de mantener un nivel especificado de desempeño cuando utilizado en condiciones especificadas.

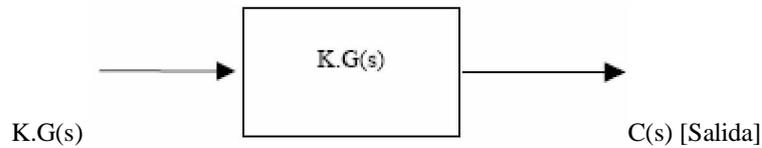
Para esto se utilizo funciones de transferencia, para representar las relaciones de causa y efecto en todo el sistema. Definimos entonces a la función transferencia de un sistema como la relación que existe entre la variable perturbada o salida, y la variable perturbadora o de entrada:

La confiabilidad de cada modulo del diagrama, esta especificado en la siguiente figura:

$\lambda$  : Probabilidad de que se presente una perturbación.

**P (T):** Probabilidad de fallo en caso de perturbación.

Que puede ser representada en diagramas en bloque por la figura (2.17)



**Fig. 2.17 Función de Transferencia**  
**Fuente: [PR07].**

**2.6.3 Facilidad de Uso.** Capacidad de ser entendido, comprendido, usado y resultar atractivo para el usuario, para utilizar el software de acuerdo con los siguientes subatributos: facilidad de comprensión, facilidad de aprendizaje y operatividad [PR07].

Medida	Calificación	Observación
Tiempo que los usuarios se toman para completar tareas específicas.		Calificación media obtenida
El número de tareas de diversos tipos que pueden ser completadas dentro de un tiempo límite dado.		Calificación media obtenida
La relación entre las interacciones exitosas y los errores.		Calificación media obtenida
El tiempo empleado en la recuperación de los errores.		Calificación media obtenida
El número de errores de usuario		Calificación media obtenida
El número de acciones erróneas inmediatamente posteriores.		Calificación media obtenida
La frecuencia del uso de manuales y/o o ayuda del sistema resolviendo el problema del usuario.		Calificación media obtenida
El número de ocasiones que el usuario se debía de la tarea real		Calificación media obtenida

**Tabla 2.10 Calculo de ajuste de complejidad**  
**Fuente: [PR07].**

**2.6.4 Facilidad de mantenimiento.** Ser modificado por correcciones, mejoras o adaptaciones a cambios en el entorno, requerimientos o especificaciones funcionales.

El estándar de calidad IEEE 982.1-1988 [IEE94] sugiere un índice de madurez del software (IMS) que proporciona una indicación de la estabilidad de un producto de software (basada

en los cambios que ocurren con cada versión del producto). Se determina la siguiente información [P07]:

$M_t$  = el número de módulos en la versión actual

$F_c$  = el número de módulos cambiados en la versión actual

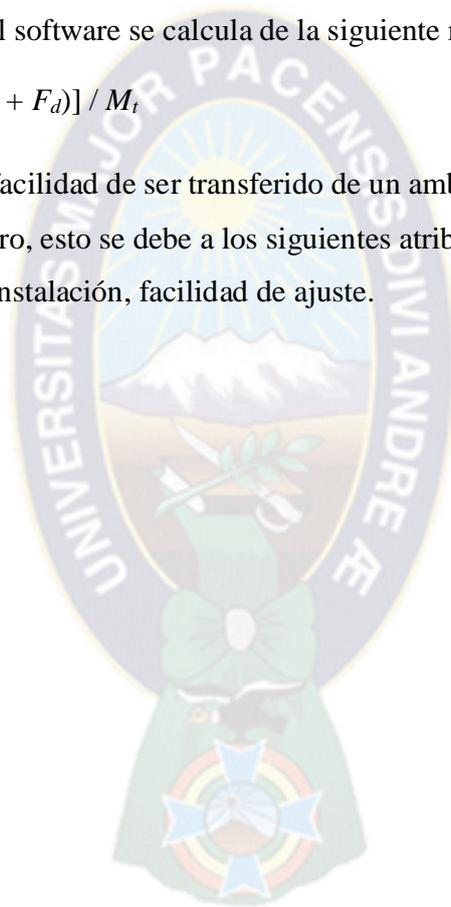
$F_a$  = el número de módulos añadidos a la versión actual

$F_d$  = el número de módulos de la versión anterior que se eliminaron en la actual

El índice de madurez del software se calcula de la siguiente manera:

$$\text{IMS} = [M_t - (F_a + F_c + F_d)] / M_t$$

**2.6.5 Portabilidad.** La facilidad de ser transferido de un ambiente (organizacional, hardware, software) a otro, esto se debe a los siguientes atributos: facilidad de adaptación al cambio, facilidad de instalación, facilidad de ajuste.



## **CAPÍTULO III**

### **MARCO APLICATIVO**

#### **3.1 INTRODUCCIÓN**

Hoy en la actualidad, se sigue utilizando métodos que se utilizaron hace ya más de 30 años. Si bien el avance que se tuvo para la representación de los métodos para el desarrollo de software que es lo que se necesita actualmente. El Proceso Unificado RUP es el resultado de muchos años de experiencia, y es una solución a diferentes problemas de proceso desarrollo de software, que a la vez hace uso del Lenguaje Unificado de Modelado (Unified Modeling Language, UML).

El proceso de software tiene tres aspectos fundamentales: dirigidos por casos de uso, esta centrado en la arquitectura, y es iterativo incremental. El Proceso Unificado es más que un simple proceso; es un marco de trabajo genérico que se adapta a una gran variedad de sistemas de software, diferentes tipos de aplicación, diferentes tipos de organizaciones y distintos tipos de tamaño de software.

#### **3.2 FASE INICIAL**

En esta fase se realizó el análisis de todos los problemas que existían en Zoonosis. Posteriormente se hizo un estudio de los requerimientos de los usuarios mediante entrevistas con el responsable de esta Unidad.

##### **3.2.1 Requerimientos del Usuario**

Para este cometido se hizo una recopilación de información y un estudio de los requerimientos del usuario con entrevistas con el encargado de Zoonosis, se pudo percibir lo siguiente.

- Y Registro de denuncias de personas que fueron por agredidos por canes
- Y Registrar al denunciante o afectado

- Y Realizar el registro de los animales domésticos.
- Y Realizar el registro de los propietarios de las mascotas
- Y Registrar los síntomas de la mascota.
- Y Registrar los Resultados de Inlasa.
- Y Registrar las capturas.
- Y Registrar los antecedentes.
- Y Consultas de las zonas, donde se presume que existe rabia.
- Y Consultas de los denunciantes que fueron agredidos.
- Y Consulta de los propietarios y sus mascotas.
- Y Realizar el reporte de Matricula.
- Y Realizar reportes de las zonas donde se presume que hay índices de rabia.
- Y Realizar el reporte de los casos de rabia que se detectaron.
- Y Consolidar la información almacenada en la BD con la documentación física.

### **3.3.2 Requerimientos Tecnológicos**

#### **I. Requerimientos de Software**

Para la realización del Sistema SIGZ se utilizará el lenguaje de programación Visual Basic.NET, Mapobject2.0 LT, y la herramienta ArcView para representar los datos georreferenciados de la ciudad de El Alto, esto porque la institución cuenta con las licencias de este lenguaje de programación.

No es complejo en su sintaxis, en otras palabras es de uso fácil. Su interfaz es muy amigable y tiene varios componentes que pueden facilitar el entendimiento y el uso de sus ventanas.

Es conocido por la institución, y existe personal capacitado en el uso y el manejo de este lenguaje de programación.

También se hará uso del motor de base de datos SQL Server 2000 del cual la institución tiene su respectiva licencia de uso. Esto porque es aconsejable utilizarlo, ya que es seguro en su manipulación, posee otras opciones y es amigable.

## **Plataforma**

La plataforma de software es Windows Server 2003.

## **II. Requerimientos de Hardware**

El hardware que se requiere para presente proyecto es:

Oficina Central de Zoonosis contara con un equipo.

Y Servidor Central

- i. Computadora Pentium IV
- ii. Procesador de 2.4 Ghz.
- iii. Memoria RAM de 256 Megabytes como mínimo.
- iv. Disco duro de almacenamiento 40 Gigabytes.

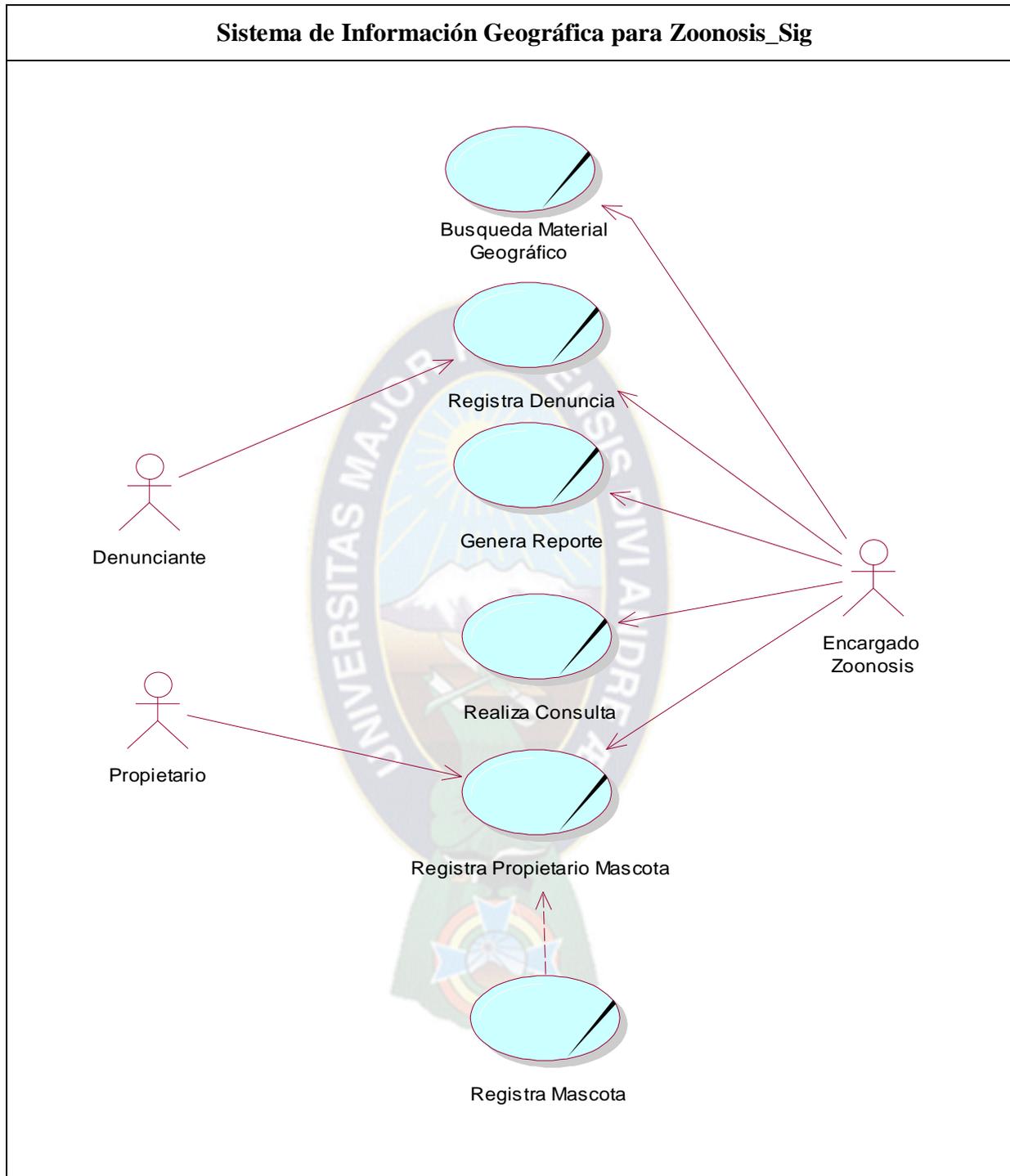
## **3.3 FASE DE ELABORACIÓN**

En esta fase se define el modelo completo del negocio (o de dominio), que describe el contexto del sistema.

### **3.3.1 Modelo del Negocio**

Establece una abstracción de la organización, y será representado por el diagrama de caso de uso general (Figura 3.1). Donde se encuentran los actores que intervienen en este caso el encargado(los técnicos de Zoonosis), y (denunciante, propietario).

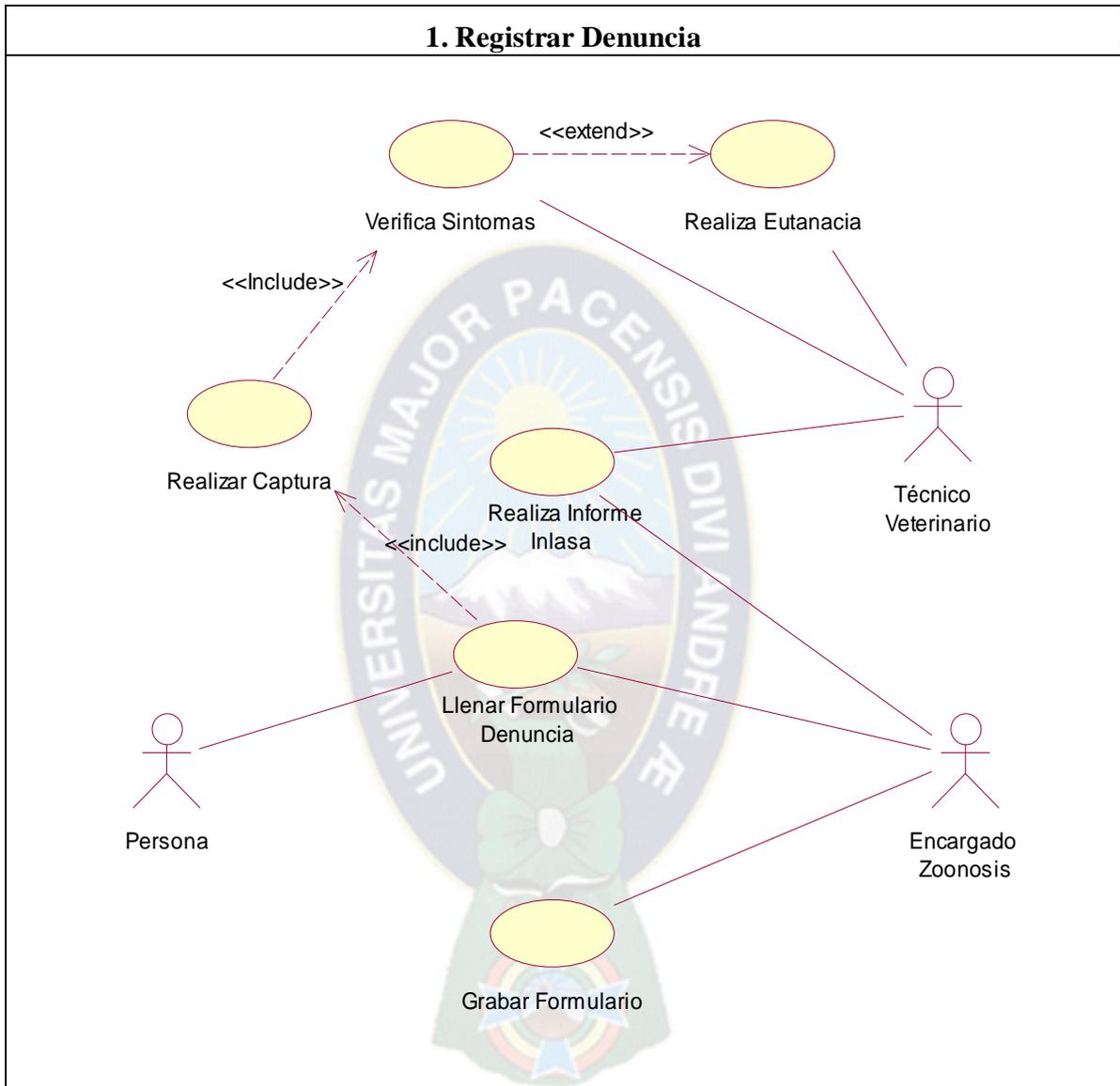
## Sistema de Información Geográfica para Zoonosis\_Sig



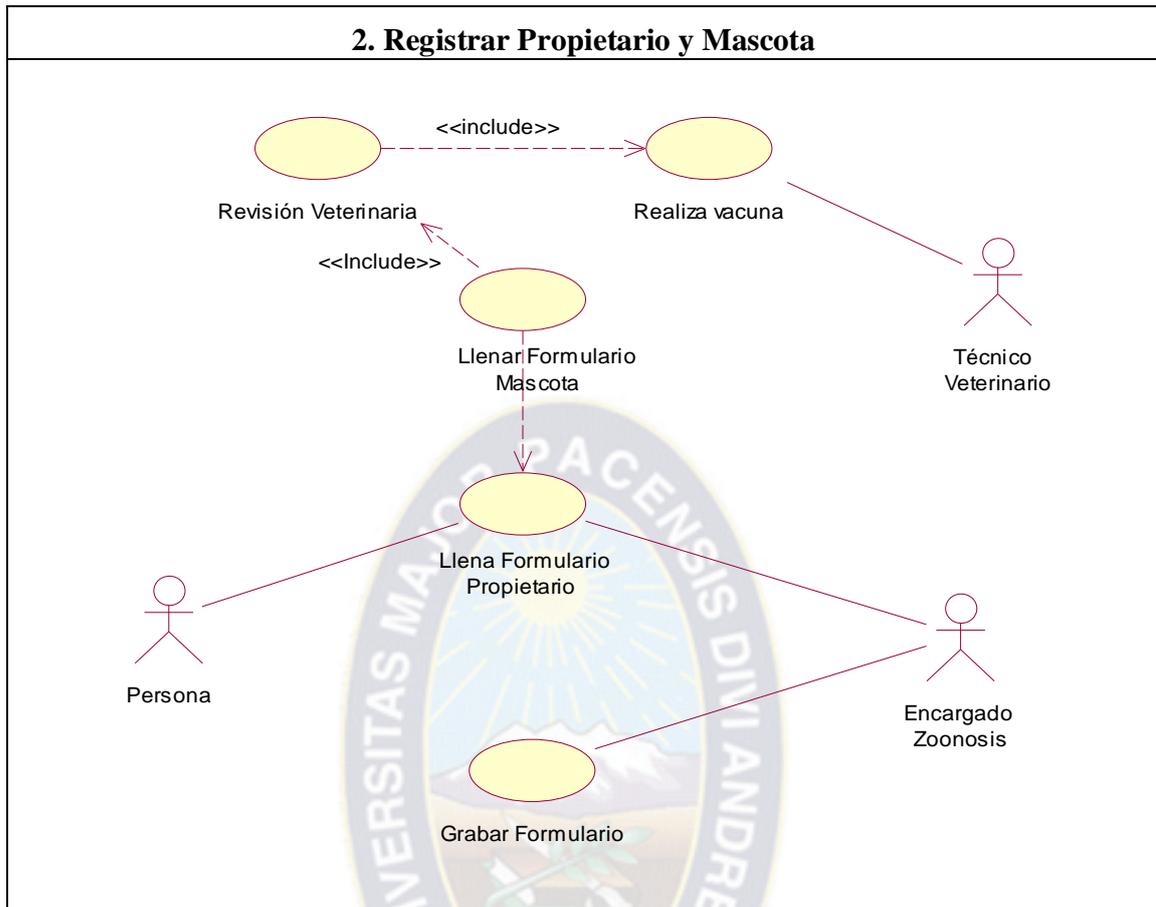
**Figura 3.1 Modelo del Negocio SIGZ**

**Fuente: Elaboración Propia**

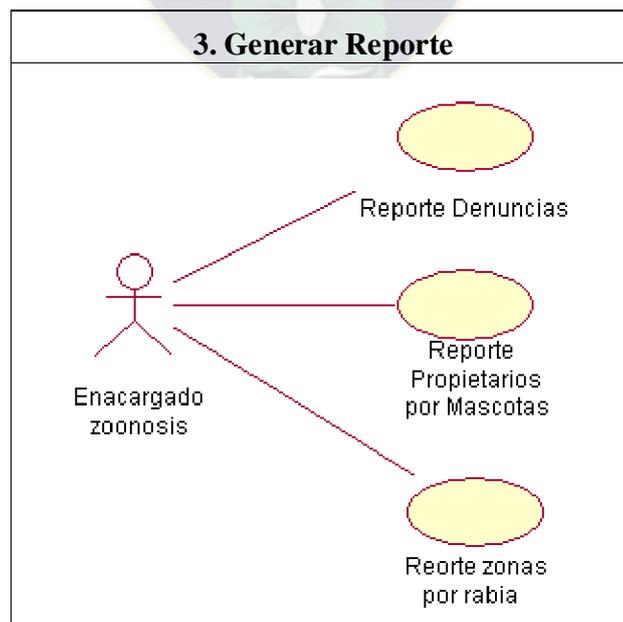
## Casos de Uso Esencial



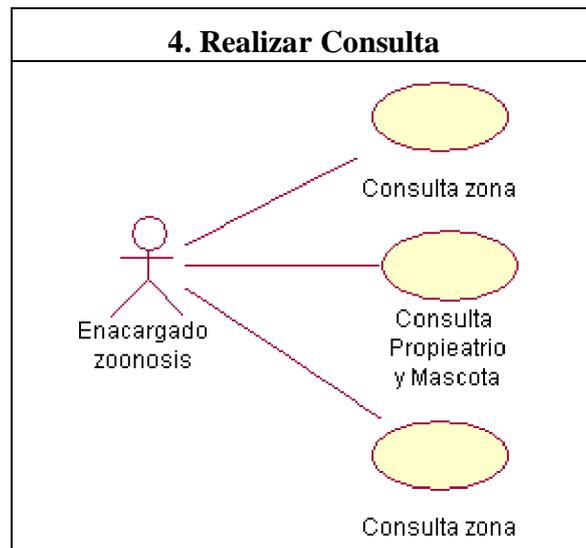
**Figura 3.2** Caso de uso: Registrar Denuncia



**Figura 3.3 Caso de uso: Registrar Mascota**



**Figura 3.4 Caso de uso: Generar Reporte**



**Figura 3.5 Caso de uso: Realizar Consulta**

- a. **Casos de Uso Específico** Los casos de Uso Específico serán expresados narrativamente en el (Anexo C).

### 3.3.2 Diagramas de Secuencia

Con los diagramas de secuencia se explicará los pasos a seguir en forma secuencial de los diferentes procesos que existan en el sistema actual y que serán desarrollados de acuerdo a los casos de uso esenciales, comenzando de la acción del actor y la respuesta del sistema mediante la acción de las clases existentes.

- a. **Registro de Denuncia**

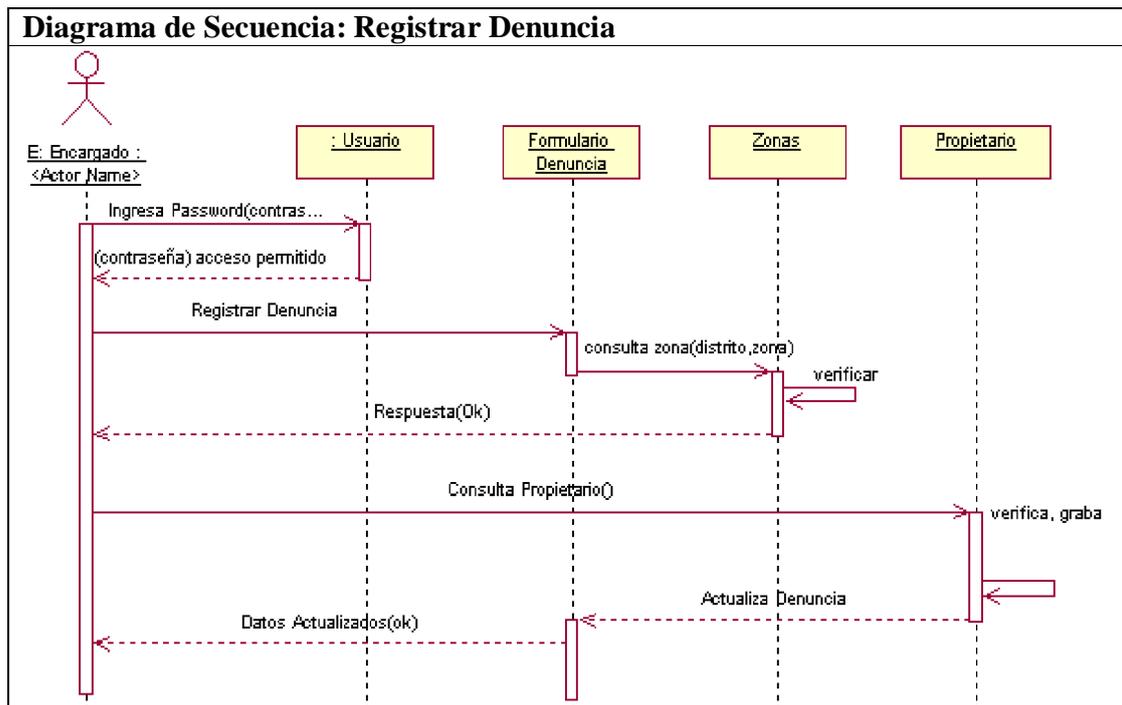


Figura 3.6 Diagrama de Secuencia: Registrar Denuncia

#### b. Registro de Mascota

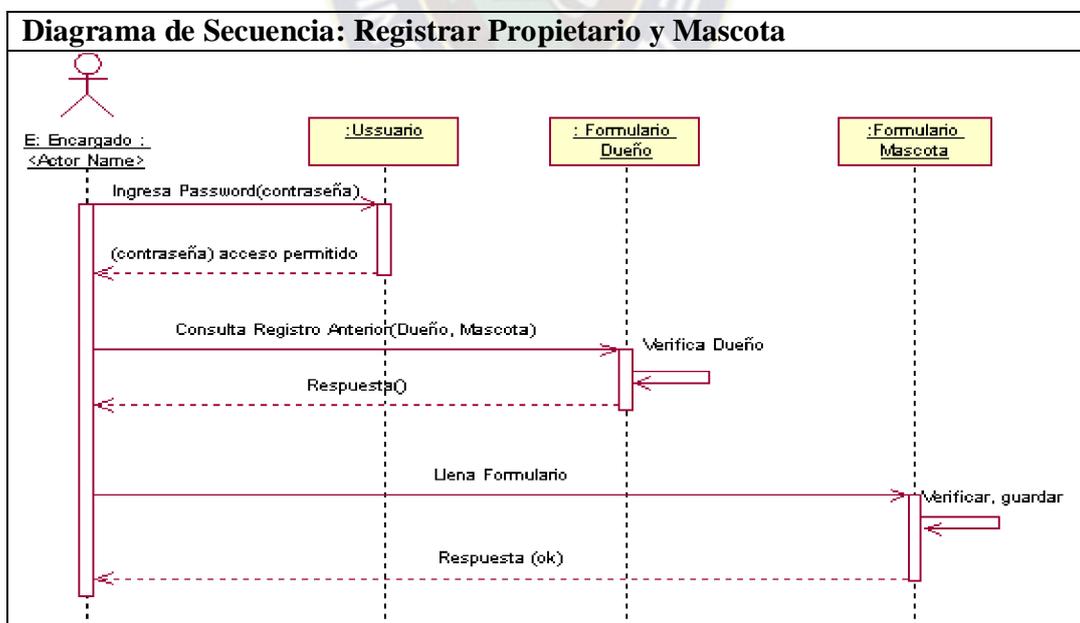
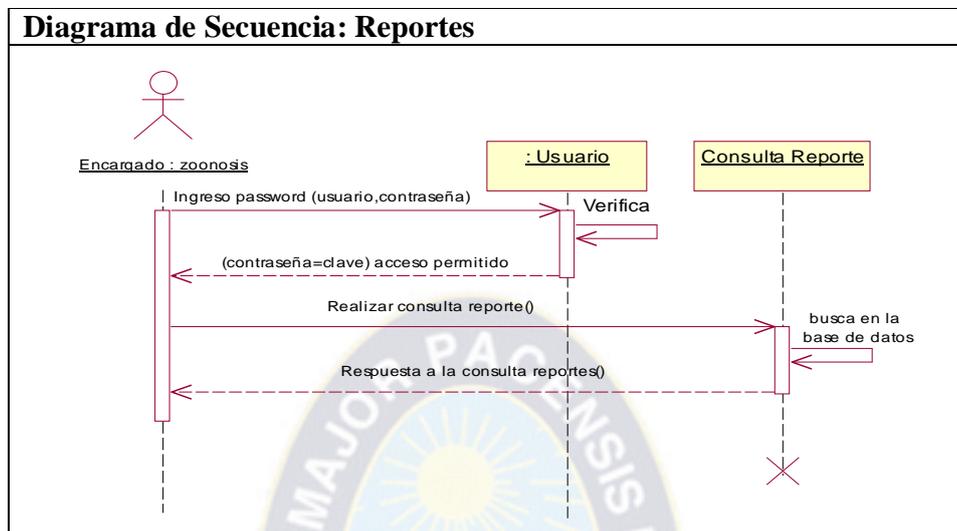


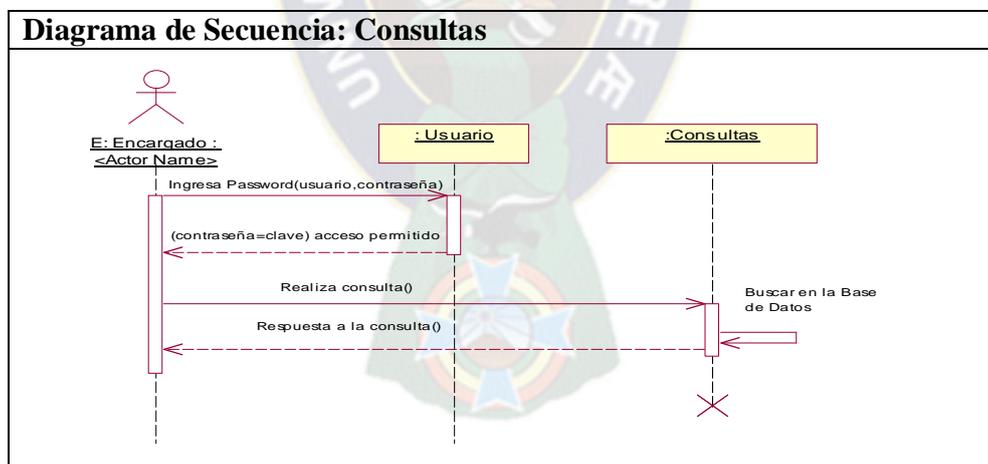
Figura 3.7 Diagrama de Secuencia: Registrar Propietario y Mascota

c. **Realizar Consulta.** Los usuarios serán (Encargado de Zoonosis).



**Figura 3.8 Diagrama de Secuencia: Reportes**

d. **Realizar Consulta.** Los usuarios serán (Encargado de Zoonosis).



**Figura 3.9 Diagrama de Secuencia: Consultas**

### 3.3.3 Contratos

Los contratos que se detectaron se encuentran explicados en el (Anexo D).

Para entender de mejor manera a que caso de uso pertenecen estos contratos, no se los mencionando; sino se esta colocando solo la numeración que corresponde un determinado caso de uso el cual esta incluyendo dicho contrato.

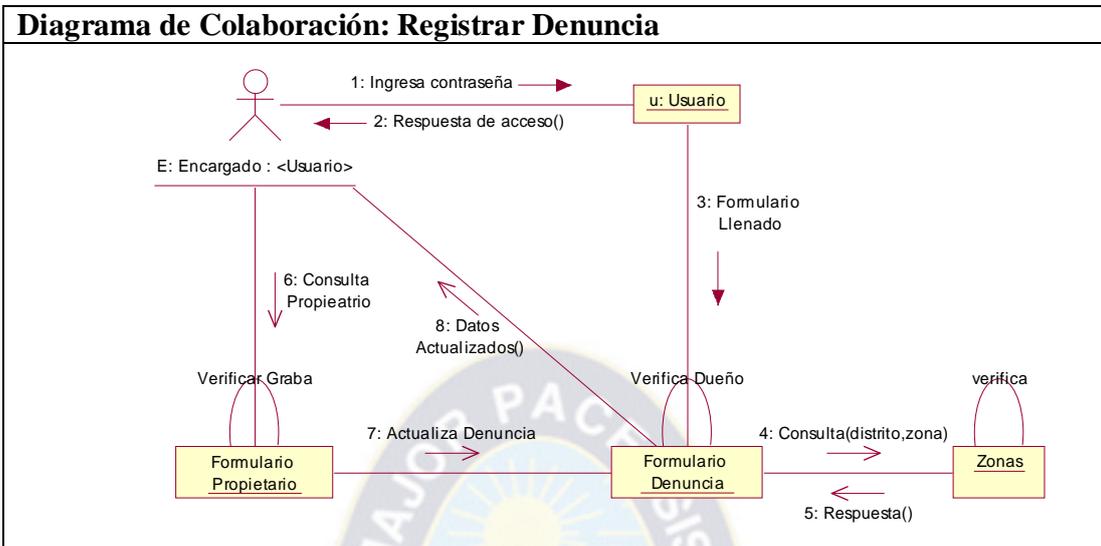
<b>Nombre:</b>	Llenar formulario de denuncia
<b>Responsabilidades:</b>	Tendrá la posibilidad de llenar los datos de los medicamentos, como fecha, lugar, datos personales, observaciones.
<b>Tipo:</b>	Interfaz de Llenar formulario
<b>Referencias cruzadas:</b>	Caso de uso 1
<b>Notas:</b>	Utilizar acceso rápido a la base de datos.
<b>Excepciones:</b>	Que no se haya Llenado de forma correcta el formulario.
<b>Salida:</b>	No salidas de la interfaz del usuario; por ejemplo, mensajes o registros que se envían afuera del sistema
<b>Precondiciones:</b>	Requerimiento de llenado de formulario.
<b>Poscondiciones:</b>	Registro de datos llenados en la tabla de formulario.

**Tabla 3.1 Contrato: Formulario de denuncia**

### 3.3.4 Diagramas de colaboración

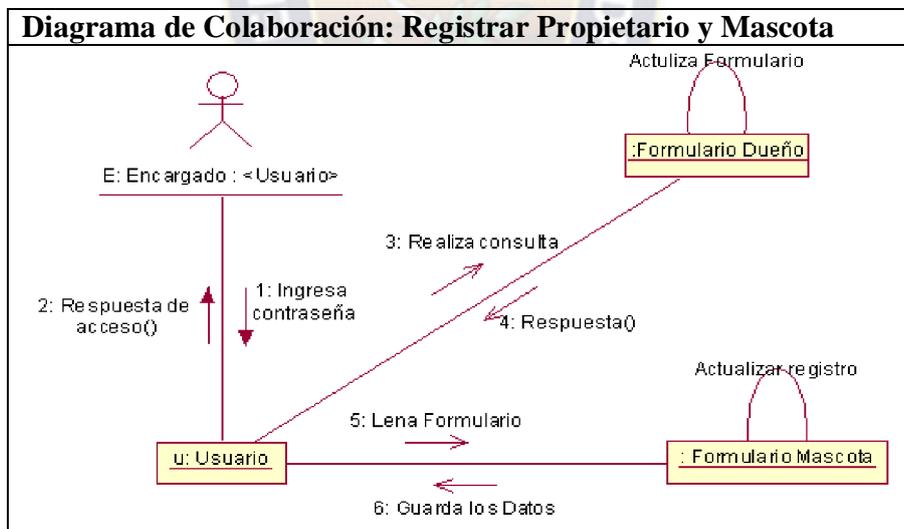
Estos diagramas serán elaborados de acuerdo a los contratos, diagramas de secuencia y a los mensajes encontrados entre las clases del sistema y se dan de acuerdo a los casos de uso generales que existen.

**a. Registro de Denuncia**



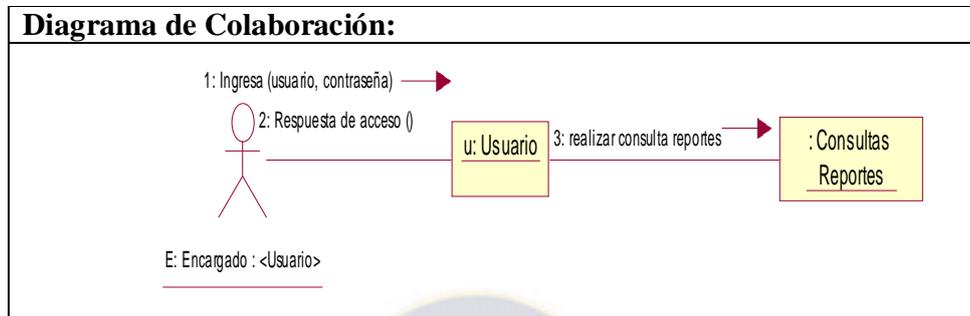
**Figura 3.10 Diagrama de Colaboración: Registrar Denuncia**

**b. Registro de Propietario y Mascota**



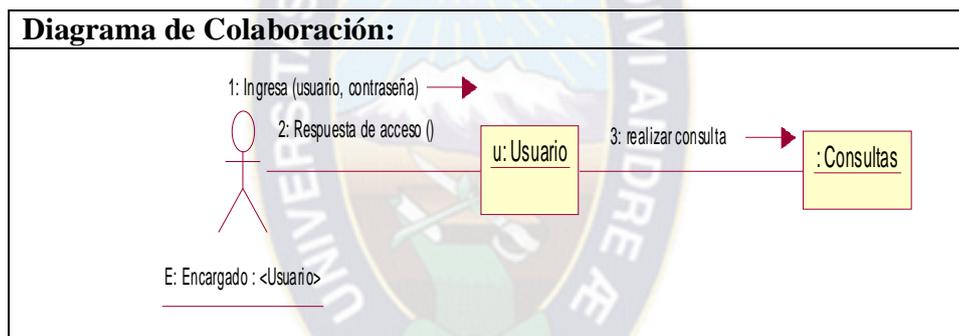
**Figura 3.11 Diagrama de Colaboración: Registrar Propietario y Mascota**

**c. Realizar consulta reporte**



**Figura 3.12 Diagrama de Colaboración: Realizar consulta reporte**

**d. Realizar consulta**



**Figura 3.13 Diagrama de Colaboración: Realizar consulta**

**3.3.4 Diagramas de estado**

Estos diagramas serán elaborados de acuerdo a los casos de uso esenciales y a los diagramas de colaboración, para saber las transiciones y estados que tiene cada caso de uso.

En estos diagramas también es importante mencionar el actor que interactúan en cada diagrama de estado para dar mayor coherencia a las transiciones y estados que tienen estos casos de uso.

a. Registrar Denuncia

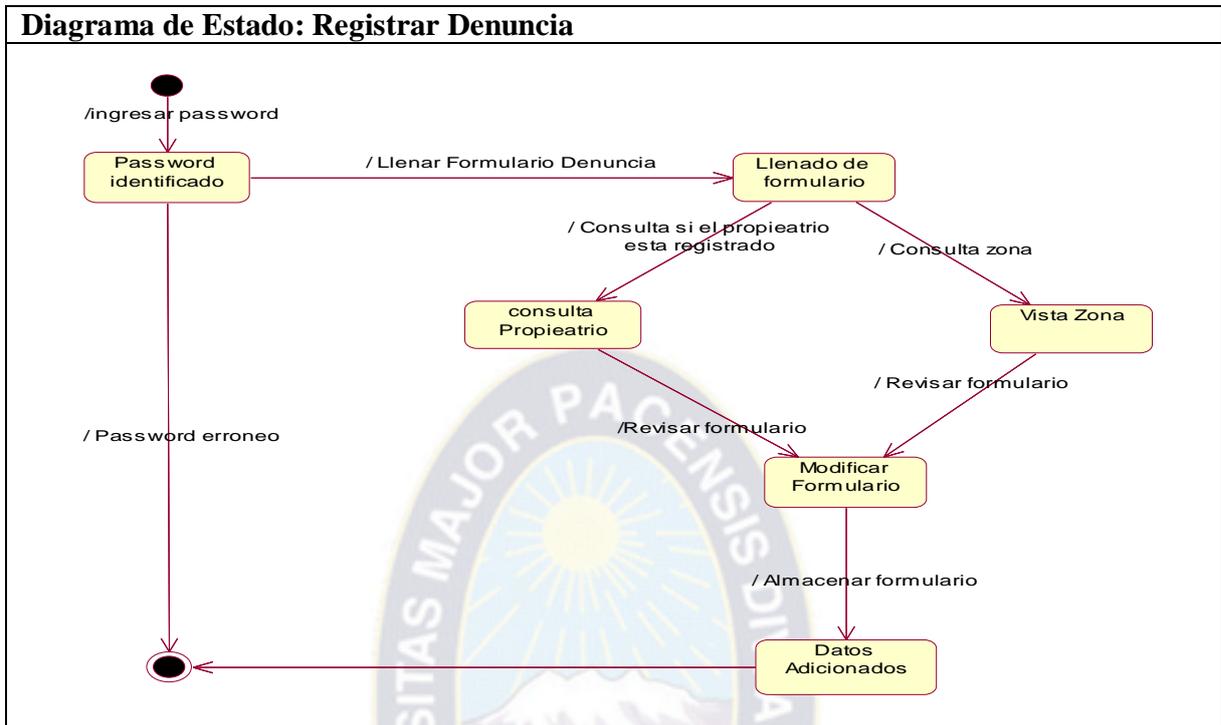
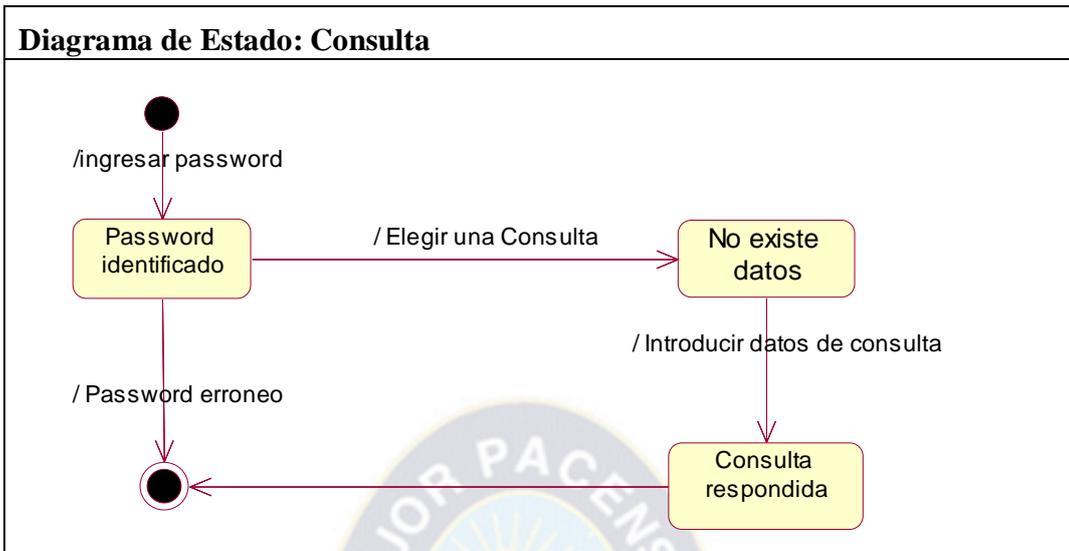


Figura. 3.14 Diagrama de Estado: Registrar Denuncia

b. Registro Mascota



Figura. 3.15 Diagrama de Estado: Registrar Propietario y Mascota



**Figura. 3.16 Diagrama de Estado: Consulta**



### 3.3.5 Diagramas de clases

El diagrama estático o de clases nos muestra todas las tablas y sus atributos, las relaciones que existe entre tablas con su respectiva cardinalidad, que se muestra en la Fig. 3.17

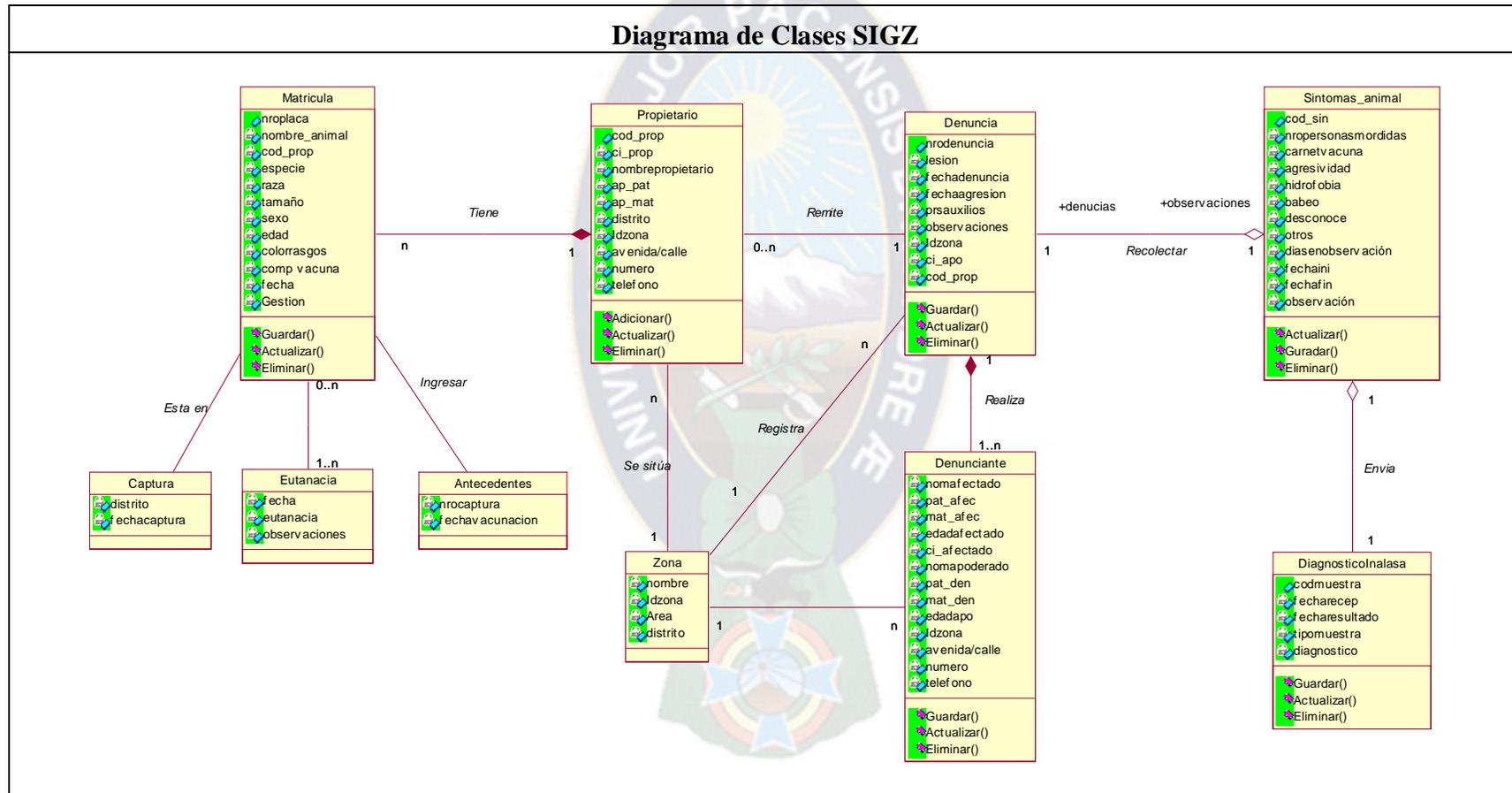


Figura. 3.17 Diagrama de Clases SIGZ

### 3.4 DISEÑO DE LA INTERFAZ GRÁFICA DEL SIG

El presente proyecto utiliza ArcView 3.3, por su simplicidad de uso, y como herramienta que permite realizar el análisis espacial, haciendo de planos georreferenciados de la ciudad de El Alto, además del almacenamiento espacial y la generación de capas temáticas, de las cuales MapObjects hace uso, Al igual que muchos programas trabaja con la interfaz estándar de Windows, es una ventaja que nos permite administrar con mucha facilidad.

ArcView cuenta con elementos principales en la ventana de aplicación que se organiza de la siguiente manera, el menú principal, la barra de botones, la barra de herramientas, y la ventana de proyecto que recoge los componentes del proyecto (vistas, tablas, gráficos, capas y escritura).

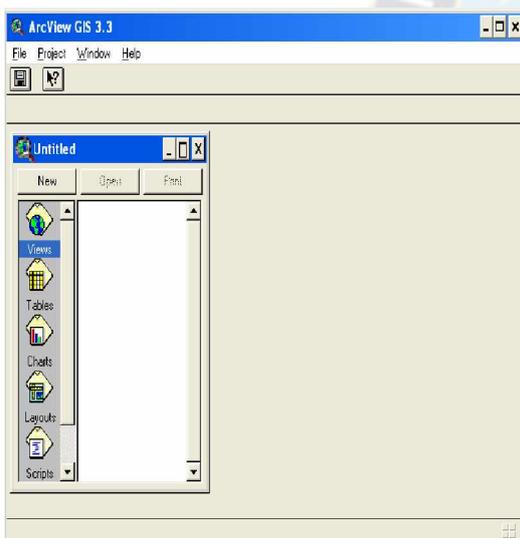


Figura. 3.18 Ventana ArcView

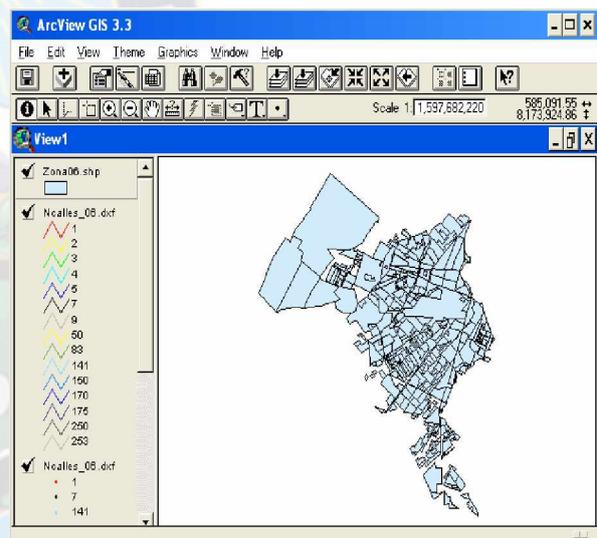


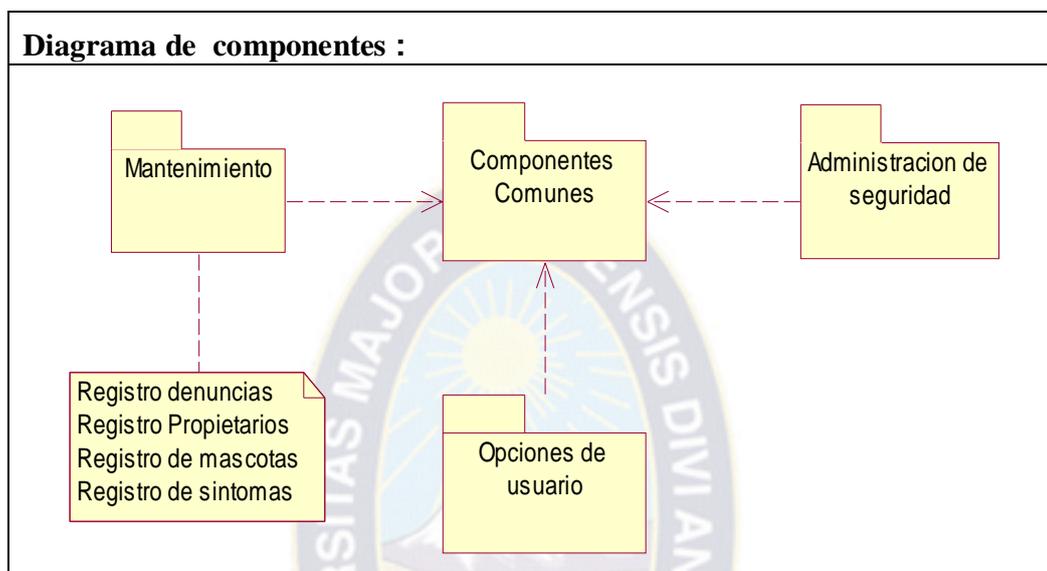
Figura. 3.19 Mapa catastral de El Alto

### 3.5 FASE DE CONSTRUCCIÓN

En esta fase se explica el diseño y la implantación del proyecto tomando en cuenta la especificación de los componentes, de los nodos, y de la interfaz. Para esto se describirá el diagrama de componentes, diagrama de despliegue, diagrama jerárquico de la interfaz y los formularios del sistema.

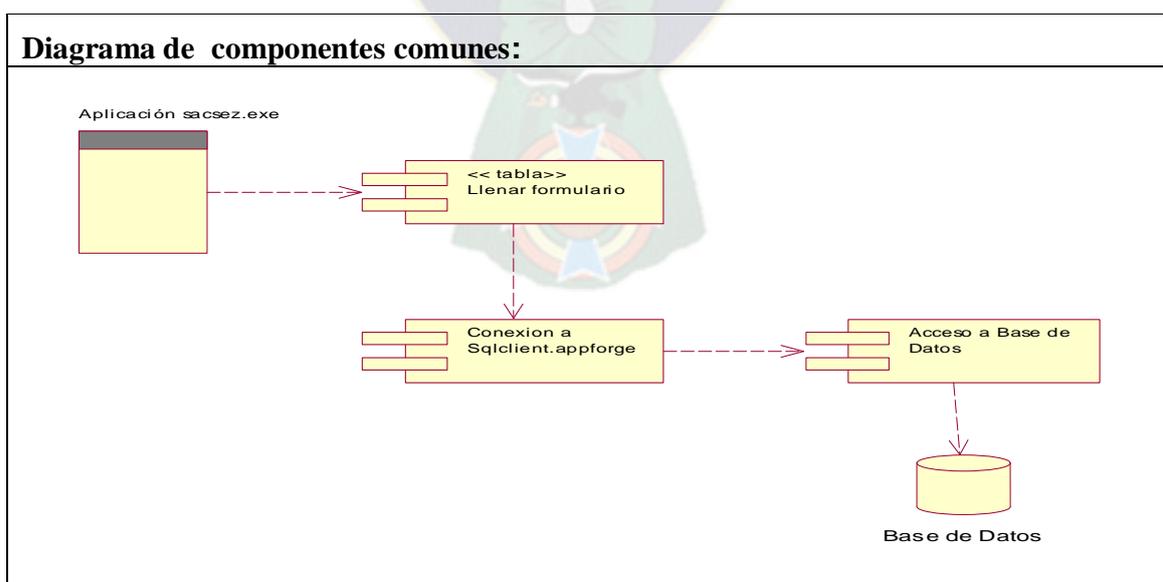
### 3.5.1 Diagrama de Componentes

En el diagrama nos describe como los elementos del modelo de diseño son empaquetados, mostrándonos los componentes de todo el sistema.



**Figura. 3.20 Diagrama de componentes SIGZ**

El diagrama de componentes comunes de la Fig. 3.21. Se muestra los componentes que son utilizados por todos los componentes del sistema para el acceso a la información



**Figura. 3.21**

### 3.5.2 Diagrama de Despliegue

Muestra los nodos a nivel físico y cliente.

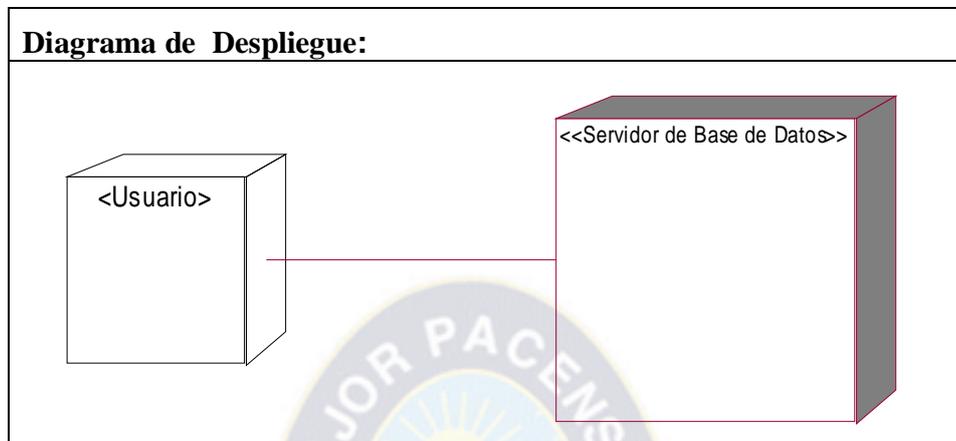
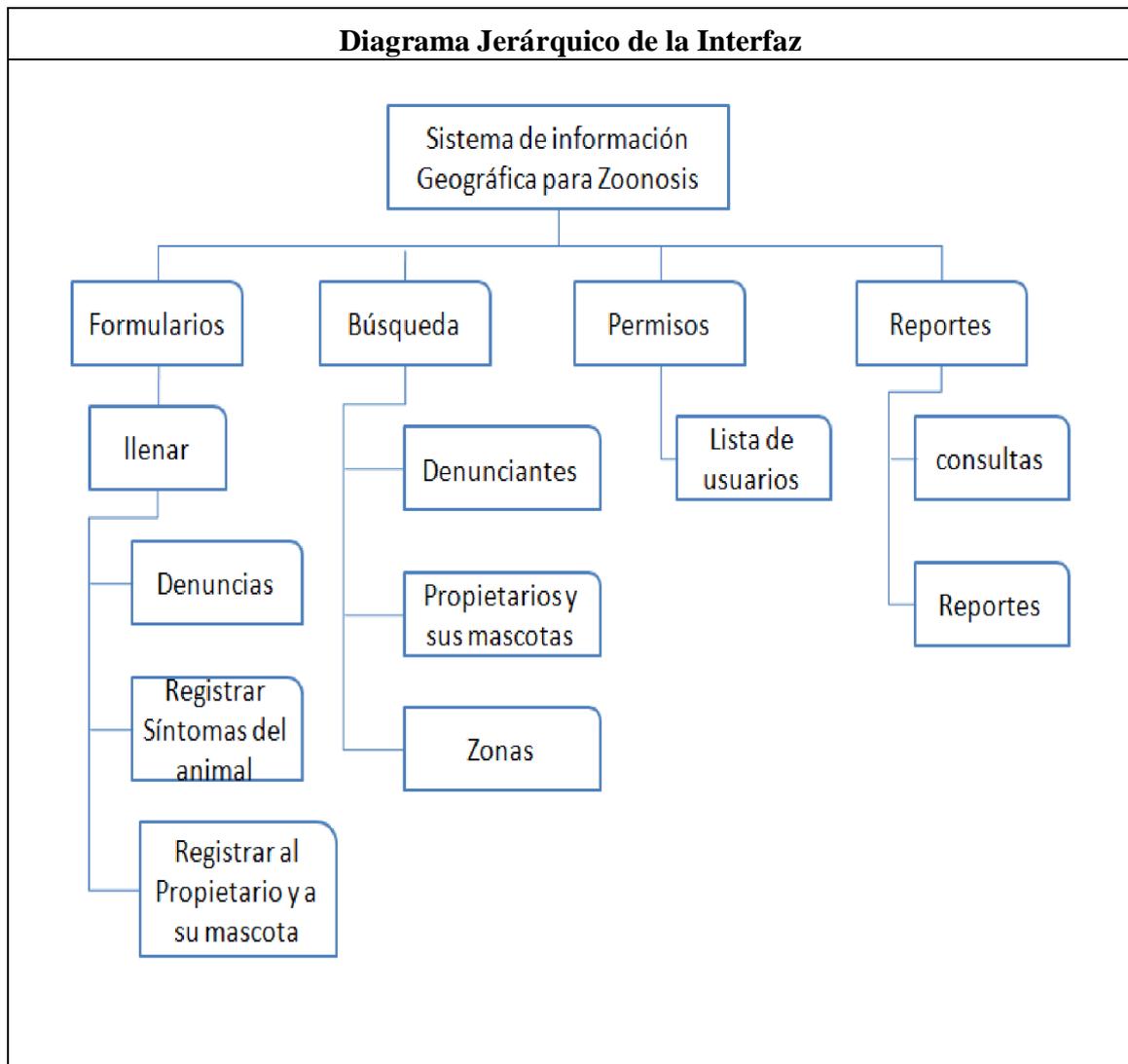


Figura. 3.22 Diagrama de despliegue a nivel de instancia del Sistema

### 3.5.3 Diagrama Jerárquico de la Interfaz

Muestra la jerarquía de los módulos dentro de una tarea: identificación de usuario, pantalla principal, menú general, formularios, consultas e informes.



**Figura. 3.23 Diagrama jerárquico que muestra el prototipo inicial de la interfaz del Sistema de Información Geográfica para Zoonosis**

### 3.5.4 Implementación

En la implementación nos basamos íntegramente en los objetos que se utiliza y las acciones que se realizan en la fase de diseño.

En la Figura. 3.24, se presenta la pantalla inicial del sistema, el que se muestra en el formulario de usuario, donde el usuario debe ingresar el nombre de usuario, y la contraseña que son únicos para cada usuario, cada usuario tiene solo tres intentos para ingresar al sistema.



**Figura. 3.24 Pantalla de Ingreso al Sistema**

En la Figura. 3.25 nos muestra la ventana principal, presenta opciones de menú que permite interactuar con el usuario, brindándoles una serie de alternativas de consultas y visualizaciones



**Figura. 3.25 Interfaz del menú principal**

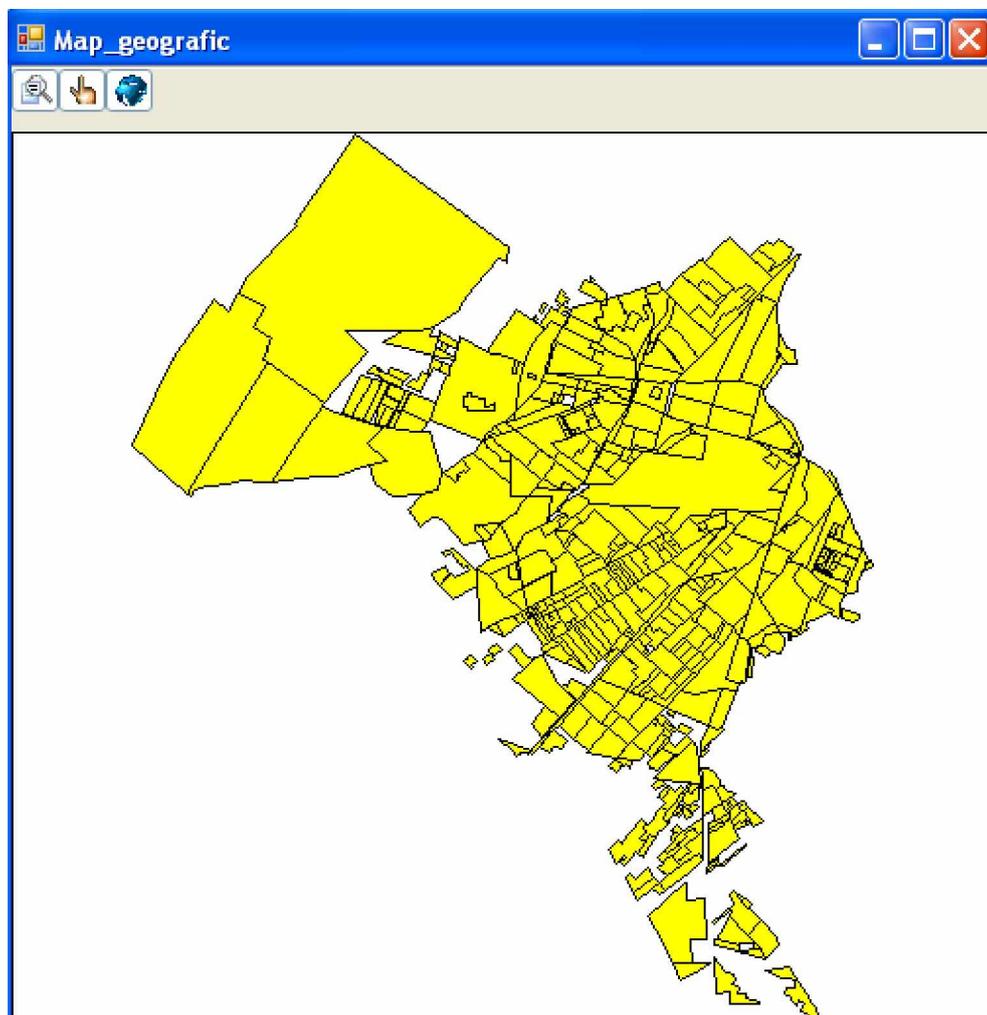
La ventana del formulario registro de denuncias, donde se ve que se encuentra los datos del afectado y del denunciante, posteriormente se tiene dos opciones, una nos muestra la zona en el mapa catastral donde ocurrió el caso, y la otra opción si el denunciante conoce el numero de domicilio del propietario del can, y realizar una búsqueda con el numero de puerta, verificando si el propietario esta registrado como se muestra en la Figura. 3.26.

**Figura. 3.26 Formulario de Denuncias**

En la Figura. 3.27 nos muestra la ventana de búsquedas de zonas por distritos, en el cual se selecciona el distrito y consecuentemente solo aparecen las zonas que pertenecen al distrito anteriormente seleccionado, presionando el botón mostrar mapa nos muestra la ubicación de la zona.

**Figura. 3.27 Consulta de búsqueda de Zonas**

En la Figura. 3.28 nos muestra el mapa catastral de La ciudad de El Alto.



**Figura. 3.28 visualización del mapa**

En la opción registrar Propietario nos muestra el formulario para adicionar, modificar y eliminar propietarios y una opción para registrar a sus mascotas, como se muestra en la Figura. 3.29

**Figura. 3.29 Formulario Propietario**

En la opción registrar Mascota nos muestra el formulario para registrar a sus mascotas y una opción de búsqueda por número de matrícula para modificar y eliminar a la mascota, como se muestra en la Figura. 3.30

**Figura. 3.30 Formulario de Matrícula**

### **3.8 FASE DE TRANSICIÓN.**

En esta fase luego de haber concluido con la fase de implementación y de haber realizado las pruebas necesarias dentro de su desarrollo se obtendrá el producto finalizado y listo para experimentarlo en el entorno de los usuarios y manipulen el sistema con datos reales, y dar un compas de espera para ver si el usuario no tiene ningún inconveniente, o si el sistema tiene algún error, o provoca una salida no deseada dar el pronto mantenimiento de dicho sistema.



## CAPITULO IV

### CALIDAD

#### 4.1 INTRODUCCIÓN

La calidad es uno de los aspectos más importantes para el desarrollo del software. Lo que en este capítulo se tratará estrictamente la medición de la calidad mediante las métricas de calidad mencionadas en el capítulo 2.

El objetivo no es necesariamente alcanzar una calidad perfecta, sino la necesaria y la suficiente para cada contexto de uso a la hora de la entrega y el uso por parte de los usuarios. Los puntos de vista que se tomo para la calidad son:

6. Funcionalidad
7. Confiabilidad
8. Facilidad de uso
9. Facilidad de mantenimiento
10. Portabilidad

Tomando como base el estándar ISO 9126, que contiene factores de calidad, se desarrollan los mismos a continuación.

#### 4.2 FUNCIONALIDAD

La funcionalidad de sistema, será evaluado aplicando el **punto de función**, para aplicar esta métrica, se deben determinar cinco características de dominio de función: Numero de entradas de usuario, número de salidas de usuarios, número de peticiones de usuario, número de archivos, número de interfaces externas.

<p><b>Número de entradas de usuario</b></p>	<p>Registro de denuncias de personas que fueron por agredidos por canes  Registrar al denunciante o afectado  Registro de los propietarios.  Registro de los animales domésticos.  Registro de Síntomas de la mascota.  Registro de Resultados de Inlasa.  Registrar de capturas.  Registrar de antecedentes.</p>
<p><b>Número de salidas de usuario</b></p>	<p>Vista espacial de zonas, donde se presume que existe rabia.  Consultas de los denunciantes que fueron agredidos.  Consulta de los propietarios y sus mascotas.  Reporte de Matricula.  Reportes de las zonas donde se presume que hay índices de rabia.  Reporte de los casos de rabia que se detectaron.  Reporte de Síntomas de las mascotas.  Mensajes de error, advertencia y sugerencia</p>
<p><b>Consultas Externas</b></p>	<p>Realizar la Matricula de la mascota.  Información de denuncias de personas que fueron por agredidos por canes  Información de los propietarios.  Información de sus mascotas.</p>
<p><b>Numero de archivos</b></p>	<p>Usuario.</p>
<p><b>Numero de interfaces externas</b></p>	<p>Ninguna.</p>

**Tabla 4.1 Descripción de parámetros de medición.**

**Fuente: Elaboración propia.**

Una vez que se tiene los dominios de la información, se debe asociar un valor de complejidad.

Parámetros de medición	cuenta	Factor de ponderación		Complejo	Resultado
		Simple	Medio		
Número de entradas de usuario	9	3	4	6	36
Número de salidas de usuario	8	4	5	7	40
Número de peticiones de usuario	5	3	4	6	20
Numero de archivos	1	7	10	15	10
Número de interfaces externas	0	5	7	10	0
<b>Cuenta Total</b>					106

**Tabla 4.2 Calculo de Puntos Función**  
Fuente: Elaboración propia

Para calcular puntos Función, se utiliza la siguiente relación:

$$PF = \text{Cuneta Total} * (0.65 + 0.01 * \sum(Fi))$$

Ahora, se debe calcular los valores de ajuste de la complejidad:  $\sum(Fi)$ , tomando en cuenta las respuestas a las siguientes preguntas que se muestran a continuación (4.2).

Valor	Descripción
0	No influencia
1	Incidental
2	Moderado
3	Medio
4	Significativo
5	Esencial

<b>Nro.</b>	<b>Valor de Ajuste de Complejidad</b>	<b>Valor</b>
<b>1</b>	¿Requiere el sistema copias de seguridad y recuperación flexible?	5
<b>2</b>	¿Se requiere comunicación de datos?	4
<b>3</b>	¿Existe funciones de procesamiento distribuido?	1
<b>4</b>	¿Es crítico el rendimiento?	0
<b>5</b>	¿Se ejecutara el sistema en un entorno operativo existente y fuertemente utilizado?	5
<b>6</b>	¿Requiere el sistema entrada de datos interactiva?	2
<b>7</b>	¿Requiere la entrada de datos interactiva que las transacciones de entrada se lleven a cabo sobre múltiples pantallas u operaciones?	0
<b>8</b>	¿Se utiliza los archivos maestros de forma interactiva?	4
<b>9</b>	¿Son complejos las entradas, las salidas, los archivos y las peticiones?	0
<b>10</b>	¿Es complejo el procesamiento interno?	4
<b>11</b>	¿Se ha diseñado el código para ser reutilizable?	4
<b>12</b>	¿Están incluidas en el diseño la conversión y la instalación?	3
<b>13</b>	¿Se ha desarrollado el sistema para soportar múltiples instalaciones en diferentes organizaciones?	3
<b>14</b>	¿Se ha diseñado la aplicación para facilitar los cambios y para ser fácilmente utilizado por el usuario?	5

**Tabla 4.3 Calculo de los valores de ajuste de la complejidad**  
**Fuente: Elaboración propia**

Entonces,  $\sum (Fi)$  tendrá el siguiente valor:

$$\sum (Fi) = 5 + 4 + 1 + 0 + 5 + 2 + 0 + 4 + 0 + 4 + 4 + 3 + 3 + 5 = 40$$

De todo lo anterior se obtiene el resultado:

$$PF = \text{Cuenta Total} * (0.65 + 0.01 * \sum (Fi))$$

$$PF = 106 * (0.65 + 0.01 * 40) = 111.3$$

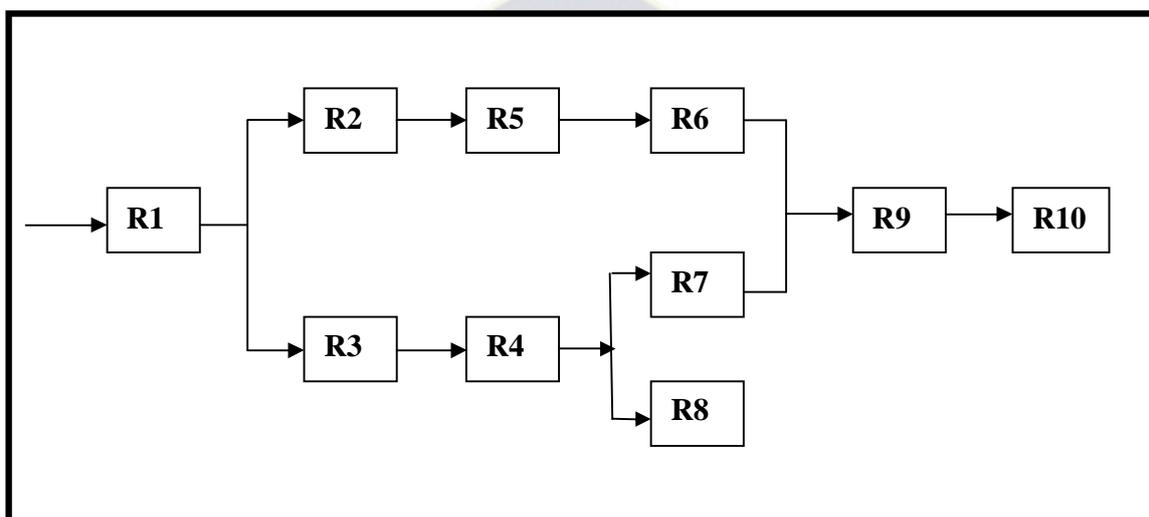
El valor de PF se encuentra en la siguiente escala:

- [300,+) : Óptima
- [200, 300) : Buena
- [100, 200) : Suficiente
- [0, 100) : Deficiente

Como se observa, el resultado de la aplicación del Punto Función es 111.3. Por lo tanto el sistema tendrá una buena funcionalidad de acuerdo a la escala en la que se encuentra el resultado obtenido.

### 4.3 CONFIABILIDAD

La confiabilidad, se la determina utilizando la teoría que se especifico en el capítulo 2. Para tal efecto, es necesario especificar el siguiente diagrama de transferencia detallando sus componentes:



**Figura 4.1 Diagrama de transferencia de módulos**  
**Fuente: Elaboración propia**

Donde:

- R1: Registro de Denuncias.
- R2: Registrar al denunciante o afectado
- R3: Registro de los propietarios.
- R4: Registro de los animales domésticos.
- R5: Registro de Síntomas de la mascota.
- R6: Registro de Resultados de Inlasa.
- R7: Registrar de capturas.
- R8: Registrar de antecedentes.
- R9: Consultas.
- R10: Reportes.

La confiabilidad de cada modulo del diagrama, esta especificado en la siguiente tabla:

$\lambda$  : Probabilidad de que se presente una perturbación.

**P (T)**: Probabilidad de fallo en caso de perturbación.

<b>R<sub>i</sub>(t)</b>	$\lambda$	<b>P (T)</b>	$e - \lambda P(T)$
<b>R1(t)</b>	0.2	0.1	0.98
<b>R2(t)</b>	0.15	0.1	0.99
<b>R3(t)</b>	0.2	0.1	0.98
<b>R4(t)</b>	0.15	0.1	0.99
<b>R5(t)</b>	0.15	0.1	0.99
<b>R6(t)</b>	0.2	0.1	0.98
<b>R7(t)</b>	0.2	0.1	0.98
<b>R8(t)</b>	0.2	0.1	0.98
<b>R9(t)</b>	0.15	0.1	0.99
<b>R9(t)</b>	0.2	0.1	0.98

**Tabla 4.4** Calculo de la confiabilidad de cada modulo  
**Fuente:** Elaboración propia.

Ahora se debe calcular la confiabilidad del sistema en su totalidad:

$$A=R2+R3$$

$$B=R3*R4*R7$$

$$C=R3*R4*R8$$

$$D=R2*R5*R6$$

$$E=R9+R10$$

$$A=1-(1-0.99)(1-0.98)=0.98$$

$$B=0.98*0.99*0.98=0.95$$

$$C=0.98*0.99*0.98=0.95$$

$$D=0.99*0.99*0.98=0.96$$

$$E=1-((1-0.99)(1-0.98))=0.98$$

Entonces la confiabilidad del sistema está dada por:

$$R(t) = R1(t) * A * B * C * D * E$$

$$R(t) = 0.98 * 0.98 * 0.95 * 0.95 * 0.96 * 0.98$$

$$R(t) = 0.82$$

Entonces el nivel de confiabilidad del sistema es del 82%.

#### 4.4 FACILIDAD DE USO

Para evaluar la usabilidad, se observó el desenvolvimiento del sistema interactuando con los usuarios, una vez realizada la observación, se utilizaron alguna de las medidas cuantificables de Nielsen y se calificaron las mismas en una escala del 0 % al 100 %, lo que significa que mientras más cercana este la calificación al 100 %, el sistema tendrá mejor usabilidad.

A continuación se detallan las medidas y sus respectivas calificaciones:

Medida	Calificación	Observación
Tiempo que los usuarios se toman para completar tareas específicas.	92 %	Calificación media obtenida
El número de tareas de diversos tipos que pueden ser completadas dentro de un tiempo límite dado.	94 %	Calificación media obtenida
La relación entre las interacciones exitosas y los errores.	92 %	Calificación media obtenida
El tiempo empleado en la recuperación de los errores.	80 %	Calificación media obtenida
El número de errores de usuario	85 %	Calificación media obtenida
El número de acciones erróneas inmediatamente posteriores.	75 %	Calificación media obtenida
La frecuencia del uso de manuales y/o o ayuda del sistema resolviendo el problema del usuario.	80 %	Calificación media obtenida
El número de ocasiones que el usuario se debía de la tarea real	85 %	Calificación media obtenida

**Tabla 4.5 Calculo de Puntos Función**  
**Fuente: Elaboración propia**

Entonces el sistema tiene una usabilidad del 85%.

El lector, podrá evaluar desde su propio punto de vista la usabilidad, en lo que respecta a la facilidad de aprendizaje, donde se encuentra el manual de usuario y así podrá sacar sus propias conclusiones en cuanto se refiere a la usabilidad del sistema (específicamente la facilidad de aprendizaje).

#### 4.5 FACILIDAD DE MANTENIMIENTO EFICIENCIA

La eficiencia del sistema, se evaluara tomando en cuenta a algunas medidas realizadas anteriormente para la usabilidad.

##### Comportamiento temporal

El tiempo que los usuarios se toman para contemplar tareas específicas y el número de tareas de diversos tipos que puedan ser completadas dentro de un tiempo límite dado, nos dan una estimación del comportamiento temporal del sistema. En este sentido, la calificación que obtiene el sistema es de 90.5 %

##### Utilización de los recursos

En cuanto a la utilización de recursos, el sistema utiliza el lenguaje SQL Server para adición, modificación y eliminación de registros. Además para la elaboración de informes, el sistema utiliza la aplicación Crystal Reports. Esto con el propósito de permitir al usuario de poder exportar los reportes a diferentes formatos como ser Adobe Acrobat (pdf), Microsoft Excel (xls) y Microsoft Word (doc).

$$IMS = [M_t - (F_a + F_c + F_d)] / M_t$$

Donde:

La fórmula está dada de la siguiente manera:

$M_t$  = el número de módulos en la versión actual

$F_c$  = el número de módulos cambiados en la versión actual

$F_a$  = el número de módulos añadidos a la versión actual

$F_d$  = el número de módulos de la versión anterior que se eliminaron en la actual

En el transcurso del desarrollo del sistema, se desarrollan cinco versiones del mismo, cada versión se modifica periódicamente con el fin de cubrir a cabalidad los requerimientos de los usuarios, las necesidades y los requerimientos cambian a medida que se avanza el desarrollo del sistema.

A continuación se detalla la información que se requiere para calcular el Índice de Madurez de software, en cada versión del sistema que se desarrollo en el transcurso de su diseño e implementación (Tabla 4.6).

<b>Versión del sistema</b>	<b>Mt</b>	<b><math>F_c</math></b>	<b><math>F_a</math></b>	<b><math>F_d</math></b>	<b>IMS</b>
Ver. 1.0	2	0	0	1	0.5
Ver. 2.0	3	0	1	0	0.67
Ver. 3.0	6	0	3	0	0.5
Ver. 4.0	8	0	2	0	0.75

**Tabla 4.6 Métrica IMS**

De la tabla 4.6, se observa los resultados que se observan en la Tabla 4.7

<b>Versión del sistema</b>	<b>Ver. 1.0</b>	<b>Ver. 2.0</b>	<b>Ver. 3.0</b>	<b>Ver. 4.0</b>
<b>IMS</b>	<b>0.5</b>	<b>0.67</b>	<b>0.5</b>	<b>0.75</b>

**Tabla 4.7 Valores del IMS en cada versión**  
**Fuente: Elaboración propia**

Cuando el IMS se acerca a 1, el software se empieza a estabilizar. Se observa en la Tabla 4.7 el sistema se empieza a estabilizar a partir de la versión cuatro, donde alcanza su máximo valor, entonces se puede afirmar que el sistema tiene un 75 % de estabilidad.

## 4.6 PORTABILIDAD

El sistema está desarrollado para ejecutarse en plataforma Windows NT, Windows 2000, Windows XP, Windows Server 2003. Por lo tanto puede ser ejecutado en cualquier computador que posee alguno de los sistemas operativos mencionados anteriormente.

El sistema está desarrollado en plataforma .Net, por lo que es necesario instalar **Framework 2.0** y **Mapobject 2.0**. La base de datos esta implementada con SQL Server 2000 por lo tanto se requiere un servidor que tenga instalado este gestor de Base de Datos. Los informes que se presenta en el sistema están elaborados y presentados con Crystal Reports.

## 4.7 ANÁLISIS COSTO/BENEFICIO

Entre la información más relevante de análisis costo beneficio es la evaluación de la justificación económica de un proyecto basado en computadora. El análisis costo beneficio señala los costos de desarrollo del proyecto y los contrasta con los beneficios tangibles e intangibles del sistema [PR07].

### 4.7.1 Análisis de costos

“El propósito de esta actividad, desde luego, es calcular todos los costos anticipados asociados con el sistema: no sólo el costo de construirlo, sino también el instalarlo” [You, 1993].

1. Costo de instalación (ver Tabla 4.2)
2. Costo de construcción (ver Tabla 4.3)
3. Costo de hardware: la empresa cuenta con las licencias necesarias para el desarrollo del proyecto

<b>DESCRIPCIÓN</b>	<b>TOTAL</b>
Costo de capacitación a usuarios	100.00 \$
Gastos de licencia software MapObject	150.00 \$
Gastos de licencia software ArcView	1500.00\$
Costos de pruebas	100.00 \$
Manual de mantenimiento	25.00 \$
<b>TOTAL COSTO DE INSTALACIÓN</b>	<b>1875.00 \$</b>

**Tabla 4.8 Costo de instalación**

<b>DESCRIPCIÓN</b>	<b>TOTAL</b>
Analista / programador	900.00 \$
Costos de documentación del análisis, diseño, e implementación	100.00 \$
<b>TOTAL COSTO DE CONSTRUCCIÓN</b>	<b>1000.00 \$</b>

**Tabla 4.9 Costo de construcción**

<b>DESCRIPCIÓN TOTAL</b>	<b>TOTAL</b>
TOTAL COSTO DE INSTALACIÓN	1875.00 \$
TOTAL COSTO DE CONSTRUCCIÓN	1000.00 \$
<b>TOTAL COSTO DE CONSTRUCCIÓN</b>	<b>2875.00 \$</b>

**Tabla 4.10 Costo Total**

#### 4.7.2 Análisis de beneficios

Los beneficios que se obtiene con el SIGZ a la implementación del mismo se especifican en la Tabla 4.5

	<b>Beneficios</b>	<b>Actividades</b>	<b>Sistema Actual</b>
SISTEMA DE INFORMACIÓN GEOGRÁFICA PARA ZONOSIS	Ahorro en tiempo en el registro de denuncias, de los propietarios y sus Mascotas	Y Registro de Denuncias	1-2 Min.
		Y Registro de propietarios y Mascotas	1-2 Min.
		Y Impresión de Denuncias, Matricula de la Mascota	15 Seg.
	Ahorro en tiempo de consultas para una toma rápida de decisiones	Y Reportes de las zonas donde se presume que existiera	1-2 Min.
		Y Reporte de propietarios y sus Mascotas	1-2 Min.

**Tabla 4.11 Cuadro de los Beneficios con el SIGZ**

## CAPÍTULO V

### CONCLUSIONES Y RECOMENDACIONES

#### 5.1 CONCLUSIONES

Se implementó el Sistema de Información Geográfico para Zoonosis, siendo una herramienta, que ayuda a la mejora de las decisiones y el suministro de la información de las zonas territoriales.

El lenguaje de modelado Unificado (UML), tiene una gran facilidad de entendimiento para los usuarios principiantes como para los usuarios avanzados, ya que expresa al sistema SIGZ en su totalidad mediante diagramas que explican toda la parte de desarrollo del sistema, los cuales ayudan a su comprensión más que operaciones narrativas escritas.

La metodología RUP optimiza el tiempo de desarrollo del sistema SIGZ, y como RUP es muy flexible ayuda a realizar cambios en el mismo sistema sin tener ningún problema. Se logró organizar de mejor manera los flujos de trabajo de: requerimientos, análisis, diseño, implementación, pruebas dando como resultado una evolución confiable del desarrollo del sistema, en todos estos flujos.

Comparando el área de zoonosis sin el sistema versus el sistema implementado, se ve claramente los beneficios que trajo el sistema SIGZ a esta área como ser:

- Y Mediante el proceso se consiguió desarrollar e implementar el “Sistema de Información Geográfico para zoonosis” el cual representa en forma gráfica la ubicación geográfica de los componentes de zonas.
- Y Mediante el Proceso se consiguió tiempo de respuesta oportuna para cualquier consulta, con respecto a la información que se maneja en esta institución. Esta información es mostrada mediante informes y reportes para el usuario.

- Y El Sistema contempla mecanismos que proporciona información georreferenciada para la toma de decisiones, y realizar prevenciones, evitar la propagación de la rabia.
- Y Con la visualización de los datos se puede tomar decisiones inmediatas.
- Y El sistema cumple con el objetivo general y los objetivos específicos del proyecto, dando respuesta a la problemática planteada, también se cumplieron los requerimientos para el sistema, de esta manera se logró implementar el “Sistema de Información Geográfica para Zoonosis” Esto se demuestra con el aval de la Unidad de Zoonosis G.M.E.A.

## **5.2 RECOMENDACIONES**

De la experiencia recogida en este trabajo se pone de manifiesto la dificultad para obtener la información adecuada, actualizada y estructurada que permita a instituciones como la Unidad de Zoonosis G.M.E.A. sacar provecho a aplicaciones SIG.

- Y Conforme la ciudad de El Alto vaya creciendo se debe complementar los mapas de los distritos 9 y 15 que aún son consideradas como comunidades.
- Y Implementar un módulo de interfaz entre el gestor de Base de Datos SqlServer con el software ArcView para utilizar la información generada por esta de forma rápida.

## **BIBLIOGRAFÍA**

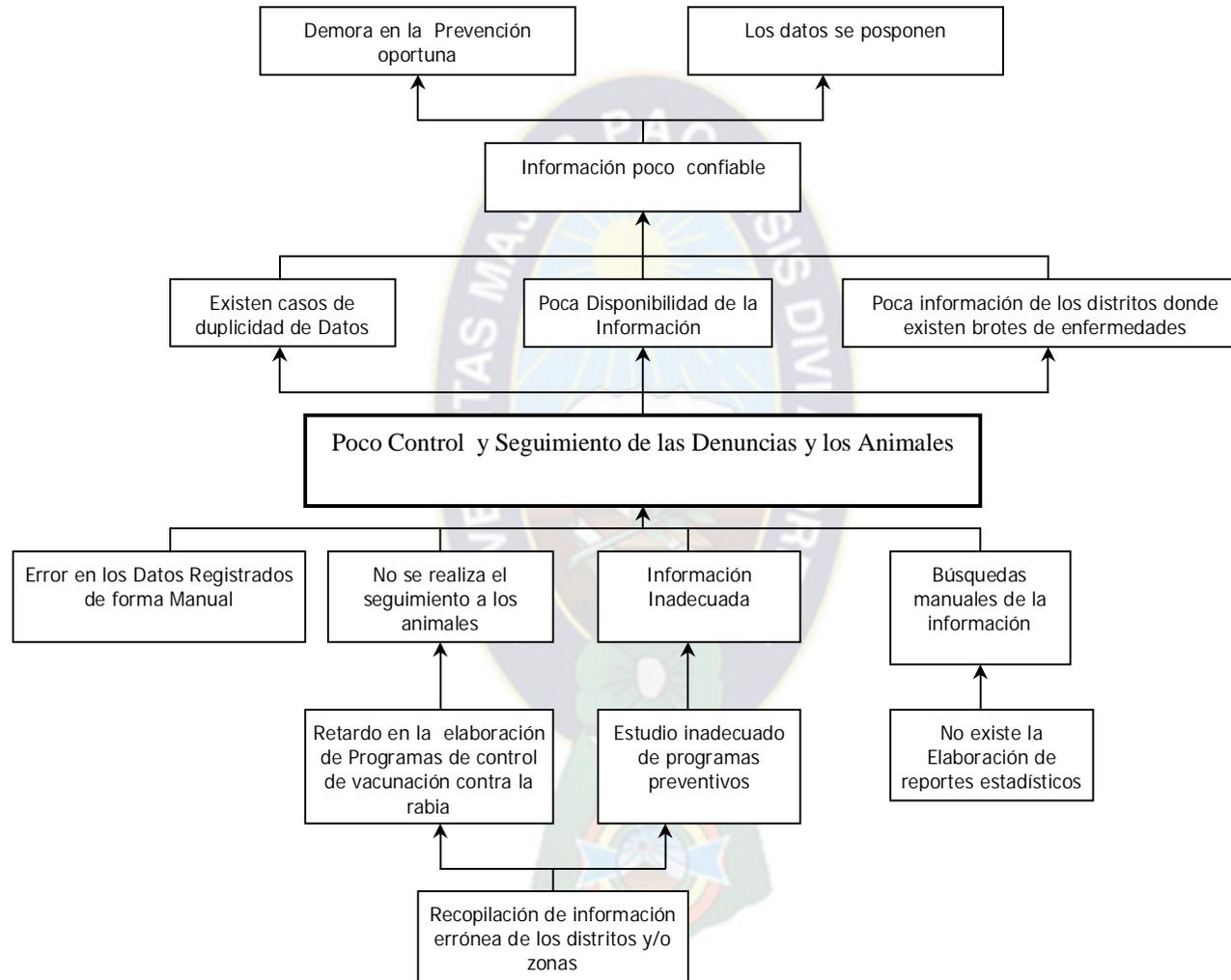
- [PR07] Roger S. Pressman (2007) Ingeniería del Software Un enfoque práctico Sexta Edición, España: Ed. McGraw-Hill.
- [JBR00] Ivar Jacobson, Grady Boch, James Rumbaugh. (2000) El Proceso Unificado de Desarrollo de software.
- [JBR99] Ivar Jacobson, Grady Boch, James Rumbaugh. (1999) El Lenguaje Unificado de Desarrollo de software.
- [KP96] Kruchten, P (1996) El Proceso Unificado de Desarrollo de software.
- [AS97] Pedro n. Acha, Boris Szyfres. (1997) Zoonosis y Enfermedades transmisibles comunes al hombre y a los animales.
- [PG03] Mario G. Piattini, Félix O. Garcia (2003) Calidad en el Desarrollo y Mantenimiento del Software.
- [PJ05] Julio Pinto Cortés MV. Ph.D, (2005) Sistema de Notificación de Enfermedades de la Lista Única de la Organización Mundial de Sanidad Animal
- [You93] Yourdon, E. (1993) Análisis Estructurado Moderno, España: Ed. McGraw-Hill

## **REFERENCIA WEB**

- [WEB01] <http://www.fao.org/ag/AGA/AGAH/Vets-1-2/5spa.htm>
- [WEB02] <http://www.ops-oms.org/Spanish/AD/DPC/CD/rbm-mesoamerica-2.pdf>
- [WEB03] <http://www.fao.org/regional/LAmerica/prior/segalim/animal/pdf/TADs.pdf>
- [WEB04] [http://www.paho.org/Spanish/DD/AIS/EB\\_v25n4.pdf](http://www.paho.org/Spanish/DD/AIS/EB_v25n4.pdf)
- [WEB05] <http://www.esri-es.com>
- [WEB06] <http://www.esri.com/devsupport/devconn/mapobjects/index.html>
- [WEB07] [http://www.cetus-links.org/oo\\_uml.pdf](http://www.cetus-links.org/oo_uml.pdf)
- [WEB08] <http://www.uml.org>
- [WEB09] <http://www.info-ab.uclm.es/assignaturas/42579>

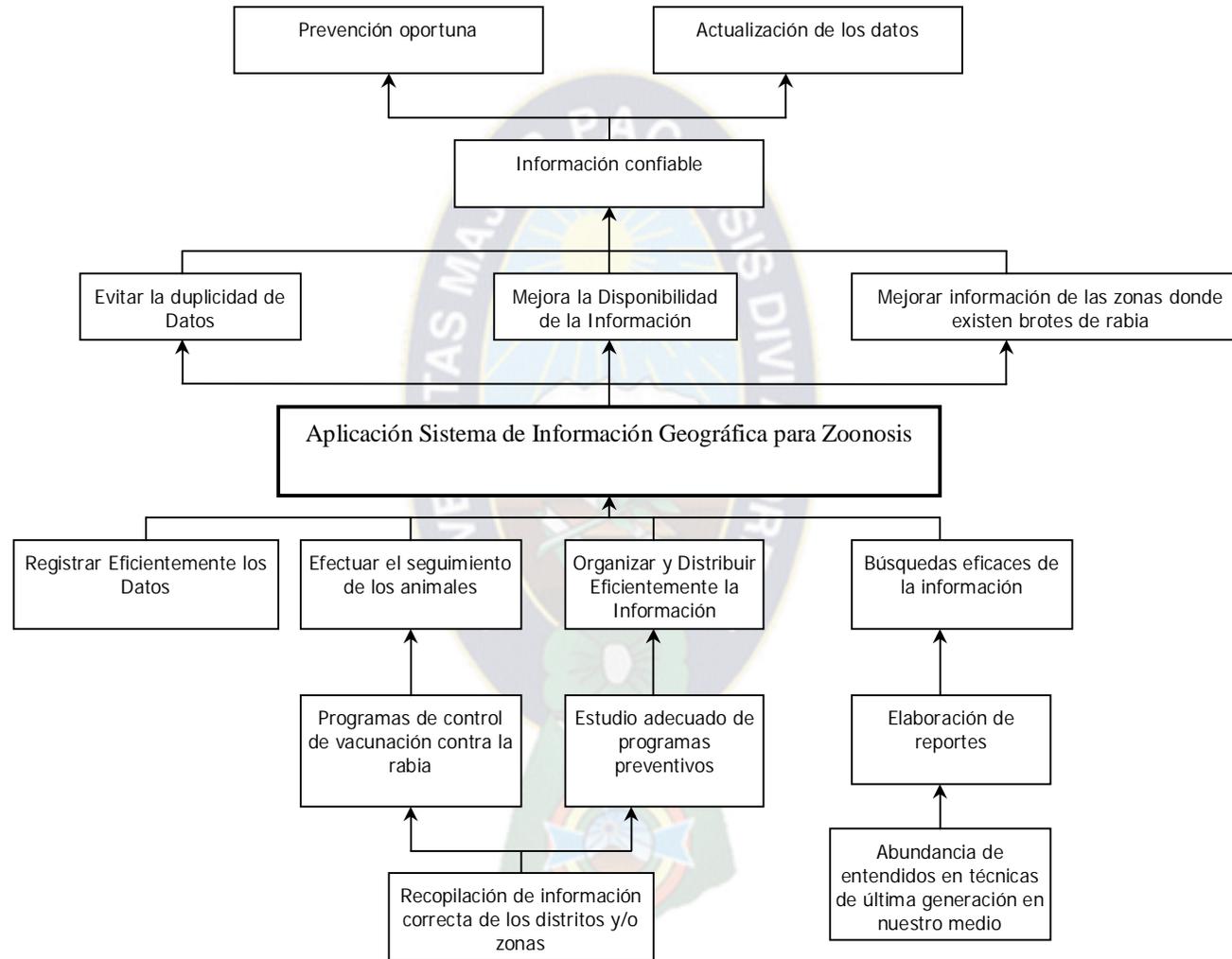
# ANEXO A

# ÁRBOL DE PROBLEMAS



## ANEXO B

## ÁRBOL DE OBJETIVOS



## ANEXO C

### CASOS DE USO ESENCIALES DE SIGZ

<b>Casos de uso:</b>	Registrar Denuncia
<b>Actores:</b>	Encargado de Zoonosis
<b>Propósito:</b>	Registrar la Denuncia y actualizar la base de datos
<b>Resumen:</b>	El encargado de Zoonosis, ingresa al sistema, llena los datos del denunciante, se verifica en que zona ocurrió, y realiza una consulta si el propietario esta registrado
<b>Tipo:</b>	Primario
<b>REFERENCIA CRUZADA</b>	
<b>CURSO NORMAL DE LOS EVENTOS</b>	
ACCIÓN DE LOS ACTORES	RESPUESTA DEL SISTEMA
<ol style="list-style-type: none"> <li>1. Introduce su password.</li> <li>3. a) Elige la opción registro denuncia.</li> <li>5. Escogerá en el menú de registrar denuncia</li> <li>7. Llenara el formulario con los datos que se solicitan.</li> <li>8. Si elige la opción mostrar la zona en el mapa.</li> <li>10. Si el denunciante conoce el numero de puerta del can agresor.</li> <li>12. Revisara y posteriormente elegirá la opción guardar.</li> <li>14. Después de realizar la operación, va al menú principal.</li> <li>16. Para salir del sistema elegir la opción salir.</li> </ol>	<ol style="list-style-type: none"> <li>2. El sistema comprueba si los datos introducidos son correctos</li> <li>a) Los datos son correctos, muestra la interfaz del menú principal.</li> <li>b) Los no son correctos, el sistema mostrara una nota de precaución al usuario.</li> <li>4. Mostrara una ventana con diferentes opciones</li> <li>6. Mostrara una ventana con el formulario vacio de denuncia para llenar.</li> <li>9. Mostrara una ventana donde se visualiza la zona.</li> <li>11. Se activa una opción y realiza una consulta de búsqueda por número de puerta en la base de datos.</li> <li>13. c) Si los datos están correctamente llenados, guardar y actualizar la base de datos.</li> <li>15. Saldrá al menú principal.</li> <li>17. Mostrara una ventana que nos indica que está en proceso de cierre.</li> </ol>

**Tabla C.1 Caso de uso: Registrar denuncia**

<b>Casos de uso:</b>	Registrar Propietario y Mascota
<b>Actores:</b>	Encargado de Zoonosis
<b>Propósito:</b>	Registrar los Datos del Propietario, de la Mascota y actualizar la base de datos.
<b>Resumen:</b>	El encargado de Zoonosis, ingresa al sistema, llena los datos del Propietario, posteriormente de la mascota.
<b>Tipo:</b>	Primario
<b>REFERENCIA CRUZADA</b>	
<b>CURSO NORMAL DE LOS EVENTOS</b>	
<b>ACCIÓN DE LOS ACTORES</b>	<b>RESPUESTA DEL SISTEMA</b>
<ol style="list-style-type: none"> <li>1. Introduce su password.</li> <li>3. a) Elige la opción registro Propietario.</li> <li>5. Escogerá en el menú de registrar denuncia</li> <li>7. Llenara el formulario con los datos que se solicitan.</li> <li>8. Revisara y posteriormente elegirá la opción guardar.</li> <li>10. Luego de registrar al propietario se presiona la opción registrar mascota.</li> <li>12. Revisara y posteriormente elegirá la opción guardar.</li> <li>13. Después de realizar la operación, va al menú principal.</li> <li>15. Para salir del sistema elegirá la opción salir.</li> </ol>	<ol style="list-style-type: none"> <li>2. El sistema comprueba si los datos introducidos son correctos</li> <li>a) Los datos son correctos, muestra la interfaz del menú principal.</li> <li>b) Los no son correctos, el sistema mostrara una nota de precaución al usuario.</li> <li>4. Mostrara una ventana con diferentes opciones</li> <li>6. Mostrara una ventana con el formulario vacio de propietario para llenar.</li> <li>9. c) Si los datos están correctamente llenados, guardar y actualizar la base de datos.</li> <li>11. d) Si los datos están correctamente llenados, guardar y actualizar la base de datos.</li> <li>14. Saldrá al menú principal.</li> <li>16. Mostrara una ventana que nos indica que está en proceso de cierre.</li> </ol>

**Tabla C.2 Caso de uso: Registrar denuncia**

<b>Casos de uso:</b>	Generación de Reportes	
<b>Actores:</b>	Encargado de Zoonosis	
<b>Propósito:</b>	Brindar respuesta oportuna a las consultas que tienen los actores mencionados.	
<b>Resumen:</b>	El encargado de Zoonosis, ingresa al sistema, se le dará la información que le compete de acuerdo a la consulta que realice el sistema.	
<b>Tipo:</b>	Primario	
<b>REFERENCIA CRUZADA</b>		
<b>CURSO NORMAL DE LOS EVENTOS</b>		
<b>ACCIÓN DE LOS ACTORES</b>	<b>RESPUESTA DEL SISTEMA</b>	
1. Introduce su password.	2. El sistema comprueba si los datos introducidos son correctos	
3. a) Elige la opción consultas.	c) Los datos son correctos, muestra la interfaz del menú principal.	
	d) Los no son correctos, el sistema mostrara una nota de precaución al usuario.	
4. Escogerá en el menú de registrar denuncia	5. Mostrara una ventana con diferentes opciones	
	6. Mostrara una ventana de consultas	
7. El usuario escogerá la opción de consultas	8. Muestra una ventana de la consulta elegida	
9. Llenara algunos datos que la consulta quiera y digitar aceptar.	9. Buscara la petición requerida, y mostrara una ventana con toda la información que se encontró.	
10. Se seguirá haciendo otras consultas, apretar botón realizar consulta.	12. Saldrá al menú principal.	
11. Después de realizar la operación, va al menú principal.	14. Mostrara una ventana que nos indica que está en proceso de cierre.	
13. Para salir del sistema elegir la opción salir.		

**Tabla C.3 Caso de uso: Generación de Reportes**

## ANEXO D

### CONTRATOS DE SIGZ

<b>Nombre:</b>	Consulta denuncia
<b>Responsabilidades:</b>	Esta consulta es disponible para el Encargado, desplegara los datos de la denuncia.
<b>Tipo:</b>	Interfaz registro de denuncia
<b>Referencias cruzadas:</b>	Caso de uso 1
<b>Notas:</b>	Utilizar el acceso rápido a la base de datos.
<b>Excepciones:</b>	Si el dato del denunciante no es válido saldrá una ventana informativa.
<b>Salida:</b>	
<b>Precondiciones:</b>	El sistema conoce el código del denunciante
<b>Poscondiciones:</b>	Si se trata de un nuevo denunciante se asigna un numero de denuncia.

**Tabla D.1 Contrato: Registrar denuncia**

<b>Nombre:</b>	Consulta Propietario y Mascota
<b>Responsabilidades:</b>	Esta consulta es disponible para el encargado, desplegara los datos de la Propietario y su mascota.
<b>Tipo:</b>	Interfaz registro propietario y mascota
<b>Referencias cruzadas:</b>	Caso de uso 2
<b>Notas:</b>	Utilizar el acceso rápido a la base de datos.
<b>Excepciones:</b>	Si el dato del propietario no es válido saldrá una ventana informativa.
<b>Salida:</b>	
<b>Precondiciones:</b>	El sistema conoce el código del Propietario
<b>Poscondiciones:</b>	Si se trata de un nuevo Propietario se asigna un código.

**Tabla D.2 Contrato: Registrar denuncia**

<b>Nombre:</b>	Generación de Matricula Mascota
<b>Responsabilidades:</b>	La generación de Matricula de la mascota emitido por el encargado
<b>Tipo:</b>	Sistema
<b>Referencias cruzadas:</b>	Caso de uso 3
<b>Notas:</b>	Utilizar el acceso rápido a la base de datos.
<b>Excepciones:</b>	Si el código del propietario no es válido saldrá una ventana informativa.
<b>Salida:</b>	
<b>Precondiciones:</b>	El sistema conoce el código del Propietario
<b>Poscondiciones:</b>	Si se trata de un nuevo Propietario se asigna un código.

**Tabla D.3 Contrato: Generación de Matricula Mascota**

<b>Nombre:</b>	Consulta zona
<b>Responsabilidades:</b>	Consulta las zonas donde se presume que existe rabia canina
<b>Tipo:</b>	Interfaz de actualización de datos
<b>Referencias cruzadas:</b>	Caso de uso 4
<b>Notas:</b>	Utilizar el acceso rápido a la base de datos.
<b>Excepciones:</b>	Si el código de usuario no es válido saldrá una ventana informativa.
<b>Salida:</b>	
<b>Precondiciones:</b>	El sistema conoce el código del usuario
<b>Poscondiciones:</b>	Se ingresa a la interfaz de consultas, y por último a la elección de la consulta

**Tabla D.4 Contrato: Consulta zona**