

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



PROYECTO DE GRADO

“Sistema de Control y Seguimiento de los Procesos
de Fiscalización Integral - GMLP”

PARA OPTAR AL TÍTULO DE LICENCIATURA EN INFORMÁTICA
MENCION INGENIERÍA DE SISTEMAS INFORMÁTICOS

Postulante: Geraldine Ocaña Gallardo

Tutor: M. Sc. Franz Cuevas Quiroz

Revisor: M. Sc. Luisa Velásquez López

La Paz – Bolivia

2006

DEDICATORIA

*Cuando veas un gigante,
Examina antes la posición del sol,
No vaya a ser la sombra de un pigmeo...
Von Hardenberg*

*A cada instante, he encontrado la posibilidad de elegir...
elegir si me permito ser... o elegir si me permito permanecer...
Hoy elegí permitirme estar...
en el momento preciso a la hora exacta...
y saberme agradecida por esta oportunidad...*

*Y... como esta es la parte en la que se dedica el esfuerzo de uno mismo,
por qué no obsequiárselo a uno mismo?
no... no es egoísmo... es amor propio!*

*A mí misma...
como prueba de amor,
y señal de cambio que abrirá un nuevo sendero...*

AGRADECIMIENTOS

*Toda idea tiene un principio,
Llega un momento en el que puede plasmarse, y después culminar...
Y durante todo este recorrido, pasan demasiadas personas,
Unas te saludan, otras se detienen a saber por tí,
Algunos te acompañan hasta la esquina,
Mientras otros más te apoyan y te ayudan hasta el final...*

*Han aparecido muchas de estas personas,
Mientras se desarrollaba este proyecto,
Pero, como en todo, son las de la última categoría...
Las que pueden entrar a este apartado!*

*Quiero expresar mis sinceros
Y espontáneos agradecimientos a estas personas:
A la Lic. Luisa Velásquez,
Por esa forma tan peculiar de demostrarme confianza,
Y por su empeño y dedicación para hacerme comprender
Que no hay nada mal hecho, sino solo equivocado...
Al Lic. Franz Cuevas,
Por enseñarme que todo tiene solución,
Y que el tiempo debe estar a disposición nuestra, y no al contrario...*

*Y para no hacer énfasis en una lista enorme de nombres...
Quiero agradecer a todas las personas de las demás clasificaciones
(y no es obstinación), solo que sin su presencia...
Los efectos no hubiesen sido los mismos!!*

RESUMEN

El Sistema de Control y Seguimiento de los Procesos de Fiscalización Integral es un software que contribuye en la fiscalización predial urbana de la Sub Alcaldía San Antonio, está encaminado hacia el registro, almacenamiento, control y seguimiento de la información, añadiendo una mejor forma de regir los procesos administrativos preestablecidos para procesar dicha información.

Los problemas a los que se enfrentaba dicha institución en su Unidad de Fiscalización, constituían en gran parte a un trabajo moroso de registro y almacenamiento de la información, por lo que la generación de reportes mensuales ocasionaba dificultades a los funcionarios. Además, la información estaba restringida solo para el encargado en jefe, siendo una ineficiente forma de coordinación administrativa, así como una pesada condición para controlar y seguir los procesos de la información.

El desarrollo de este sistema implica la aminoración a estos problemas, gracias a la aplicación de la metodología RUP (Proceso Racional Unificado) que utiliza la notación UML orientada a objetos para el análisis y diseño del sistema. El RUP presenta cuatro fases de desarrollo: Análisis, Diseño, Construcción e Implementación.

Y, en lo que se refiere al diseño navegacional del sistema, se utilizó la metodología OOHDM (Método de Diseño Hipermedia Orientado a Objetos). El sistema muestra interfaces web, con un servidor Apache XAMP, cuyo enlace programador es PHP, respondiendo a los requerimientos y necesidades del cliente.

Finalmente, los resultados obtenidos en las evaluaciones de calidad del software demuestran satisfactoriamente el grado de utilidad del sistema, en conclusión alrededor de un 84 por ciento de aprobación de parte de los clientes.

ÍNDICE

CAPÍTULO I	5
1 INTRODUCCIÓN	9
1.2 Antecedentes	10
<u>1.2.1</u> Marco Institucional.....	11
<u>1.2.2</u> Descripción de Procedimientos y Funciones.....	12
1.3 Identificación y Formulación del Problema.....	14
1.4 Objetivos.....	15
<u>1.4.1</u> Objetivo General	15
<u>1.4.2</u> Objetivos Específicos.....	16
1.5 Justificación.....	16
1.6 Alcances	16
1.7 Aportes.....	17
1.8 Metodología	17
1.9 Técnicas y Herramientas	18
CAPÍTULO II	19
MARCO TEÓRICO	19
2.1 Análisis y Diseño Orientada a Objetos.....	19
2.2 El Modelo UML.....	19
<u>2.2.1</u> Vocabulario de UML	21
<u>2.2.2</u> Mecanismos de Extensibilidad del UML.....	24
2.3 El Proceso Unificado de Desarrollo del Software	25
2.4 Fases y Flujos de Trabajo del Desarrollo del Software	28
<u>2.4.1</u> Fase de Inicio - Requisitos.....	29
<u>2.4.2</u> Fase de Elaboración - Análisis	31
<u>2.4.3</u> Fase de Construcción - Diseño	33
<u>2.4.4</u> Fase de Transición - Implementación	37
2.5 Entorno de Seguridad	40
2.6 Calidad del Software.....	41

2.7 Factores de Calidad ISO 9126	41
2.8 Calidad de la Aplicación Web.....	42
2.9 Metodología de Evaluación de Calidad de Sitios Web (Web-Site QEM)	42
CAPÍTULO III	45
MARCO APLICATIVO	45
3.1 Fase de Inicio - Requisitos	45
3.2 Fase de Elaboración - Análisis.....	54
3.3 Fase de Construcción - Diseño.....	59
3.4 Fase de Transición - Implementación.....	65
3.5 Seguridad del Sistema	71
CAPÍTULO IV	72
CALIDAD DEL SISTEMA	72
4.1 Calidad del Software.....	72
__4.1.1 Análisis de Punto Función.....	72
__4.1.2 Cálculo del Tiempo de Esfuerzo	76
4.2 Calidad en la Aplicación Web.....	76
__4.2.1 Evaluación Elemental.....	79
__4.2.2 Agregación de Preferencias Parciales	84
CAPÍTULO V	91
CONCLUSIONES Y RECOMENDACIONES	91
5.1 Conclusiones	91
5.2 Recomendaciones	92
BIBLIOGRAFÍA	93
ANEXOS	95

LISTA DE FIGURAS

Figura 1.1 Organigrama de la Unidad de Fiscalización Integral	4
Figura 1.2 Diagrama de los Procedimientos de la Unidad de Fiscalización	5
Figura 2.1 Diagramas de UML	14
Figura 2.2 Notación de los Estereotipos de UML	17
Figura 2.3 Un Proceso de Desarrollo de Software	17
Figura 2.4 Flujos de Trabajo VS Fases	20
Figura 2.5 Modelos y Flujos de Trabajo	21
Figura 2.6 Vista Clásica de una Arquitectura de Tres Capas	26
Figura 2.7 Definición de Clase Enlace	28
Figura 2.8 Traslación del Modelo Relacional al Modelo Orientado a Objetos	31
Figura 3.1 Modelo del Negocio - Procesos de la Unidad de Fiscalización	39
Figura 3.2 Diagrama de Actividades - Control y Seguimiento de Procesos	41
Figura 3.3 Diagrama de Casos de Uso del Sistema	44
Figura 3.4 Modelo Conceptual del Sistema de Control y Fiscalización	46
Figura 3.5 Diagrama de Colaboración del Caso de Uso Registro de Información	47
Figura 3.6 Diagrama de Colaboración del Caso de Uso Control de Inspección	47
Figura 3.7 Diagrama de Colaboración del Caso de Uso Seguimiento de Procesos	48
Figura 3.8 Métodos de las Clases de Control Gestor de Registros y Gestor de Consultas	49
Figura 3.9 Métodos de las Clases Entidad - Propietario, Vivienda, Inspección	49
Figura 3.10 Métodos de Clases Entidad – Infracción, Notificación, Proceso	49
Figura 3.11 Atributos de Clases Entidad – Propietario, Vivienda, Inspección	50
Figura 3.12 Atributos de Clases Entidad – Infracción, Notificación, Proceso	50
Figura 3.13 Diagrama de Asociaciones del Sistema	51
Figura 3.14 Diagrama de Clases – Registro Información	52
Figura 3.15 Diagrama de Clases del Proceso Particular	53
Figura 3.16 Diseño Navegacional – Administrador y/o Fiscalizador	54
Figura 3.17 Diseño Navegacional – Inspector	55
Figura 3.18 Diagrama de Componentes en general	56
Figura 3.19 Modelo de Despliegue – Procesos de Fiscalización	56
Figura 3.20 Modelo de Implementación – Control y Seguimiento de Procesos	57
Figura 3.21 Pantalla de Ingreso al Sistema de Fiscalización	61
Figura 3.22 Pantalla del Menú de Opciones	61
Figura 3.23 Pantalla de Listado de Propietarios	62
Figura 3.24 Pantalla de Registro de Propietarios	62

Figura 4.1 Estructura de Agregación para la característica Usabilidad	77
Figura 4.2 Estructura de Agregación para la característica Funcionalidad	77
Figura 4.3 Estructura de Agregación para la característica Confiabilidad	78
Figura 4.4 Estructura de Agregación para la característica Eficiencia	78
Figura 4.5 Estructura de Agregación para la característica Portabilidad	79
Figura 4.6 Estructura de Agregación para la característica Mantenibilidad	79
Figura 4.7 Estructura de Agregación de Preferencias Parciales y Globales	79

LISTA DE TABLAS

Tabla 1.1 Tabla de Funciones de los Empleados de la Unidad de Fiscalización	4
Tabla 2.1 Niveles de Aceptabilidad	36
Tabla 3.1 Actores del Sistema de Fiscalización	42
Tabla 3.2 Caso de Uso - Solicitar Servicio	42
Tabla 3.3 Caso de Uso - Registrar Datos	43
Tabla 3.4 Caso de Uso – Buscar Datos	43
Tabla 3.5 Caso de Uso – Actualizar Datos	43
Tabla 3.6 Caso de Uso – Control de Inspección	87
Tabla 3.7 Caso de Uso – Seguimiento del Proceso	87
Tabla 3.8 Caso de Uso – Emitir Reportes	87
Tabla 3.9 Caso de Uso expandido - Registrar Datos	45
Tabla 3.10 Caso de Uso expandido - Buscar Datos	45
Tabla 3.11 Caso de Uso expandido - Actualizar Datos	88
Tabla 3.12 Caso de Uso expandido - Control de Inspección	88
Tabla 3.13 Caso de Uso expandido - Seguimiento del Proceso	89
Tabla 3.14 Caso de Uso expandido - Emitir Reportes	90
Tabla 3.15 Base de Datos Normalizada del Sistema	58
Tabla 3.16 Niveles de Seguridad de Acceso al Sistema	63
Tabla 4.1 Identificación de los Componentes del Sistema	65
Tabla 4.2 Cálculo del PF con un Factor de Complejidad	66
Tabla 4.3 Escala de Parámetros de Evaluación	66
Tabla 4.4 Cálculo de los Valores de los Atributos de Ajuste	67
Tabla 4.5 Árbol de Requerimientos de Calidad	69
Tabla 4.6 Resultados Parciales y Globales de Calidad	80
Tabla 4.7 Indicadores Globales de Calidad para el Sistema Control y Seguimiento de Procesos de Fiscalización	82

CAPÍTULO I

1 INTRODUCCIÓN

Reconocemos la importancia de tener la capacidad de ver el mundo de los sistemas de información desde muchas perspectivas diferentes, [KENDALL & KENDALL, 1997].

De hecho, los sistemas de información son tantos como empresas, instituciones y organizaciones en general que existen en el mundo.

Los sistemas informáticos tienen como principal función el solucionar o reducir los problemas con carácter repetitivo. Tal es el caso de aquellas dificultades concebidas en el almacenamiento y archivo de información, incluso llevar la cuenta descriptiva de cada trabajo resulta imposible para la memoria humana, así también, cuando se tiene a un personal de trabajo es necesario saber quienes pueden conocer la información y quienes no, y hasta que niveles pueden acceder a la misma, de manera que se especifiquen las funciones de acceso de cada miembro del personal.

Por lo expuesto anteriormente, el presente proyecto contiene cinco capítulos, estrechamente relacionados y encaminados a solucionar los problemas mencionados.

En el Capítulo I, se puede distinguir el ambiente de trabajo o contexto con el que interactuará el sistema, conjuntamente se identifican los problemas de la Unidad de Fiscalización de la Sub Alcaldía San Antonio, para luego trazar objetivos que solucionen dichas dificultades; todo esto a través de la utilización de una metodología y técnicas previamente aprobadas.

El Capítulo II, muestra en general, el marco teórico que acompañara y respaldara el desarrollo del sistema, exponiendo así, un recorrido por las definiciones, metodología, herramientas y otros factores importantes para comprender el sistema.

Así mismo, el Capítulo III contiene el desarrollo del trabajo; el análisis, el diseño, y la implementación del sistema se contrastan en este capítulo para demostrar la adecuada aplicación de las concepciones del marco teórico.

De forma similar, el Capítulo IV recoge definiciones del marco teórico para comprobar y evaluar la calidad del software. Por esta razón, durante todo este capítulo se hallan datos cuantitativos que respaldan científicamente el desarrollo del sistema.

Finalmente, el Capítulo V traduce los resultados obtenidos, en los anteriores capítulos, en conclusiones. Dichas conclusiones revelan cuantitativamente si los objetivos especificados por los problemas identificados, se han cumplido satisfactoriamente como se requerían. Asimismo, se detallan recomendaciones para ampliar los estudios a dicho caso.

Por tanto, el presente trabajo coadyuvará en la gestión municipal de la Sub Alcaldía San Antonio, a través de la construcción, desarrollo e implementación de un *Sistema de Información que realice un Control y Seguimiento de los Procesos Técnicos y Administrativos de Fiscalización*.

1.2 ANTECEDENTES

Los sistemas que competen un Control de las operaciones y procesos de cualquier institución, competen también el Seguimiento de dichos procesos y operaciones, de esta manera, un Sistema de Control es a la vez un Sistema de Seguimiento.

Por esta razón, dichos sistemas son muy requeridos en los organismos públicos y privados y en la Carrera de Informática de la Universidad Mayor de San Andrés ya se elaboraron algunos sistemas semejantes como:

- [Acarapi F.,2005] Acarapi Choquehuanca Freddy, *“Sistema de Información para los Procesos de Registro, Control y Seguimiento de Actividades - caso Dirección de Coordinación Distrital G.M.L.P.”*
- [Cruz M.,2005] Cruz Ali Mariela H., *“Sistema de Control y Seguimiento de Activos Fijos para la Prefectura del departamento de La Paz”*
- [Tapia J.,2006] Tapia Selaez Javier F., *“Sistema de Información para el Seguimiento y Control de Procesos de Fiscalización Tributaria de Bienes Inmuebles”*
- [Chambi B.,2005] Chambi Aguilar Brigida D., *“Sistema de Seguimiento y Control de Correspondencia para la Prefectura del departamento de La Paz”*

1.2.1 MARCO INSTITUCIONAL

El Gobierno Municipal de La Paz (GMLP), funciona como un organismo planificador y ejecutor de actividades administrativas, técnicas y financieras, siempre tratando de incurrir en los ideales del buen trabajador: la eficiencia y transparencia. Dicha labor va en beneficio de la población, y por supuesto, del medio ambiente.

De esta manera, la distribución de tareas se ha repartido estratégicamente entre cinco Sub Alcaldías, que corresponden a los cinco Macro Distritos de la ciudad de La Paz. Cada Sub Alcaldía se constituye en un nivel de Delegación Operativa desconcentrada del GMLP, y tienen como función administrar y controlar los recursos financieros del Macro Distrito correspondiente, formulando políticas y estrategias para optimizar sus objetivos: la administración y captación de recursos para el mejoramiento predial.

En la misma línea, este es el caso de la Sub Alcaldía San Antonio, encargada íntegramente del crecimiento y mejoramiento predial y vecinal del Macro Distrito 8 (M8).

Dicha Sub Alcaldía, para su mejor funcionamiento ha dividido su extenso trabajo en Unidades de Servicio, una de estas unidades, la que nos interesa como caso de estudio es la Unidad de Fiscalización Integral.

La Unidad de Fiscalización Integral tiene el objeto de establecer mecanismos para controlar el correcto cumplimiento de las normas de fiscalización referidas al uso del suelo urbano en los Distritos de su dependencia, todo esto mediante la aplicación de las normas legales vigentes y si fuera necesario, de la fuerza pública.

1.2.2 DESCRIPCIÓN DE PROCEDIMIENTOS Y FUNCIONES

En la Unidad de Fiscalización Integral de la Sub Alcaldía San Antonio, se siguen ciertos procedimientos para alcanzar los objetivos de la misma. Además, para realizar su trabajo, los mismos mantienen una jerarquía organizacional, basada en los siete funcionarios que actualmente integran esta unidad. Hay que destacar que la labor esencial de dicha unidad se sostiene sobre los *inspectores prediales*, son estos funcionarios quienes ejecutan la parte dinámica en general. Pero para lograr una mayor comprensión de lo expuesto, a seguir se muestra el organigrama de la Unidad de Fiscalización Integral: Ver *Figura 1.1*



Figura 1.1 Organigrama de la Unidad de Fiscalización Integral

Fuente: [Elaboración Propia]

A continuación, se tiene un diagrama que explica y describe los procedimientos y las actividades que se realizan en la Unidad de Fiscalización: *Figura 1.2*

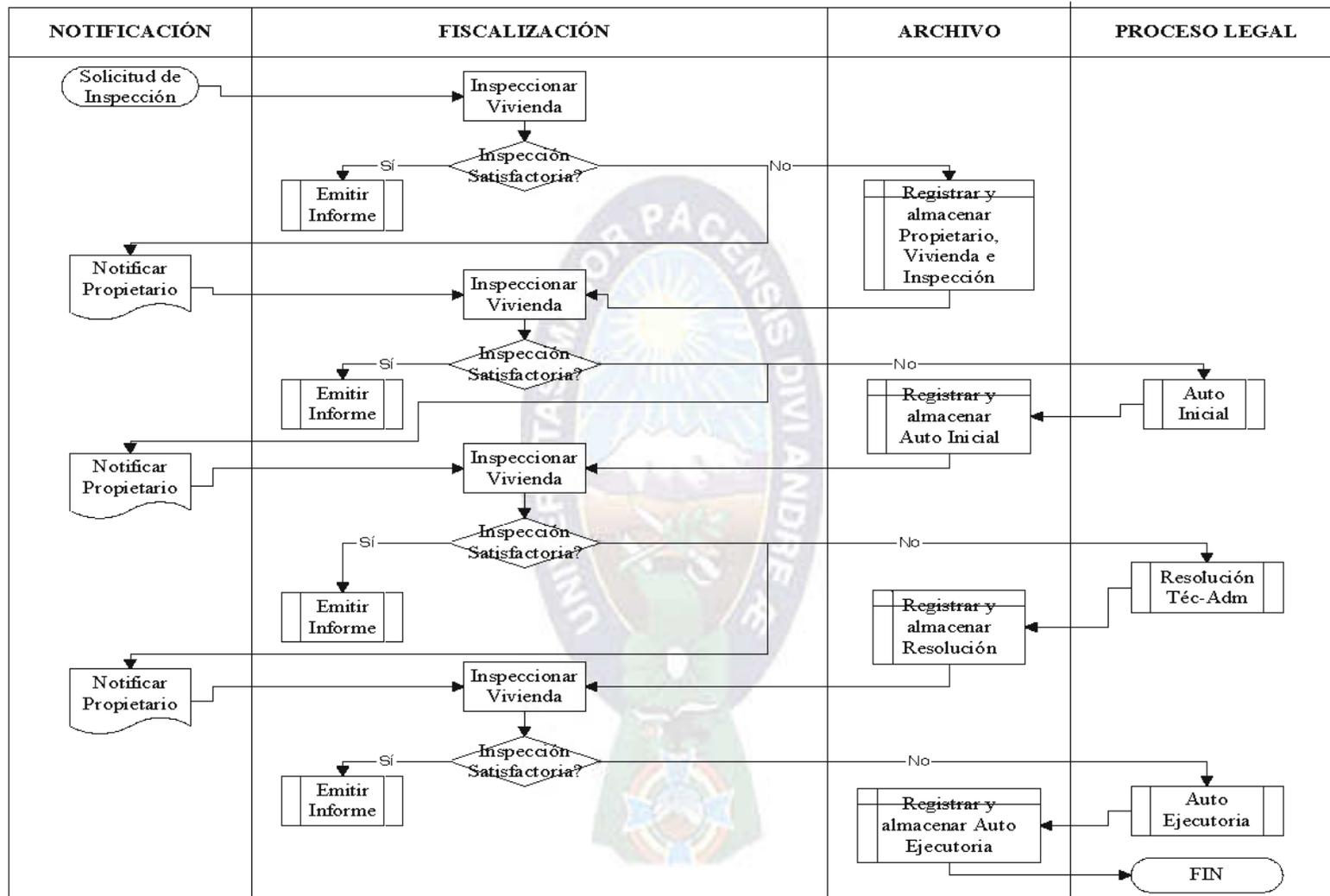


Figura 1.2 Diagrama de los Procedimientos de la Unidad de Fiscalización

Fuente: [Elaboración Propia]

1.3 IDENTIFICACIÓN Y FORMULACIÓN DEL PROBLEMA

La Unidad de Fiscalización Integral de la Sub Alcaldía San Antonio del Gobierno Municipal de La Paz (GMLP) es un organismo público con carácter municipal. Las principales tareas de la Unidad de Fiscalización implican un control exhaustivo de las construcciones prediales, así como el seguimiento de las mismas, mediante inspecciones diarias, las que deben ser registradas y almacenadas en una Base de Datos.

Estas inspecciones logran identificar infracciones a las leyes convenidas para la autorización de construcciones urbanas, dichas inspecciones se registran en forma de informes, los cuales se almacenan en una Base de Datos realizada en el programa Microsoft Excel, de esta forma, la cantidad de datos que se hallan en una misma pantalla a la hora de registrar nuevos datos y revisarlos después resulta ser una labor morosa, aburrida y fatigosa.

En la misma línea, después del manejo de la información y los procedimientos ejecutados, los reportes que sujetan los resultados no contienen un resumen provechoso, la presentación de estos reportes no son lo bastante claros para los funcionarios.

Por otro lado, la inserción de los datos al registro no involucran demasiada confianza, ya que cualquier funcionario de la Unidad puede realizarla, de esta misma forma, cualquier funcionario puede modificarlos y hasta eliminarlos.

Nos encontramos frente a problemas relacionados con el registro, almacenamiento, manejo y seguridad de la información puesto que a la vista tenemos cantidades de información aglomerada, de modo que la manipulación de la misma sea ineficiente y morosa, es mas, vemos que la integridad de la información se encuentra subordinada a la buena voluntad y consciencia de los funcionarios, esto con respecto a la seguridad de la información.

Por lo tanto, se implica e identifica el problema principal: **existe una inadecuada disponibilidad de la información y una escasa generación de reportes de control y seguimiento de la fiscalización predial.**

➤ LISTA DE PROBLEMAS

1. La actual base de datos es un solo archivo sin distinción de relaciones entre los datos de la información, que se rige por una lógica secuencial de procedimientos.
2. Resulta moroso recurrir a la lista extensa de los detalles de la base de datos actual.
3. La información no se halla disponible a todos los funcionarios.
4. Los actuales procesos de control y seguimiento de información declinan en un pesado y desordenado manejo de la información.
5. Existe dependencia funcionaria en la manipulación de información, es decir, solo el Jefe de la Unidad puede administrarlos.
6. La información se encuentra a disposición de la destreza de los funcionarios, es decir, que por cualquier descuido se pueden modificar y hasta eliminar los datos.
7. La base de datos se encuentra resguardada por una contraseña que puede ser transgredida por los modernos programas actuales.
8. No existe un generador de reportes estadísticos.

1.4 OBJETIVOS

1.4.1 OBJETIVO GENERAL

Desarrollar e implementar un Sistema de Información para el Registro, Control y Seguimiento de los procesos de fiscalización predial de la Unidad de Fiscalización Integral de la Sub Alcaldía San Antonio GMLP, de manera que se logre la optimización de estos procesos, orientados a mejorar la eficiencia de la gestión municipal.

1.4.2 OBJETIVOS ESPECÍFICOS

- Establecer una mejor forma de registrar y archivar la información.
- Determinar nuevos procesos que mejoren el control y seguimiento de la información.
- Implementar un nuevo mecanismo de seguridad para el sistema.
- Generar reportes estadísticos apropiados, de manera que la información obtenida sea útil para la toma de decisiones.

1.5 JUSTIFICACIÓN

La puesta en marcha de este Sistema gozará de una bienvenida particular, tanto social como tecnológica, por un lado los funcionarios de la Unidad de Fiscalización de la Sub Alcaldía San Antonio organizaran de mejor forma los procesos y optimizaran los mismos, por otro lado, los clientes que vienen a ser los vecinos de este Macro Distrito se beneficiaran indirectamente, pues sus trámites serán atendidos con mayor brevedad y puntualidad.

Por esto mismo, el Sistema de Control y Seguimiento de los Procesos de Fiscalización Integral de la Unidad de Fiscalización de la Sub Alcaldía San Antonio será desarrollado para mejorar la calidad de vida de los funcionarios y los vecinos.

1.6 ALCANCES

El alcance del proyecto contempla el análisis, diseño e implementación de un Sistema de Información.

En lo que se refiere al Análisis y Diseño del sistema, se delinearán los nuevos procesos de control y seguimiento de fiscalización, de manera que exista una optimización de los mismos.

Para la Implementación del Sistema de Control y Seguimiento de los Procesos de la UFI se desarrollaran los siguientes módulos:

- Modulo de Registro de Información
- Modulo de Verificación de la Información
- Modulo de Actualización de Información
- Modulo de Consultas o peticiones
- Control de los Procesos de Fiscalización
- Seguimiento de los Procesos de Fiscalización
- Modulo de Generación de Reportes

1.7 APORTES

Dentro del análisis del sistema, se pudo detectar lo siguiente: la información que toda institución municipal utiliza, debe ser resumida de manera práctica, por ende la Generación de los Reportes estarán orientados a las derivaciones estadísticas.

Esto quiere decir que, se tomaran en cuenta los gráficos de estadística, como las tortas, los histogramas y otros; además de ciertas formulas estadísticas.

1.8 METODOLOGÍA

En lo que se refiere a la investigación y recolección de datos se utilizará las técnicas de observación de los procedimientos, entrevistas y cuestionarios a los funcionarios, además de la investigación de libros y sitios Web.

Para el desarrollo del software se utilizará la metodología que contempla el “Proceso Unificado Racional (RUP)” puesto que se sostiene sobre las fases de análisis, diseño, construcción e implementación del sistema, además brinda un panorama de modelado visual apropiado para sistemas de estrategia estadística.

1.9 TÉCNICAS Y HERRAMIENTAS

Para una mejor proyección del sistema, se hará uso de una herramienta visual *orientada a objetos* utilizando de esta forma el “Lenguaje Unificado de Modelado (UML)”

Dentro de las herramientas para el desarrollo del software, se tomara en cuenta el uso del HTML como Editor de Interfaces Web, además del Lenguaje de Alto Nivel PHP para operar las consultas del sistema. Así también se trabajará con un Gestor de Bases de Datos como el My SQL para realizar las consultas a dicha base de datos.

La plataforma que sostendrá al Sistema será Windows 2000 y XP.



CAPÍTULO II

MARCO TEÓRICO

2.1 ANÁLISIS Y DISEÑO ORIENTADA A OBJETOS

El objetivo del análisis orientado a objetos es desarrollar una serie de modelos que describan el software de computadora al trabajar, para satisfacer un conjunto de requisitos definidos por el cliente. El diseño orientado a objetos transforma el modelo de análisis en un modelo de diseño que sirve como anteproyecto para la construcción del software, a diferencia de los métodos convencionales. [PRESSMAN, 2002]

El análisis y diseño orientado a objetos resulta ser la mejor forma para desarrollar sistemas enfocados en procesos y flujos de trabajo, esto quiere decir, que con la utilización de objetos es posible una mejor abstracción del mundo real, o mejor dicho, de los requisitos del sistema, todo esto gracias a otra ventaja de el diseño orientado a objetos, la modularidad.

2.2 EL MODELO UML

El Modelo UML es un lenguaje estándar OMG (Object Management Group) diseñado para visualizar, especificar, construir y documentar software orientado a objetos.

Un **Modelo** es una simplificación de la realidad. [BERZAL, 2001]

El Modelado es esencial en la construcción de software, pues permite:

- Comunicar la estructura de un sistema complejo.
- Especificar el comportamiento deseado del sistema.
- Comprender mejor lo que estamos construyendo.
- Descubrir oportunidades de simplificación y reutilización.

Un Modelo proporciona “los planos” de un sistema y puede ser más o menos detallado, en función de los elementos que sean relevantes en cada momento. Por esto se dice que el modelo ha de capturar *lo esencial*. [Berzal, 2001]

Entonces podemos decir, que UML es un lenguaje de propósito general para el modelado orientado a objetos. Impulsado por el Object Management Group (OMG), es una consolidación de muchas de las notaciones y conceptos más usados para objetos. Empezó como una consolidación del trabajo de Grade Booch, James Rumbaugh, e Ivar Jacobson, creadores de tres de las metodologías orientadas a objetos más populares.

El Lenguaje Unificado de Modelado preescribe 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 ejemplo, los símbolos o iconos característicos utilizados para capturarlos requisitos son: una elipse para representar un caso de uso y un monigote para representar un usuario que utiliza el caso de uso.

De forma similar, el icono principal utilizado en diseño es un rectángulo para representar una clase. Estos iconos no son más que una notación gráfica, es decir, una sintaxis, y tras esta notación UML específica una semántica (significado).

A continuación, se presenta una vista panorámica y esencial de la estructura del Lenguaje Unificado de Modelado.

2.2.1 Vocabulario de UML

UML provee a los desarrolladores un vocabulario que incluye tres categorías: **Elementos, Relaciones y Diagramas**, constituyen entonces, la estructura básica del lenguaje. [JACOBSON, BOOCH, RUMBAUGH, 2000]

Dentro de la categoría **Elementos**, hay cuatro tipos, que a su vez se subdividen en subtipos:

- Estructurales: *Casos de Uso, Clases, Clases Activas, Interfaces, Componentes, Colaboraciones y Nodos.*
- De Comportamiento: *Interacciones y Maquinas de Estado.*
- De Agrupación: *Paquetes, Modelos, Subsistemas y Marcos de Trabajo.*
- De Anotación: *Notas.*

Para la categoría **Relaciones**, existen tres tipos:

- De Dependencia
- De Asociación
- De Generalización

Y la tercera categoría, la de **Diagramas**, contempla nueve tipos:

- Diagramas de Casos de Uso
- Diagramas de Clases
- Diagramas de Objetos
- Diagramas de Secuencia
- Diagramas de Colaboración
- Diagramas de Estados
- Diagramas de Actividad
- Diagramas de Componentes
- Diagramas de Despliegue

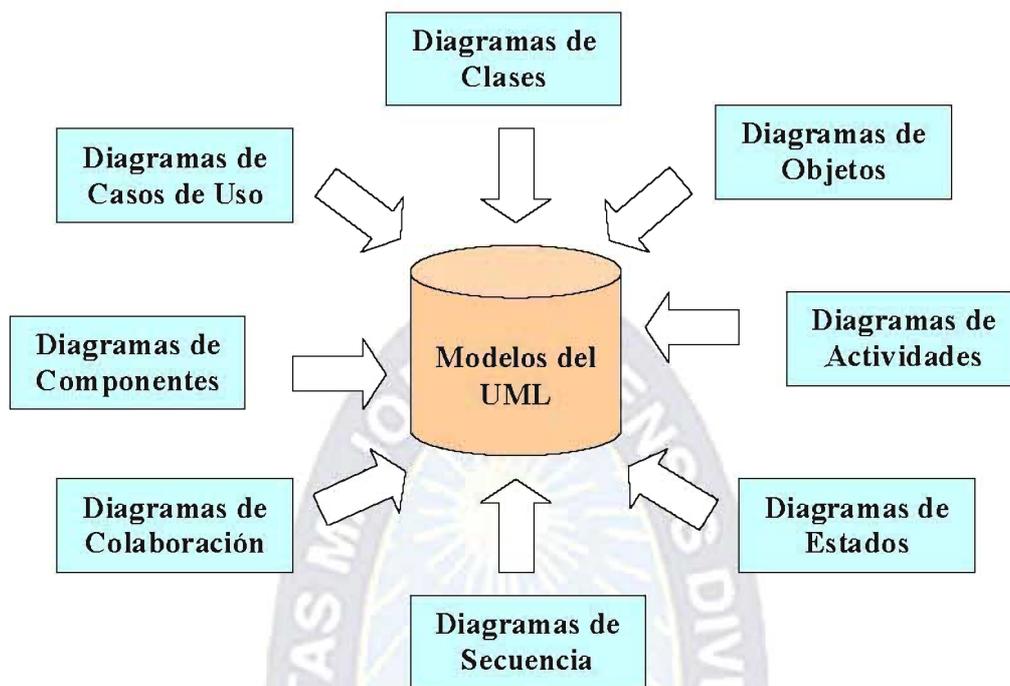


Figura 2.1 Diagramas de UML

Fuente: [JACOBSON, BOOCH, RUMBAUGH, 2000]

Al respecto, solo se mencionarán brevemente aquellos mecanismos que se aplicaran durante el desarrollo del sistema.

a) Diagramas de Casos de Uso

Los *diagramas de casos de uso* se utilizan en el modelado del sistema desde el punto de vista de los usuarios para representar las acciones que realiza cada tipo de usuario. Representa la funcionalidad que ofrece el sistema en lo que se refiere a su interacción externa. [LARMAN, 1999]

b) Diagramas de Clases

Los *diagramas de clases* son los diagramas principales para el análisis y diseño del sistema. Un diagrama de clases presenta las clases del sistema con sus relaciones

estructurales y de herencia. La definición de clase incluye definiciones para atributos y operaciones. [LARMAN, 1999]

Estos diagramas proporcionan una perspectiva estática del sistema, pues representan su diseño estructural.

c) Diagramas de Actividades

Los *diagramas de actividades* sirven fundamentalmente para modelar el flujo de control entre actividades. Los diagramas de actividades muestran el orden en el que se van realizando tareas dentro de un sistema.

La idea es generar una especie de diagrama Pert, en el que se puede ver el flujo de actividades que tienen lugar a lo largo del tiempo, así como las tareas concurrentes que pueden realizarse a la vez. Gráficamente un diagrama de actividades será un conjunto de arcos y nodos. [LARMAN, 1999]

d) Diagramas de Estado

Los *diagramas de estado* son especialmente importantes para describir el comportamiento de un sistema reactivo, el cual está dirigido por eventos.

Dichos diagramas muestran 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.

En cuanto a la representación, un diagrama de estados es un grafo cuyos nodos son estados y cuyos arcos dirigidos son transiciones etiquetadas con los nombres de los eventos.

e) Diagramas de Colaboración

Los *diagramas de colaboración* pertenecen a la clasificación de los Diagramas de Interacción, esto por que suelen mostrar patrones de interacción entre objetos.

Un diagrama de colaboración resalta la organización estructural de los objetos que intercambian mensajes. La distribución de los objetos en el diagrama permite observar adecuadamente la interacción de un objeto con respecto de los demás.

La perspectiva estática del sistema viene dada por las relaciones existentes entre los objetos. La vista dinámica de la interacción viene indicada por el envío de mensajes a través de los enlaces existentes entre los objetos. Y los mensajes se numeran para ilustrar el orden en que se emiten.

2.2.2 Mecanismos de Extensibilidad del UML

Los mecanismos de extensibilidad permiten a los desarrolladores refinar la sintaxis y semántica del UML, de forma, que se ajuste a un sistema, proyecto o proceso de desarrollo específico si es necesario. Estos mecanismos incluyen **Estereotipos, Valores Etiquetados y Restricciones**. [JACOBSON, BOOCH, RUMBAUGH, 2000]

Los Estereotipos proporcionan una forma de definir nuevos elementos extendiendo y refinando la semántica de elementos como elementos y relaciones ya existentes.

Los Valores Etiquetados suministran una forma de definir nuevas propiedades de elementos ya existentes.

Finalmente, las Restricciones plantean una forma de imponer reglas (como reglas de consistencia o reglas de negocio) sobre los elementos y sus propiedades.

El Proceso Unificado utiliza una notación para realizar los diagramas anteriormente descritos, se llaman *estereotipos*, por esta razón los graficaremos brevemente a continuación:

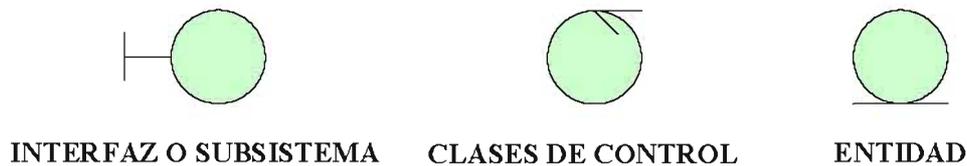


Figura 2.2 Notación de los Estereotipos de UML

Fuente: [JACOBSON, BOOCH, RUMBAUGH, 2000]

2.3 EL PROCESO UNIFICADO DE DESARROLLO DEL SOFTWARE

Al realizar un proyecto, llega un momento en el que se debe decidir por qué metodología inclinarse, ésta debe adaptarse a los requisitos y necesidades de los usuarios, de manera que sirva de guía para desarrollar una aplicación de software.

Es aquí donde entra el “Proceso Unificado de Desarrollo de Software”, mas conocido como “Proceso Unificado Racional” (Rational Unified Process, RUP), es una Metodología Orientada a Objetos.

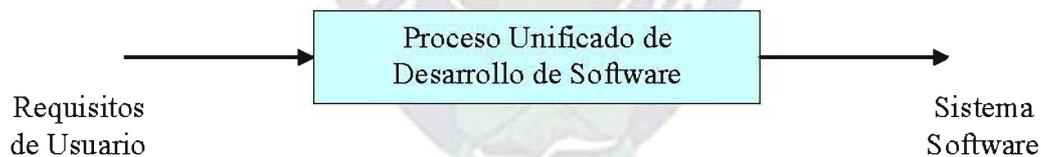


Figura 2.3 Un Proceso de Desarrollo de Software

Fuente: [JACOBSON, BOOCH, RUMBAUGH, 2000]

Un *proceso* de desarrollo del software es el conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema software. [JACOBSON, BOOCH, RUMBAUGH, 2000]. El Proceso Unificado utiliza el “Lenguaje Unificado de Modelado” (Unified Modeling Language, UML) para preparar todos los esquemas de un sistema software. De hecho, UML es una parte esencial del Proceso Unificado.

De esta manera, los aspectos definitorios del Proceso Unificado se resumen en tres principios clave:

a) Dirigido por Casos de Uso

El Proceso Unificado está “dirigido” por Casos de Uso, por que el proceso de desarrollo avanza a través de una serie de flujos de trabajo que parten de los casos de uso, esto quiere decir que existen ventajas en la utilización de los casos de uso:

- Proporcionan un medio sistemático e intuitivo de capturar **requisitos funcionales**, centrándose en el valor añadido para el usuario.
- Además, dirigen todo el proceso de desarrollo debido a que la mayoría de las actividades como el análisis, diseño y prueba se llevan a cabo partiendo de los casos de uso.

b) Centrado en la Arquitectura

Esta metodología esta centrada en la arquitectura del sistema software por que permite describir el sistema software mediante diferentes vistas de diseño del sistema en desarrollo.

Esto quiere decir, que resulta útil representar el sistema desde diferentes perspectivas para la mejor comprensión del diseño. La arquitectura software abarca decisiones importantes sobre:

- La organización del sistema software.
- Los elementos estructurales que constituirán el sistema y sus interfaces, junto con sus comportamientos, tal y como se especifican en las colaboraciones entre estos elementos.

- La composición de los elementos estructurales y de su comportamiento en el sistema.
- El *estilo de la arquitectura* que guía esta organización: elementos y sus interfaces, colaboraciones y su composición.

Sin embargo, la arquitectura software no solo obedece a la estructura y el comportamiento, sino también al uso, la funcionalidad, el rendimiento, la flexibilidad, la reutilización, la facilidad de comprensión, las restricciones, la estética y los compromisos tecnológicos.

Como ya se dijo, la arquitectura del software se representa mediante **vistas del modelo**, básicamente una vista de modelo es un extracto o una parte de ese modelo, por ejemplo, una vista del modelo de casos de uso, se parece al propio modelo de casos de uso, pues tiene actores y casos de uso, pero solamente aquellos que son arquitectónicamente significativos.

c) Iterativo e Incremental

Todo Proceso de desarrollo de Software se auto-evalúa acatando criterios de evaluación, valga la redundancia, que sirven de toma de decisiones para pasar a la siguiente fase del desarrollo. Dentro de cada fase se abordan una serie de iteraciones e incrementos que conducen a esos criterios, en el sentido que propone un desarrollo de software por partes, de este modo, las iteraciones son controladores y hacen referencia a pasos en el flujo de trabajo, y los incrementos, al crecimiento del producto.

El Proceso Unificado es una metodología cíclica, cada ciclo consta de cuatro fases evolutivas asociadas al tiempo, y en cada fase se encuentra una secuencia de actividades a desarrollar que también siguen una curva cíclica asociados, esta vez, al contenido.

Por esta razón, el Proceso Unificado es descrito en dos dimensiones: tiempo (fases) y contenido (flujos de trabajo). Las Fases: Inicio, Elaboración, Construcción y Transición. Los Flujos de Trabajo: Requisitos, Análisis, Diseño, Implementación y Prueba.

La *Figura 2.2* mostrará la descripción del Proceso Unificado en dos dimensiones.

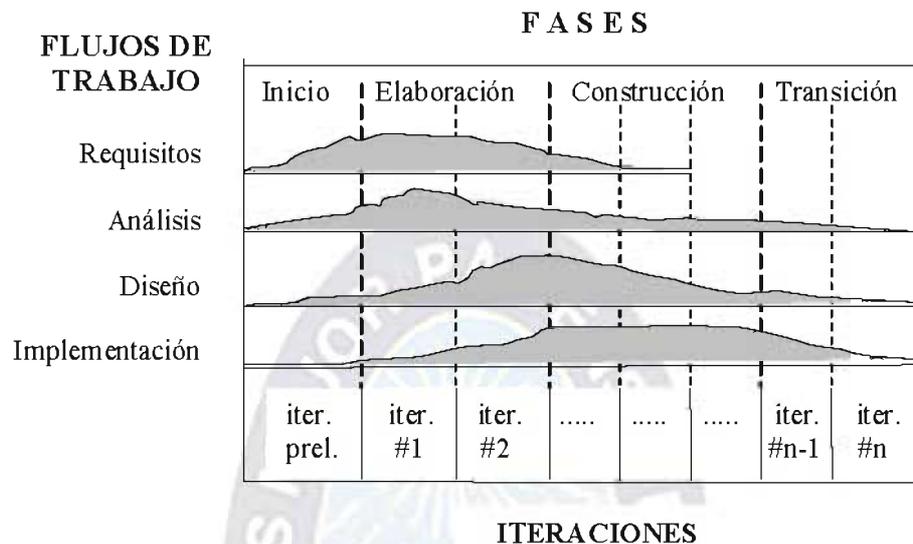


Figura 2.4 Flujos de Trabajo VS Fases

Fuente: [JACOBSON, BOOCH, RUMBAUGH, 2000]

2.4 FASES Y FLUJOS DE TRABAJO DEL DESARROLLO DEL SOFTWARE

Dentro del desarrollo del software esta inmerso el ciclo de vida del mismo, éste recorre cuatro fases a través de una secuencia de modelos que permiten visualizar lo que sucede en cada fase. Ver *Figura 2.2*.

Cada fase termina con un hito, es decir, con un punto en el cual deben tomarse importantes decisiones a partir de ciertos modelos y documentos que han sido desarrollados para alcanzar un estado predefinido.

El RUP para su mejor comprensión proporcionado un conjunto de modelos cuidadosamente seleccionados, cada uno de estos modelos vienen a ser el producto de cada fase. Cada modelo brinda una vista auto-contenida del sistema. Este conjunto de modelos

clarifica el desarrollo del sistema para todos los interesados, incluyendo usuarios, clientes y desarrolladores. Ver Figura 2.3

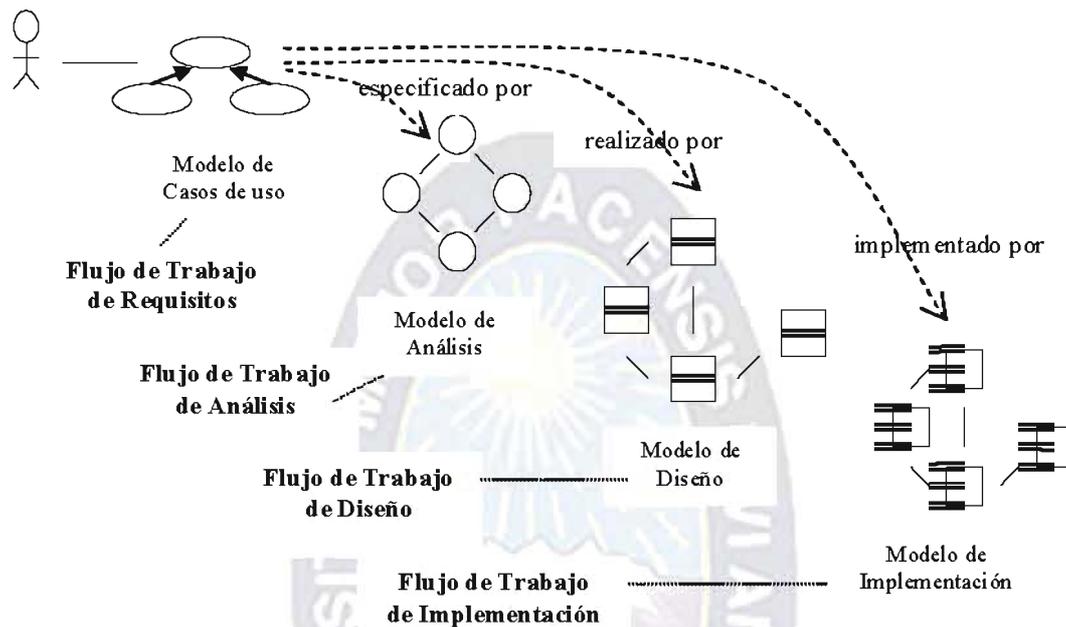


Figura 2.5 Modelos y Flujos de Trabajo

Fuente: [JACOBSON, BOOCH, RUMBAUGH, 2000]

2.4.1 FASE DE INICIO - REQUISITOS

En esta fase se desarrolla una descripción del producto final a partir de una idea y se presenta el análisis del sistema para el producto. [JACOBSON, BOOCH, RUMBAUGH, 2000]

El propósito fundamental del flujo de trabajo de los Requisitos es guiar el desarrollo hacia el sistema correcto. Un **requisito** es una descripción de necesidades o aspiraciones respecto a un producto. El objetivo principal de la actividad de definición de requisitos consiste en identificar qué es lo que realmente se necesita, separar el grano de la paja. Esto se hace en un modo que sirva de comunicación entre el cliente y el equipo de desarrollo.

a) Lista de Características de los Requisitos

Es una lista enumerada que surge de la experiencia de empleados, usuarios y clientes con sistemas anteriormente ya vistos y utilizados, además claro, de las nuevas expectativas para el sistema propuesto.

b) Modelo del Negocio

Describe los procesos existentes u observados con el objetivo de comprenderlos, estos procesos serán los que soportará el sistema e implican las competencias requeridas en cada proceso.

Un Modelo del Negocio describe los procesos del negocio, un proceso se describe en términos de flujos de trabajo, donde un flujo de trabajo es un conjunto de actividades, una actividad, a su vez, es un fragmento de trabajo realizado por un actor en el flujo de trabajo, es decir, es la ejecución de una de las operaciones de los actores. Por tanto, un Modelo del Negocio se detalla mediante los *Diagramas de Actividades*.

En la misma línea, existe una actividad a realizar para coadyuvar a la comprensión del Modelo del Negocio, ésta se llama:

c) Identificación de los Procesos

Dado que se conoce que el Modelo de Negocio describe los procesos contextuales del sistema estudiado, es preciso determinar y enumerar los procesos que se van modelar, por esto, es preciso realizar una lista de los procesos identificados y visualizados en un diagrama de actividades.

d) Actores y Casos de Uso

- **Casos de Uso de Alto Nivel:** Encontrar actores y casos de uso, los importantes, aquellos que estén relacionados con el ámbito del sistema. Cada caso de uso representa una forma de usar el sistema, de dar soporte a un usuario durante un proceso de negocio.
- **Casos de Uso Expandido:** Permite una mejor comprensión de la relación que existe entre el sistema y su entorno, un caso de uso expandido muestra detalles específicos de dicha interacción.

e) Diagrama de Casos de Uso

Describe gráficamente las condiciones y posibilidades que debe cumplir el sistema, contiene actores, casos de uso y sus relaciones.

f) Prioridad de los Casos de Uso

Determinar la prioridad de los casos de uso, es clasificar el subconjunto de casos de uso necesarios para los objetivos del sistema, de manera que se describen los casos de uso, aquellos que son fundamentales para reconocer el contexto del sistema y los que son necesarios para planificar la arquitectura del software.

2.4.2 FASE DE ELABORACIÓN - ANÁLISIS

Durante la fase de elaboración se especifican en detalle la mayoría de los casos de uso del sistema y se diseña la línea base de la arquitectura del mismo.

El objetivo del Análisis es obtener una especificación precisa de los requisitos capturados en la anterior fase, utilizando un lenguaje formal, de este modo se facilita su

comprensión, su preparación, su modificación, y en general, su mantenimiento, para reconocer el funcionamiento interno del sistema.

a) Modelo Conceptual

Un modelo conceptual es una representación de las entidades del mundo real como objetos, que tienen su propia identidad y atributos que participan en las relaciones. Por esta razón, este modelo proporciona un medio para asociar los significados de las cosas de la realidad con las construcciones en el modelo. [HANSEN, 1997]

En UML se ilustra como un diagrama de estructura estática donde no se define ninguna operación. La designación del modelo conceptual ofrece la ventaja de subrayar fuertemente una concentración en los conceptos del dominio o contexto. [LARMAN, 1999]

b) Identificación de Clases y Descripción de Interacciones

Una clase es una descripción de un conjunto de objetos que comparten los mismos atributos, operaciones, relaciones y semántica; por tanto en este paso, se identifican las clases de los casos de uso, para luego refinarlas y asignarles responsabilidades.

Identificación de clases, para encontrar las clases de control, entidades e interfaces necesarias para realizar los casos de uso y esbozar sus nombres, responsabilidades, atributos y relaciones.

Y descripción de interacciones, después de esbozar las clases necesarias para efectuar los casos de uso, se debe describir cómo interactúan sus correspondientes objetos de análisis (actores, clases de control y entidades). Esto se ejecuta mediante los diagramas de colaboración.

c) Identificación de Métodos o Responsabilidades

Es decir, reconocer los métodos que seguirán las clases de análisis; ya sean clases de control y/o clases entidad. Es conveniente que los nombres de los métodos sigan una semántica verbal.

d) Identificación de Atributos

Se trata de identificar atributos, o propiedades de cada clase de análisis, es recomendable que los atributos sean sustantivos y conceptuales, esto se realiza solo para las clases entidad.

e) Identificación de Asociaciones

Las mínimas, pues resulta conveniente identificar las relaciones significativas para el siguiente paso. Estas asociaciones pueden ser singulares o múltiples, o una combinación de ambas.

2.4.3 FASE DE CONSTRUCCIÓN - DISEÑO

El objetivo de esta fase es la obtención del producto software, que estaría listo para la operación inicial del usuario final; cuya evolución iterativa e incremental será preparada a través de todos los modelos de casos de uso y escenarios restantes, para lograr la integración de los subsistemas.

En el Diseño modelamos el sistema y encontramos su forma, incluida la arquitectura, para que soporte todos los requisitos y mantenga las restricciones. Se profundiza en los aspectos relacionados con los lenguajes de programación, componentes reutilizables, sistemas operativos, tecnologías de interfaz de usuario, etc.

a) Organización de Paquetes de Elementos

Organizar los elementos en paquetes ofrece la ventaja de separar los elementos detallados en abstracciones más amplias, lo cual brinda soporte a una vista de nivel superior y permite contemplar el modelo en agrupamientos más simples.

b) Diseño de la Arquitectura del Sistema

Una arquitectura de sistemas de información es la arquitectura de tres capas, esta divide las transacciones en procesos independientes que cooperan entre si para intercambiar información, servicios o recursos.

La arquitectura de tres capas consiste en aislar la lógica de la aplicación, y convertirla en una capa intermedia bien definida y lógica del software.



Figura 2.6 Vista Clásica de una Arquitectura de Tres Capas

Fuente: [LOAYZA, 2004]

Los principales componentes de la arquitectura de tres capas son:

- **Presentación**

También llamado esquema externo, muestra al usuario el sistema mediante una interfaz Web de acceso universal con protocolos estándares universales.

- **Dominio de Aplicación**

En esta capa se especifica todo el trabajo que se debe realizar en el sistema y ciertas reglas que regirán los procesos. Para modelar estos procesos es conveniente partir de una cualidad del modelado orientado a objetos, la modularidad, reutilización y simplicidad.

- **Almacenamiento**

En esta última capa se realiza el almacenamiento persistente de los datos, lo que permite distinguir entre los conjuntos de datos por cada aplicación y los almacenes de datos orientados al manejo corporativo de estos.

c) Diseño Orientado a Objetos para Aplicaciones Web

Cuando se esta utilizando la web como vía de salida, existen ciertos requisitos navegacionales que son necesarios para el diseño de sistemas encauzados a una presentación web.

Una característica fundamental de las presentaciones o paginas web, son los enlaces, estos permiten al usuario recorrer, de pagina en pagina, la estructura del sitio web, o en este caso del sistema de información.

En la actualidad se han desarrollado tecnologías para el diseño de aplicaciones web, uno de estos es el Método de Diseño Hipermedia Orientado a Objetos (OOHDM, Object-Oriented Hypermedia Design Method) que presenta claridad y factibilidad.

Dicha metodología es utilizada por una de sus facetas, la identificación de tres diferentes formas de diseño independientes de la implementación: diseño conceptual, diseño navegacional (que se utilizara en este caso) y diseño de interfaces abstractas.

Esto quiere decir, que después del diseño de las clases, se diseñan los aspectos navegacionales del sistema, a través del esquema de clases navegacionales. De esta forma, el diagrama de clases sufrirá cambios, de manera que se mostrara un nuevo modelo que incluye nodos y enlaces, que permiten representar las posibilidades de navegación en el sistema.

- **Clase Nodo**

Una clase nodo es un tipo esencial de clase que muestra una agrupación de información bajo un determinado criterio []. Los atributos y operaciones de una clase nodo van a ser los mismos atributos y operaciones de clases del diagrama de clases. De forma que se puede contener atributos y operaciones de diferentes clases del diagrama de clases de la misma forma que una propiedad de una clase del diagrama de clases puede estar en diferentes nodos.

- **Enlace**

De esta misma forma, los enlaces definen un tipo, pues, mientras los nodos son propios de cada aplicación, la definición de la clase tipo enlace es genérica.

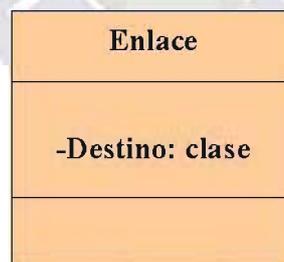


Figura 2.7 Definición de Clase Enlace

Fuente: [SILVA, MARCERAT, 2002]

El único atributo que tiene es el que indica la clase a la que se llega al ejecutar en enlace y que es del tipo parámetros en el que se instancia.

d) Modelo de Despliegue del Sistema

Es un modelo de objetos que describe la distribución física del sistema en términos de cómo se distribuye la funcionalidad entre los nodos de cómputo.

2.4.4 FASE DE TRANSICIÓN - IMPLEMENTACIÓN

La fase de transición cubre el periodo en el que el software es entregado y ‘probado’ por los usuarios, de esta forma los defectos y deficiencias pueden ser corregidos, así como algunas últimas características pospuestas pueden ser cubiertos.

En la implementación empezamos con el resultado del diseño e implementamos las clases del sistema en términos de componentes, es decir, ficheros de códigos fuente, ficheros de códigos binarios, ejecutables y similares. Lo importante de esta fase es la prueba individual de los componentes para luego integrar el sistema.

a) Modelo de Implementación del Sistema

Describe cómo los elementos del modelo del diseño, como las clases, se implementan en términos de componentes, como ficheros de código fuente, ejecutables, etc. tomando en cuenta el entorno de implementación del lenguaje de programación utilizado y cómo dependen los componentes unos de otros.

b) Diseño e Implementación de la Base de Datos Relacional

El diseño y construcción de la Base de Datos Relacional para un sistema orientado a objetos implica una traslación, pues los objetos no pueden ser diseñados directamente como un conjunto de entidades relacionales, ya que los objetos tienen identidad, estados y comportamiento que no son soportados por la estructura de datos estática.

Dicha traslación se realiza mediante una técnica que manipule el diseño de las clases de objetos, hacia la construcción de un conjunto de entidades con las semejanzas y diferencias que se encuentran en los diagramas de clases y objetos.

De esta manera, las entidades relacionales usan objetos creando referencias directas, usando llaves primarias y extranjeras.

Mientras tenemos que considerar lo siguiente:

- El modelo relacional de datos busca una eficaz forma de presentar la relación de los datos usando como técnica la normalización de los datos, esto lo hace eliminando la redundancia que existe entre los datos de las tablas, para luego obtener una rápida recuperación de los datos de la BD.
- El diseño orientado a objetos plantea modelar y representar lo mas semejante posible el mundo real, mediante la creación de objetos que contienen los datos y su comportamiento.

Para realizar este paso de traslación, UML brinda una interrelación entre el Modelado de Datos y el Modelo Conceptual, esto se hace cuando se define la estructura de la base de datos, se define también el **modelo de datos**, que es el conjunto de herramientas conceptuales para describir datos, sus relaciones, su significado y sus restricciones de consistencia.

La característica principal de un modelo de datos es analizar los aspectos de interés para una organización y la relación que tiene con su entorno, describiendo los datos más importantes de sus principales actividades.

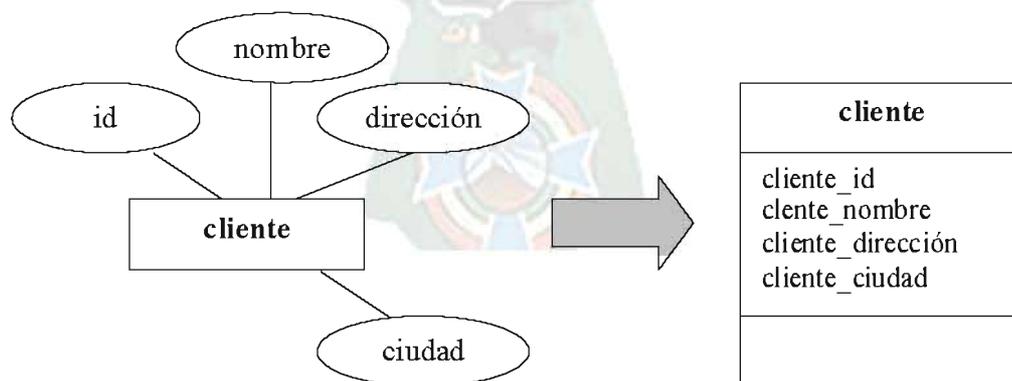
c) Traslación del Diagrama de Clases al Modelo Relacional

Los diagramas de clases del UML pueden modelar algunos aspectos del diseño de base de datos relacional, aunque no cubre toda la semántica involucrada en el modelo relacional.

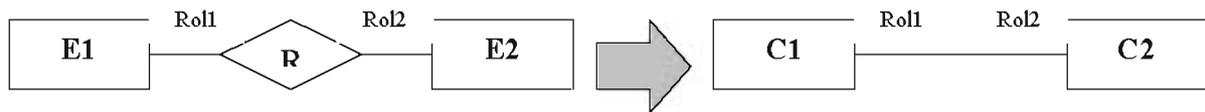
En esta misma línea, para la captura de la información, un diagrama Entidad/Relación se utilizara como un mecanismo de extensión del UML. Y para modelar la estructura lógica de la base de datos, independientemente de su condición orientada a objetos o relacional, se representara una clase como una tabla, y los atributos de clase como las columnas o campos de la tabla. Si una Base de Datos Relacional es el método de implementación escogido, entonces un diagrama de clases puede ser referenciado por un diagrama Entidad/Relación.

Entonces, las clases y sus atributos hacen referencia directamente a las entidades lógicas y sus atributos respectivamente. De esta forma, también se puede inferir las relaciones entre entidades e identificar que atributos son claves primarias y claves foráneas o extranjeras, conforme a las reglas de normalización de datos.

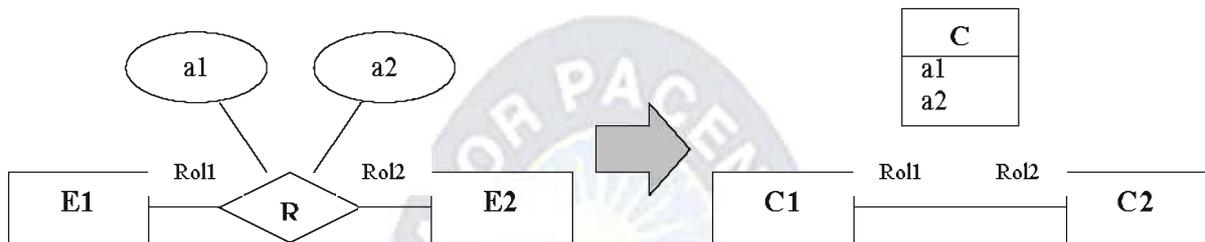
La siguiente figura graficará lo dicho anteriormente:



1) ENTIDADES Y ATRIBUTOS



2) RELACIONES



3) CARDINALIDAD ENTRE ENTIDADES

Figura 2.8 Traslación del Modelo Relacional al Modelo Orientado a Objetos

Fuente: [HANSEN, 1997]

2.5 ENTORNO DE SEGURIDAD

Basados en la información, las bases de datos son la parte significativa de todo sistema, sin ellas la mayoría de los productos desarrollados carecen de valor.

Al respecto, toca mencionar, que la protección de las bases de datos es una parte fundamental dentro del desarrollo del software. La protección de una base de datos también refleja ciertos atributos, por ejemplo:

- **Accesibilidad:** forma de acceder a la información concentrada en disco.
- **Operatividad:** grado de facilidad y comprensión de la manipulación de los datos.

Al respecto, los modernos gestores de bases de datos rescatan estos atributos, y, de esta forma, ya no es necesario diseñar y crear nuevas funciones y módulos en los sistemas para garantizar dichos atributos.

Tal es este caso, para el SGBD My SQL, que cuenta con autenticación de usuario y otorga diferentes niveles de privilegios a los distintos usuarios.

2.6 CALIDAD DEL SOFTWARE

Existen muchas definiciones sobre lo que puede y no puede significar la *calidad*, sin embargo, para los objetivos de estudio, se tomara en cuenta, la definición de Pressman Roger, que dice: *la calidad del software hace hincapié en la concordancia con los requisitos funcionales y de rendimiento explícitamente establecidos, los estándares de desarrollo explícitamente documentados y las características implícitas que se esperan de todo software desarrollado profesionalmente.*

La calidad del software es una compleja mezcla de actores que varían de acuerdo a la aplicación que se desarrolla y según a los requerimientos del cliente, por esto, las medidas para medirlo no son absolutas.

2.7 FACTORES DE CALIDAD ISO 9126

El estándar ISO 9126 identifica cinco atributos de calidad: [PRESSMAN, 2002]

a) **Confiabilidad**

Cantidad de tiempo que el software esta disponible para su uso. Se refiere a la tolerancia de fallos y a la recuperación de errores.

b) **Usabilidad**

Grado en que el software es fácil de usar, esta referido a la facilidad de comprensión, facilidad de aprendizaje y operatividad.

c) **Eficiencia**

Grado en el que el software hace óptimo el uso de los recursos del sistema, viene indicado por el tiempo de uso y los recursos utilizados.

d) Capacidad de Mantenimiento

Esta dada por la facilidad con la que una modificación puede ser realizada, y se refiere a la facilidad de análisis, facilidad de cambio y estabilidad.

e) Portabilidad

Dado por la facilidad con que el software puede ser llevado de un entorno a otro, se refleja en los siguientes atributos: facilidad de instalación, facilidad de ajuste, facilidad de adaptación al cambio.

2.8 CALIDAD DE LA APLICACIÓN WEB

El objetivo de medir la calidad de las aplicaciones web significa que se pueda saber a que nivel se logro una aplicación necesaria y suficiente de forma que cumpla con los requisitos de los usuarios del sistema.

Hoy en día, se están realizando varias investigaciones sobre metodologías capaces de validar los sitios web, en este punto, se tomara en cuenta la alternativa para el control de calidad de aplicaciones web presentada por Luis Antonio Olsina, llamada: *Metodología Cuantitativa de Evaluación y Comparación de la Calidad de Sitios Web*, conocida abreviadamente por Web-Site QEM. [OLSINA, 1999]

Dicha metodología es un modelo jerárquico de requerimientos de calidad, que toma las características de calidad del modelo ISO 9126, de estas características se derivan otras *sub-características* que se descomponen como un árbol jerárquico, los cuales se definen como *atributos*.

2.9 METODOLOGÍA DE EVALUACIÓN DE CALIDAD DE SITIOS WEB (WEB-SITE QEM)

La metodología Web-Site QEM tiene como meta principal la evaluación y comparación de calidad de artefactos web para comprender el grado de cumplimiento de un

conjunto de características y sub-características con respecto a los requerimientos de calidad establecidos. [OLSINA, 1999]

Web-Site QEM incluye una serie de fases, actividades, procesos y modelos; dentro de los modelos a los cuales se rige se puede mencionar el Modelo de Calidad de Mac Call, resumido y modificado, en la norma de calidad ISO 9126.

Así también sigue unas cuantas fases de procedimiento para medir la calidad de las aplicaciones web, estas fases no son absolutas, es decir, pueden modificarse dependiendo de la aplicación a evaluar.

De esta forma, para lograr la medición y comparación de calidad de una aplicación web, es preciso determinar los *Criterios Elementales e Implementar los Procedimientos de Medición*, para obtener un puntaje elemental de medición.

Definiendo Criterios Elementales e Implementando los Procedimientos de Medición

Un criterio de evaluación especifica cómo medir atributos cuantificables, además de facilitar su comprensión en términos de interpretación. El resultado final es una preferencia elemental, la cual puede ser interpretada como el grado o porcentaje del requerimiento elemental satisfecho.

Entonces decimos que, para cada variable de calidad medida:

$$X_i, \quad i=1,2,\dots,n \quad \text{donde } X_i \text{ es un atributo } i$$

Se define una función que representa el criterio elemental. Por definición un criterio elemental es una correspondencia del valor de la variable de calidad X_i en el valor de la preferencia o indicador de calidad elemental IE_i .

En términos generales, el valor medido de la variable es un número real:

$$X_i \in R_i \subset R$$

El valor de la preferencia de calidad elemental es también un número real pero perteneciente al intervalo unitario I, de manera que:

$$IE_i \in I, \quad i = 1, \dots, n \quad \text{con } I = [0,1]$$

Así, la preferencia elemental de calidad es el grado de verdad en la declaración que afirma “el valor de la variable de calidad X_i satisface completamente el requerimiento de calidad del i -ésimo criterio elemental”.

Con frecuencia, en vez de usar el intervalo unitario es útil emplear la escala porcentual de [0,100%]. En este sentido se interpreta a la preferencia como el porcentaje del requerimiento satisfecho para algún atributo X_i .

Entonces, si $IE_i = 0\%$ denota una situación totalmente insatisfactoria, $IE_i = 100\%$ representa una situación totalmente satisfactoria. Así el puntaje elemental cae en uno de los tres niveles de aceptabilidad:

NIVELES DE ACEPTABILIDAD	VALOR PORCENTUAL
Insatisfactorio	De 0 a 40%
Marginal	Desde 40 a 60%
Satisfactorio	Desde 60 a 100%

Tabla 2.1 Niveles de Aceptabilidad

Fuente: [OLSINA, 1999]

CAPÍTULO III

MARCO APLICATIVO

3.1 FASE DE INICIO - REQUISITOS

a) Lista de Características de los Requisitos del Sistema

Esta lista surge de la experiencia de los funcionarios de la Unidad de Fiscalización con el actual sistema de base de datos ejecutado en el paquete Microsoft Excel, y también de las nuevas expectativas para el sistema propuesto:

1. *Información Disponible*, para los funcionarios de la unidad en el tiempo en el que se pretende usarla.
2. *Integración de la Información*, de forma que ni intencionalmente, ni accidentalmente se la vulneré.
3. *Independencia de Datos*, para no confundir la información, de forma que se maneje información concreta y específica.
4. *Registro y Asignación de roles*, para facilitar las funciones de los empleados y su acceso de la información.
5. *Acceso a la Información*, y restricción de usuarios a determinadas funciones del nuevo sistema.
6. *Seguridad de la Información*, que se mantenga un control sobre la lectura de datos solo por funcionarios de la unidad.
7. *Control de Operativos*, ya sean de inspección como de fiscalización, ejecutados a partir de la información coleccionada en el sistema.

8. *Control y Seguimiento de los Procesos*, que sea adecuado para la organización y coordinación de actividades y operativos desde la notificación hasta el auto de ejecutoria.
9. *Reportes Estadísticos*, de los porcentajes de recaudaciones por infracciones de cada distrito, necesarios para la toma de decisiones.

Las características adicionales para los requisitos son:

10. *La interfaz de usuario*, está diseñado con un modelado web, por tanto goza de ventajas de accesibilidad, como la intuición y la manejabilidad, tanto para los funcionarios como para los clientes.
11. *El sistema seguro y accesible*, solo para los funcionarios autorizados de la unidad de fiscalización.
12. *El software del sistema es flexible*, para la aplicación de posteriores cambios y/o desarrollo de otros módulos.
13. *Aplicar el Modelo de Tres Capas*, para la arquitectura del sistema.

b) Modelo del Negocio de la Unidad de Fiscalización

El Modelo del Negocio de la Unidad de Fiscalización se representa mediante un Diagrama de Actividades, en el cual existen tres calles: Propietario, Fiscalización e Inspector, por las cuales las tareas de la unidad se distribuyen, de manera que muestre el orden de las mismas.

El diagrama de actividades que representa el Modelo del Negocio de los Procesos de la Unidad de Fiscalización se ve en la *Figura 3.1*

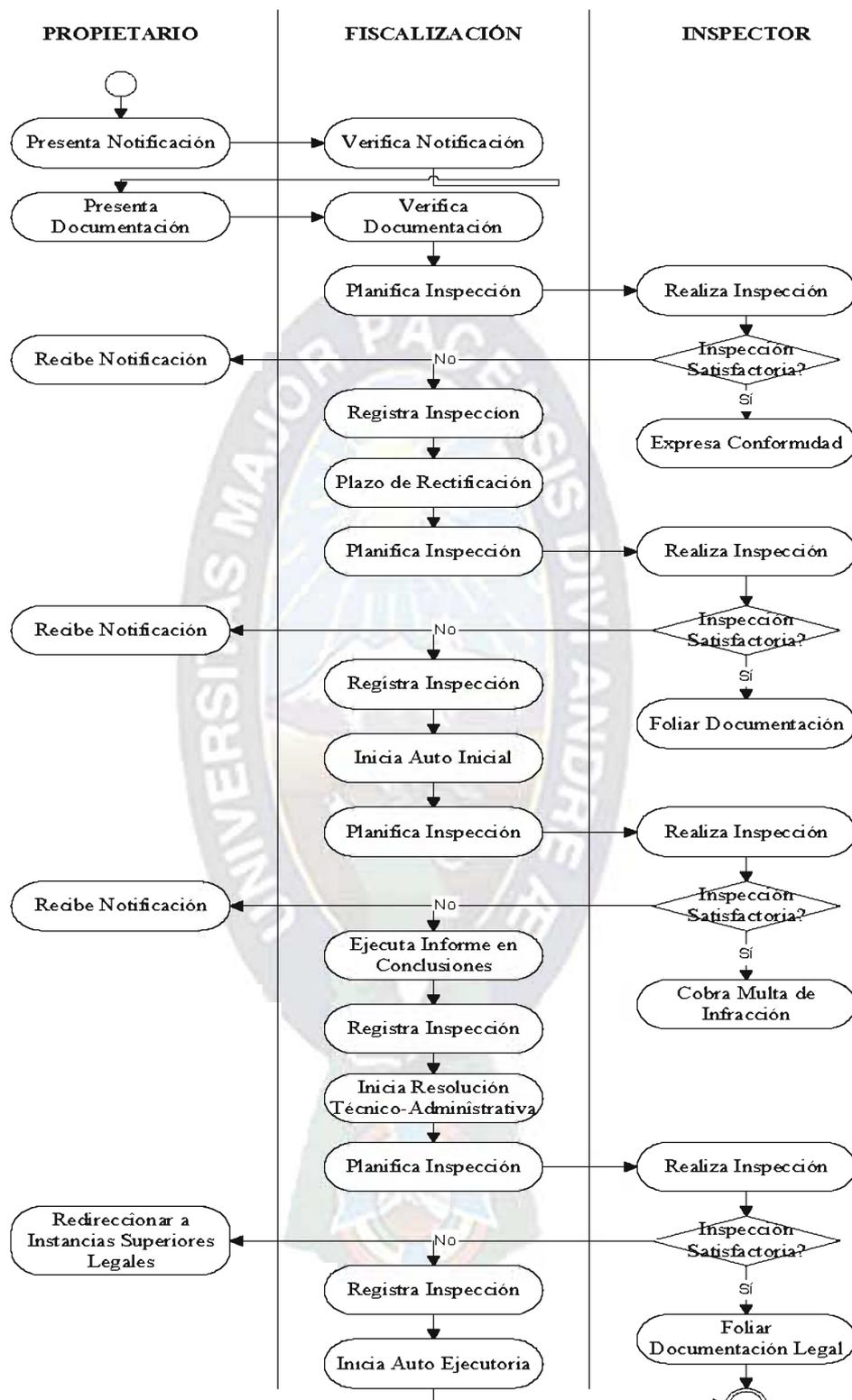


Figura 3.1 Modelo del Negocio – Procesos de la Unidad de Fiscalización

Fuente: [Elaboración Propia]

Luego es necesario identificar los procesos de fiscalización que se suceden en la Unidad de Fiscalización Integral, además también se los visualiza en otro diagrama de actividades. Ver *figura 3.2*

c) Identificar Procesos de Control y Seguimiento

- *Registro de la información:* Este proceso se encarga del almacenamiento de datos de los propietarios y sus viviendas o construcciones, de las inspecciones realizadas a las mismas, y, de los procesos iniciados, si es que hubieran.
- *Control de los operativos de inspección:* Se encarga de realizar una verificación continua del estado de las construcciones o viviendas, tomando en cuenta los datos relativos a cada caso, para luego concertar actividades de inspección en un periodo de tiempo.
- *Seguimiento de los procesos:* Se encomienda a este proceso el control continuo de pasos secuenciales que incluyen la revisión y comprobación de la información, referida a las inspecciones realizadas a las viviendas, y de los procesos legales iniciados a los propietarios, todo esto para llegar a los objetivos de la unidad.
- *Actualización de la información:* Con este proceso la información puede actualizarse mediante órdenes de modificación y eliminación, correspondientes al funcionario autorizado.
- *Búsquedas y vistas de la información:* En este proceso, la tarea implica fijar parámetros de búsqueda de los datos. Y además, las vistas son realizadas de forma apropiada para los funcionarios.

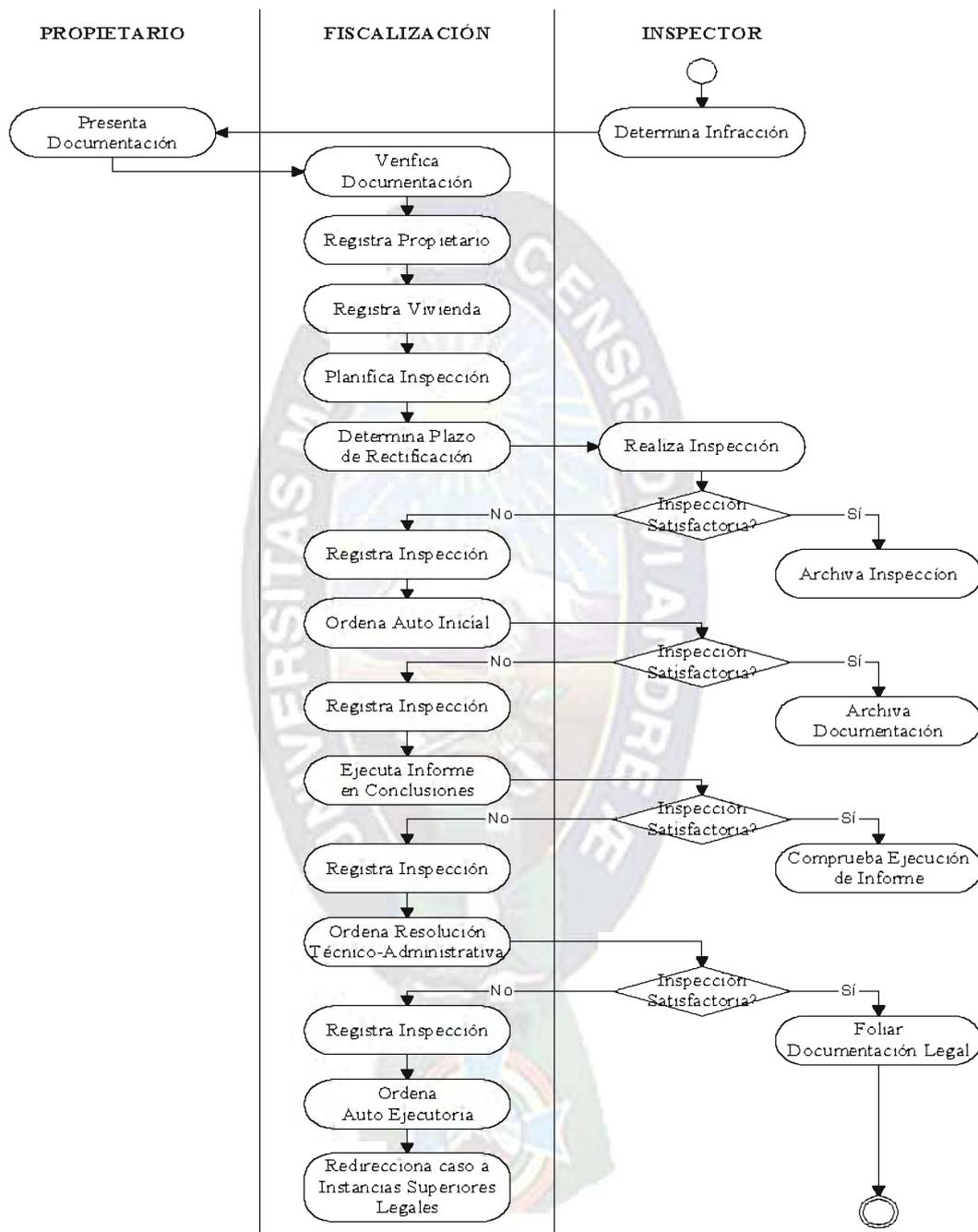


Figura 3.2 Diagrama de Actividades – Control y Seguimiento de Procesos

Fuente: [Elaboración Propia]

d) Actores y Casos de Uso de Alto Nivel

Esencialmente, se reconocen los actores que interactúan con las funciones del sistema, así como los casos de uso que representan los requerimientos del sistema. Ver ANEXO A.

NRO.	ACTOR	DESCRIPCIÓN
1	Propietario	Representa a una persona que es responsable de solicitar servicio. Esta persona puede ser un individuo (no asociado a la institución), o alguien asociado con la institución. El Propietario necesita del sistema de Control y Seguimiento de Procesos para comprobar y asegurar su solicitud.
2	Inspector	Representa a una persona que es responsable de distribuir los servicios de inspección a los Propietarios. El Inspector utiliza el sistema de Control y Seguimiento de Procesos para registrar, almacenar, buscar y verificar los datos de los Propietarios.
3	Fiscalizador	Representa a una persona que es responsable de distribuir los servicios de fiscalización a los Propietarios y los servicios de coordinación a los Inspectores. El Fiscalizador utiliza el sistema de Control y Seguimiento de Procesos para registrar, almacenar, verificar, buscar y modificar los datos de los Propietarios. Además, de hacer una lectura de los reportes generados por el sistema para la toma de decisiones.

Tabla 3.1 Actores del Sistema de Fiscalización

Fuente: [Elaboración Propia]

Caso de Uso :	Solicitar Servicio
Actores :	Propietario, Inspector, Fiscalizador
Tipo :	Secundario
Descripción :	El Propietario utiliza este caso de uso para solicitar el servicio de atención después de una notificación que le indicaba apersonarse en la unidad de fiscalización. El caso de uso Solicitar Servicio es el inicio para entrar al sistema de Control y Seguimiento de Procesos.

Tabla 3.2 Caso de Uso - Solicitar Servicio

Fuente: [Elaboración Propia]

Caso de Uso :	Registrar Datos
Actores :	Inspector, Fiscalizador
Tipo :	Primario
Descripción :	El Inspector utiliza este caso de uso para registrar y almacenar los datos del Propietario en la base de datos, también los datos de su vivienda y la inspección realizada a la misma. El caso de uso Registrar Datos efectúa el registro inicial y general de la información.

Tabla 3.3 Caso de Uso - Registrar Datos

Fuente: [Elaboración Propia]

Caso de Uso :	Buscar Datos
Actores :	Inspector, Fiscalizador
Tipo :	Secundario
Descripción :	El Inspector y/o el Fiscalizador utilizan este caso de uso para buscar los datos de un Propietario en particular y brindarle el servicio requerido. El caso de uso Buscar Datos efectúa una búsqueda general en la base de datos sobre algún parámetro especificado por el Inspector o Fiscalizador.

Tabla 3.4 Caso de Uso – Buscar Datos

Fuente: [Elaboración Propia]

Caso de Uso :	Actualizar Datos
Actores :	Fiscalizador
Tipo :	Primario
Descripción :	El Fiscalizador utiliza este caso de uso para realizar algún cambio en la información contenida en el sistema. El caso de uso Actualizar Datos efectúa actualizaciones de la base de datos siguiendo órdenes de modificación o eliminación de la información.

Tabla 3.5 Caso de Uso – Actualizar Datos

Fuente: [Elaboración Propia]

e) Diagrama de Casos de Uso

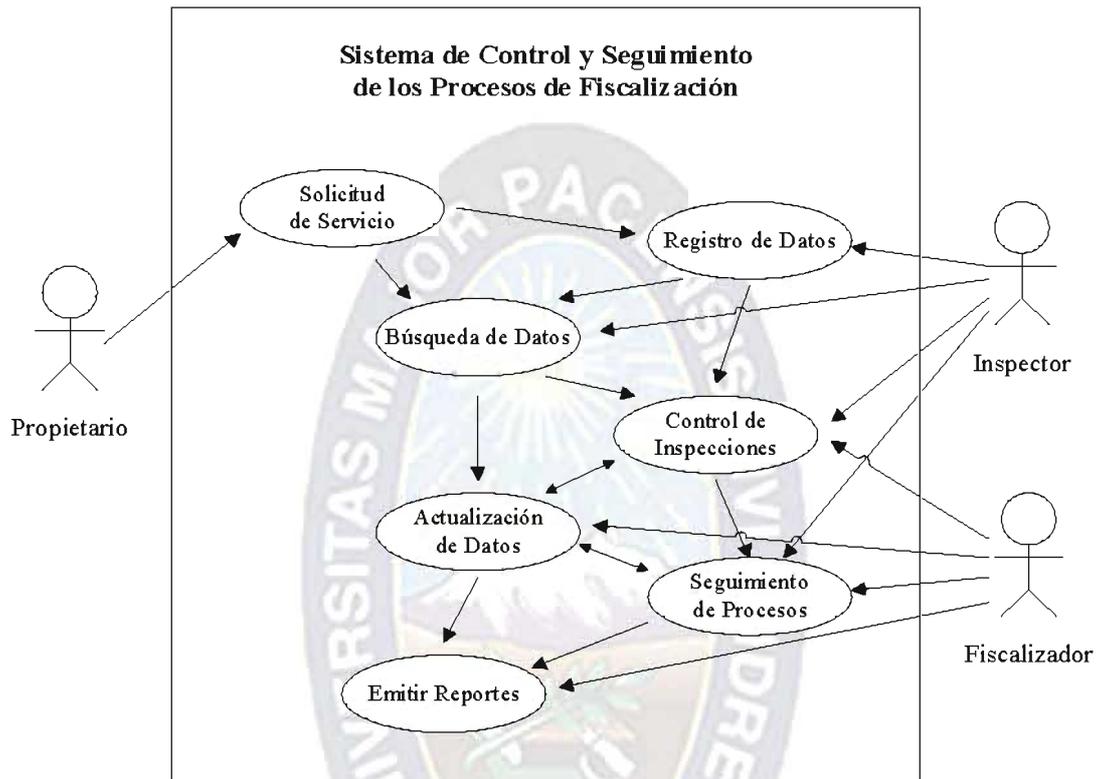


Figura 3.3 Diagrama de Casos de Uso del Sistema

Fuente: [Elaboración Propia]

f) Prioridad de los Casos de Uso

Ahora, se describen los casos de uso importantes para el caso de estudio: Control y Seguimiento de Procesos, tomando en cuenta su especificación (detalles), como los actores que intervienen, una descripción textual, objetivos, precondiciones, postcondiciones, flujo de sucesos y alternativas. Ver ANEXO B.

Caso de Uso: Registrar datos	Flujo de Sucesos: 1. El Inspector predial realiza inspección y determina tipo de infracción. 2. El Fiscalizador inicia sesión en el sistema, con su contraseña. 3. El Fiscalizador registra los datos del Propietario, su vivienda y de la inspección a la misma. 4. El Inspector predial junto al Propietario coordinan con el Fiscalizador una nueva inspección ocular para verificar cambios.
Actores: Propietario, Inspector, Fiscalizador	
Objetivos: Registrar datos del Propietario, su vivienda y la inspección realizada a la misma.	
Descripción: El Fiscalizador registra los datos de identificación del Propietario en el sistema, así como su vivienda y los datos resultantes de la inspección realizada con anterioridad.	
Precondiciones: Realización de la inspección y determinación de la infracción.	Alternativas: En el paso 2, si el Fiscalizador escribe una contraseña errónea, el sistema no inicia sesión.
Postcondiciones: Planificación de próxima entrevista e inspección ocular.	

Tabla 3.9 Caso de Uso expandido - Registrar Datos

Fuente: [Elaboración Propia]

Caso de Uso: Buscar datos	Flujo de Sucesos: 1. El Inspector y/o Fiscalizador inician sesión en el sistema, cada uno con su contraseña correspondiente. 2. El Inspector y/o Fiscalizador se dirigen al enlace relacionado con las búsquedas 'Buscar' e ingresan. 3. El Inspector y/o Fiscalizador ingresan un parámetro especificado para las búsquedas, ejemplo la Hoja de Ruta (HR). 4. El sistema ofrece respuesta a la búsqueda, mostrando los datos requeridos y los relacionados.
Actores: Inspector, Fiscalizador	
Objetivos: Buscar información necesaria para continuar con el trabajo.	
Descripción: El Inspector y/o Fiscalizador revisan el sistema para encontrar algún dato necesario sobre un trámite en particular, como el número de trámite, o sea su Hoja de Ruta o el nombre del propietario, etc.	
Precondiciones: Existencia del registro de la información necesaria en la base de datos.	Alternativas: En el paso 4, si el Inspector y/o Fiscalizador desean seguir haciendo búsquedas, pueden hacerlo realizando los mismos pasos, a partir del paso 2.
Postcondiciones: Concluir con la sesión del sistema si es que no se lo requerirá más.	

Tabla 3.10 Caso de Uso expandido - Buscar Datos

Fuente: [Elaboración Propia]

3.2 FASE DE ELABORACIÓN - ANÁLISIS

a) Modelo Conceptual del Sistema

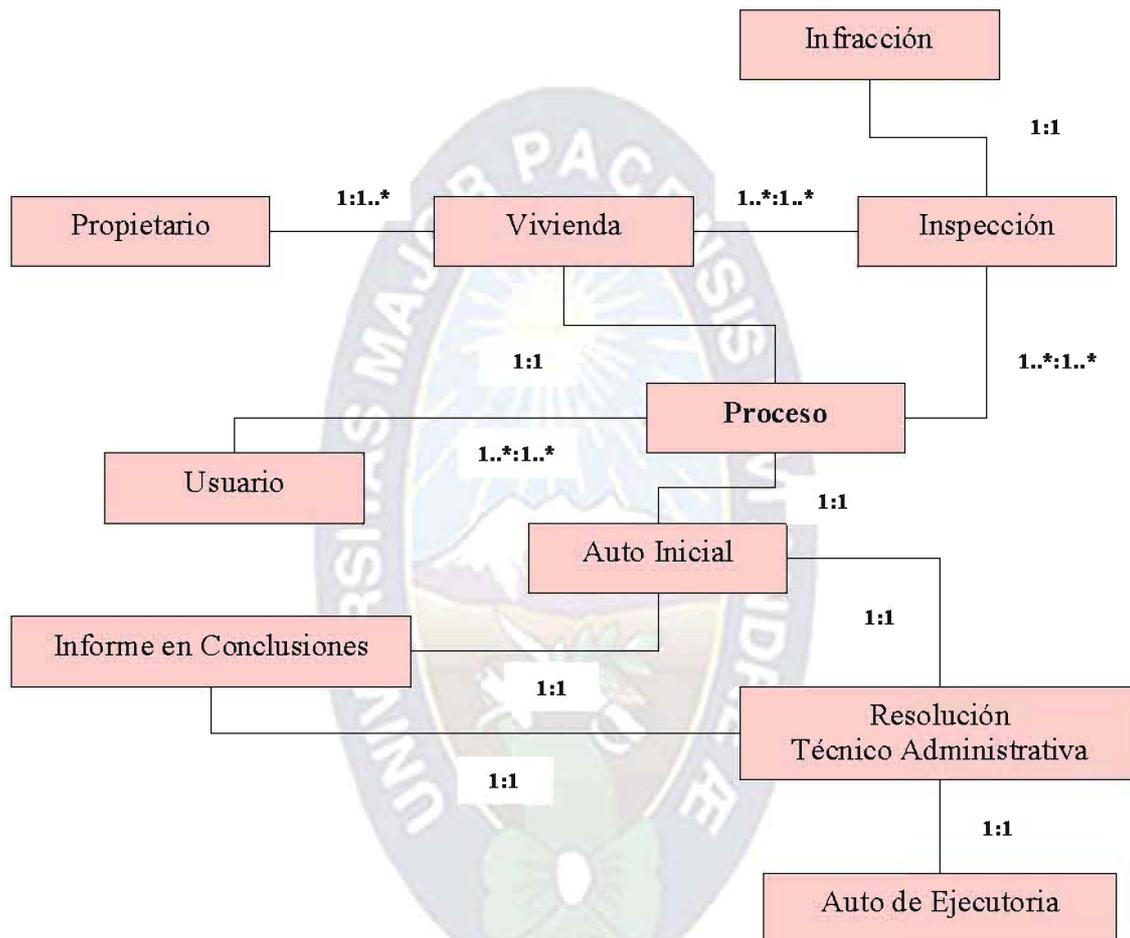


Figura 3.4 Modelo Conceptual del Sistema de Control y Fiscalización

Fuente: [Elaboración Propia]

b) Identificar Clases y Describir Interacciones

Se identifican las clases de control, entidad e interfaz necesarias para realizar los casos de uso y esbozamos sus nombres, responsabilidades, atributos y relaciones.

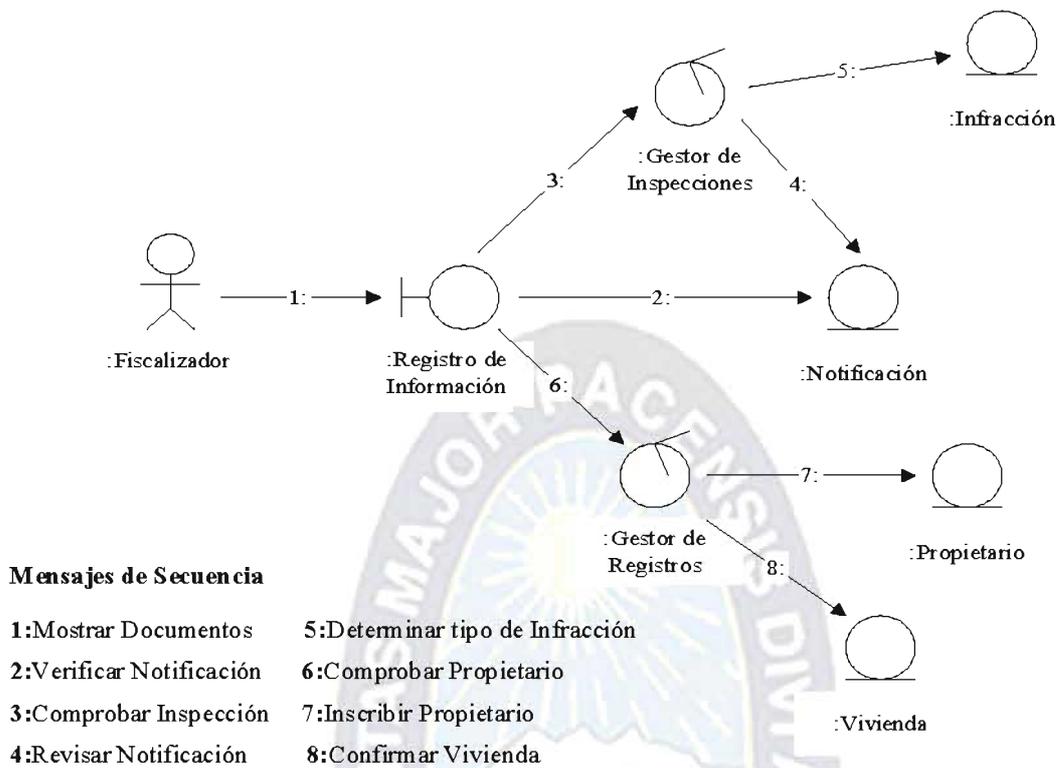


Figura 3.5 Diagrama de Colaboración del Caso de Uso Registro de Información

Fuente: [Elaboración Propia]

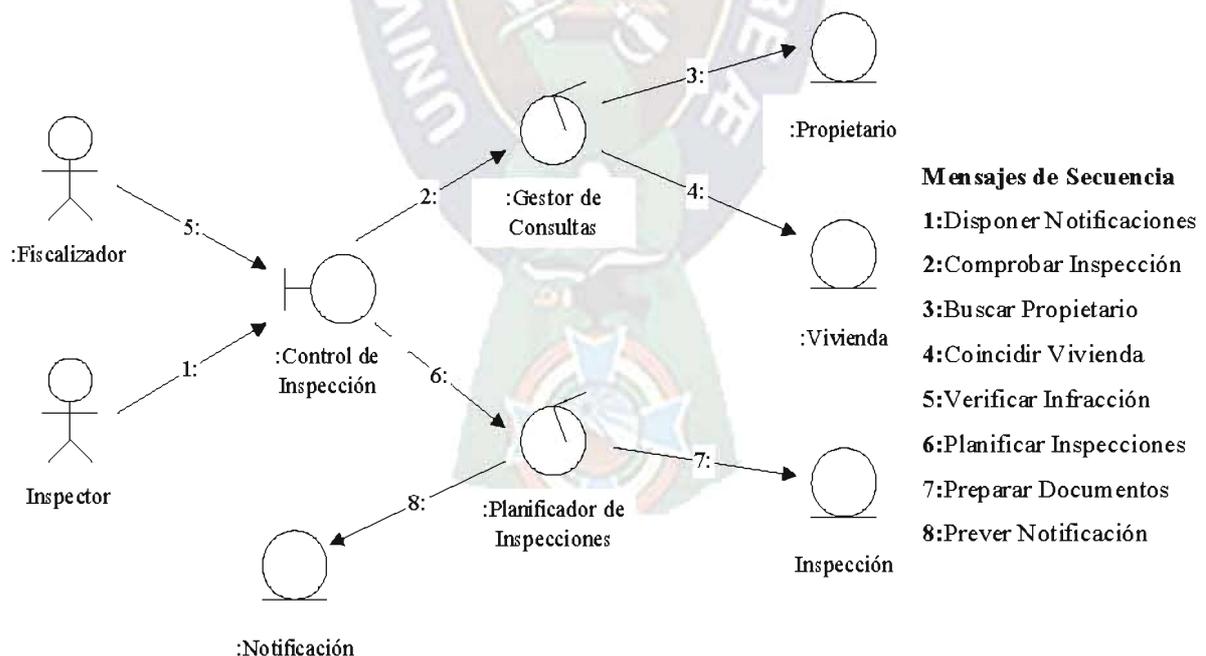


Figura 3.6 Diagrama de Colaboración del Caso de Uso Control de Inspección

Fuente: [Elaboración Propia]

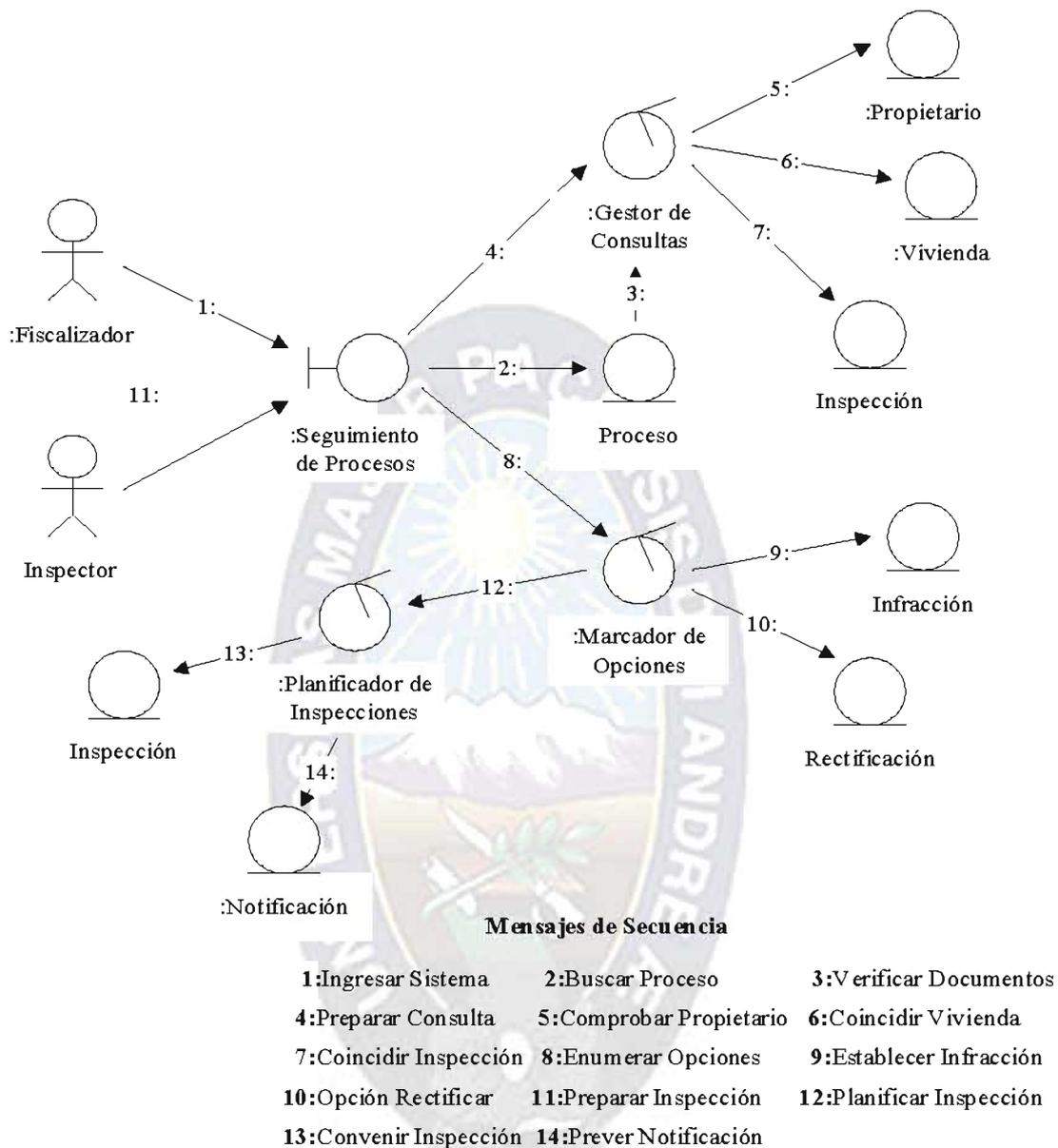


Figura 3.7 Diagrama de Colaboración del Caso de Uso Seguimiento de Procesos

Fuente: [Elaboración Propia]

c) Identificar Métodos o Responsabilidades

Las responsabilidades de una clase se refieren a los roles de cada entidad, clase de control e interfaz, y para encontrarlos se escriben los métodos de estos objetos.

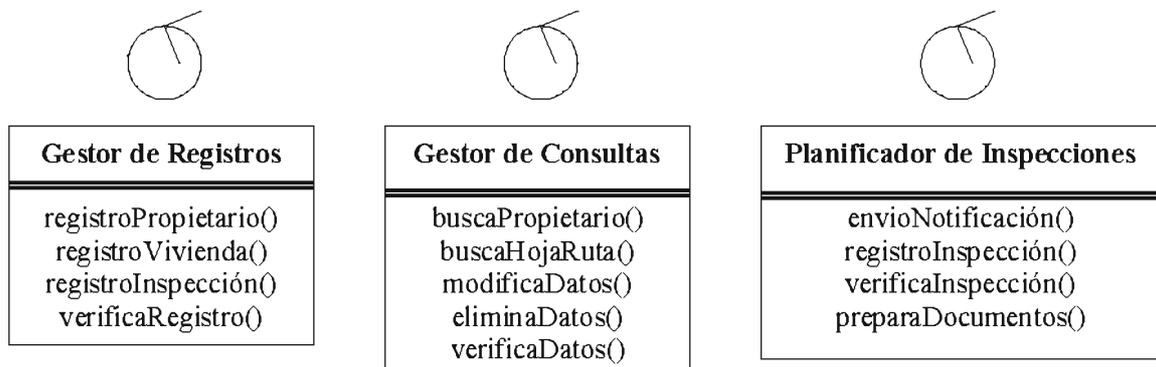


Figura 3.8 Métodos de las Clases de Control Gestor de Registros y Gestor de Consultas

Fuente: [Elaboración Propia]

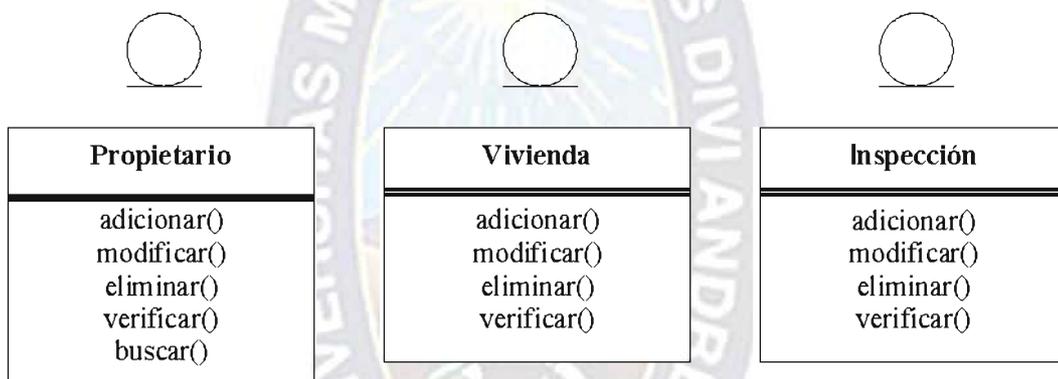


Figura 3.9 Métodos de las Clases Entidad - Propietario, Vivienda, Inspección

Fuente: [Elaboración Propia]

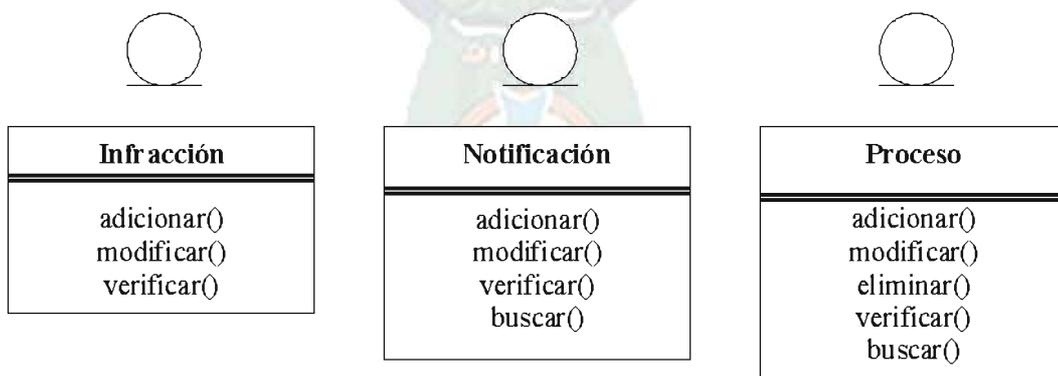


Figura 3.10 Métodos de Clases Entidad – Infracción, Notificación, Proceso

Fuente: [Elaboración Propia]

d) Identificar Atributos

Un atributo especifica una propiedad de una clase de análisis, y normalmente es necesaria para las responsabilidades de su clase. Para esto, solo se visualizarán las clases entidad, dado que las clases de control son clases abstractas, que existen solo durante la ejecución de los casos de uso, por lo que no poseen atributos físicos.

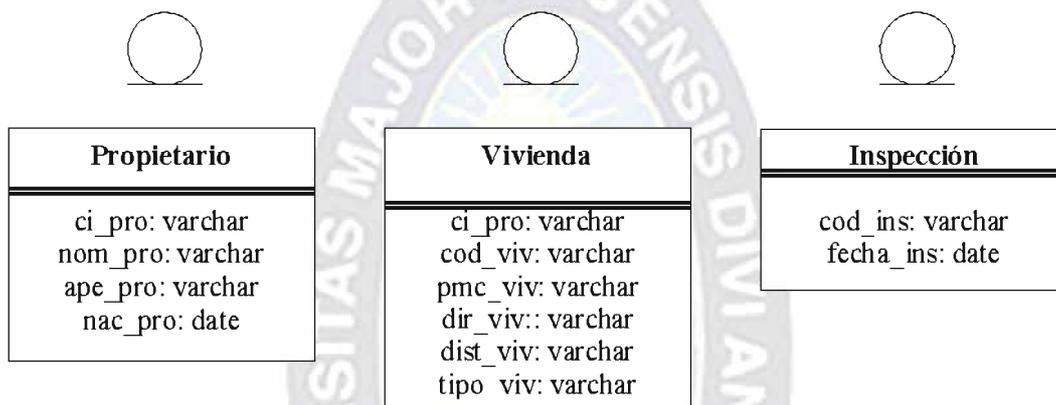


Figura 3.11 Atributos de Clases Entidad – Propietario, Vivienda, Inspección

Fuente: [Elaboración Propia]



Figura 3.12 Atributos de Clases Entidad – Infracción, Notificación, Proceso

Fuente: [Elaboración Propia]

e) Identificar Asociaciones

A seguir, se muestra el diagrama de clases incorporando las asociaciones necesarias y el tipo de multiplicidad que describen:

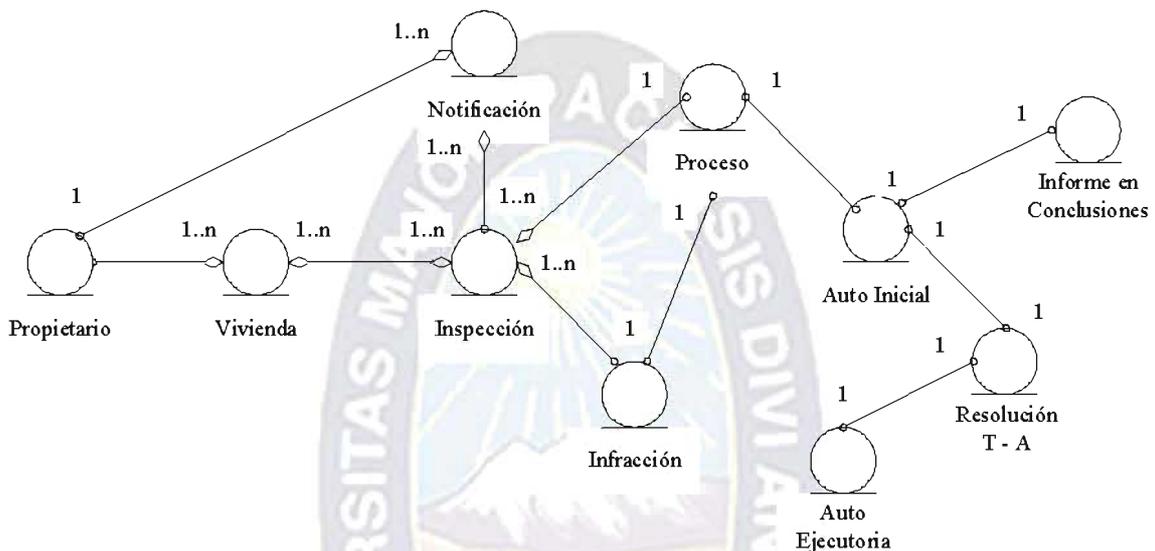


Figura 3.13 Diagrama de Asociaciones del Sistema

Fuente: [Elaboración Propia]

3.3 FASE DE CONSTRUCCIÓN - DISEÑO

a) Organizar Paquetes de Elementos

Los paquetes se usan con el propósito de dividir los elementos de un diagrama de clases en partes más pequeñas, de esta forma, se puede facilitar la comprensión de los diagramas y del sistema en si. A continuación se presenta la clasificación de los elementos con el uso de paquetes.

Véase *figura 3.14* y *figura 3.15*

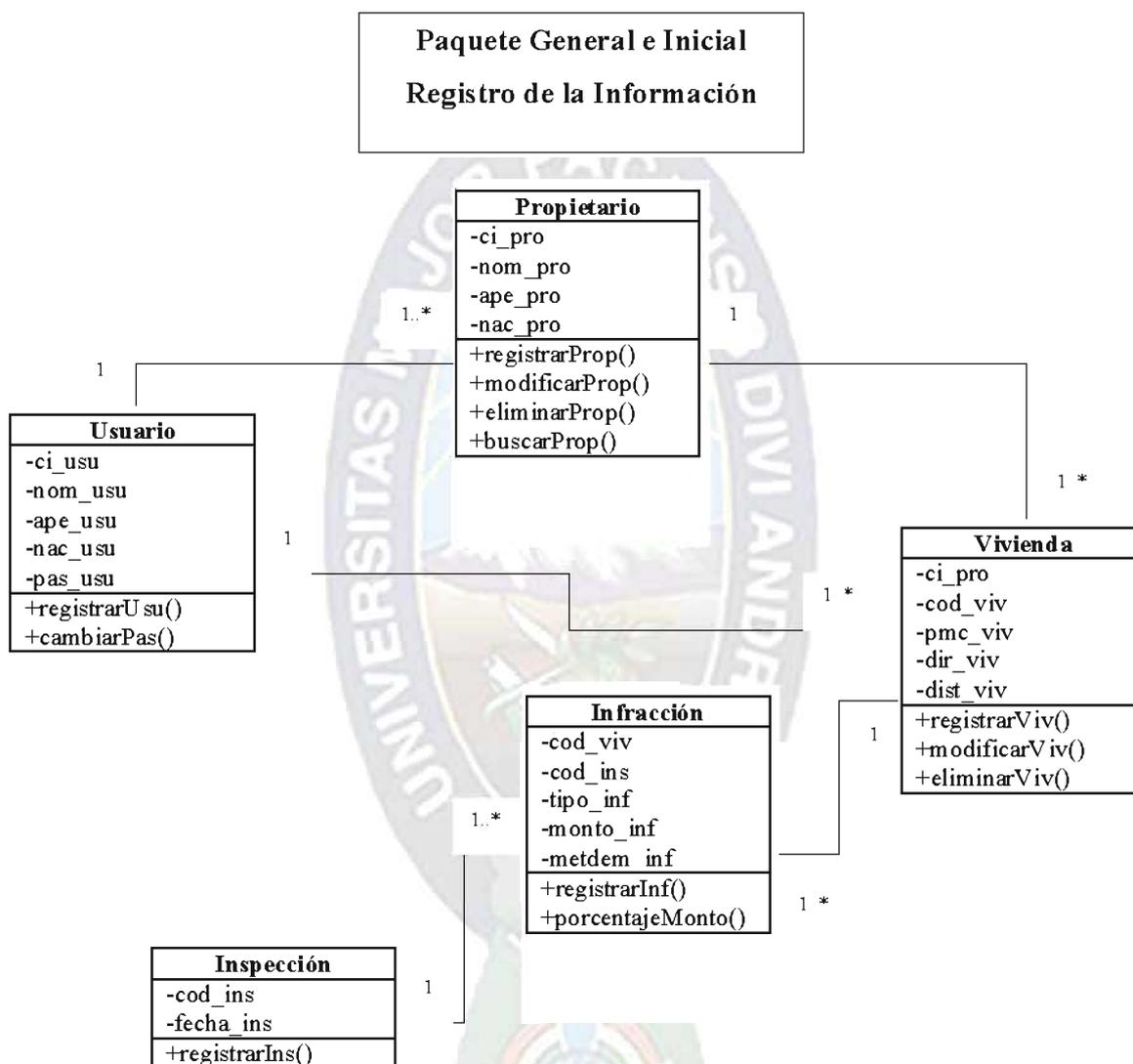


Figura 3.14 Diagrama de Clases – Registro Información

Fuente: [Elaboración Propia]

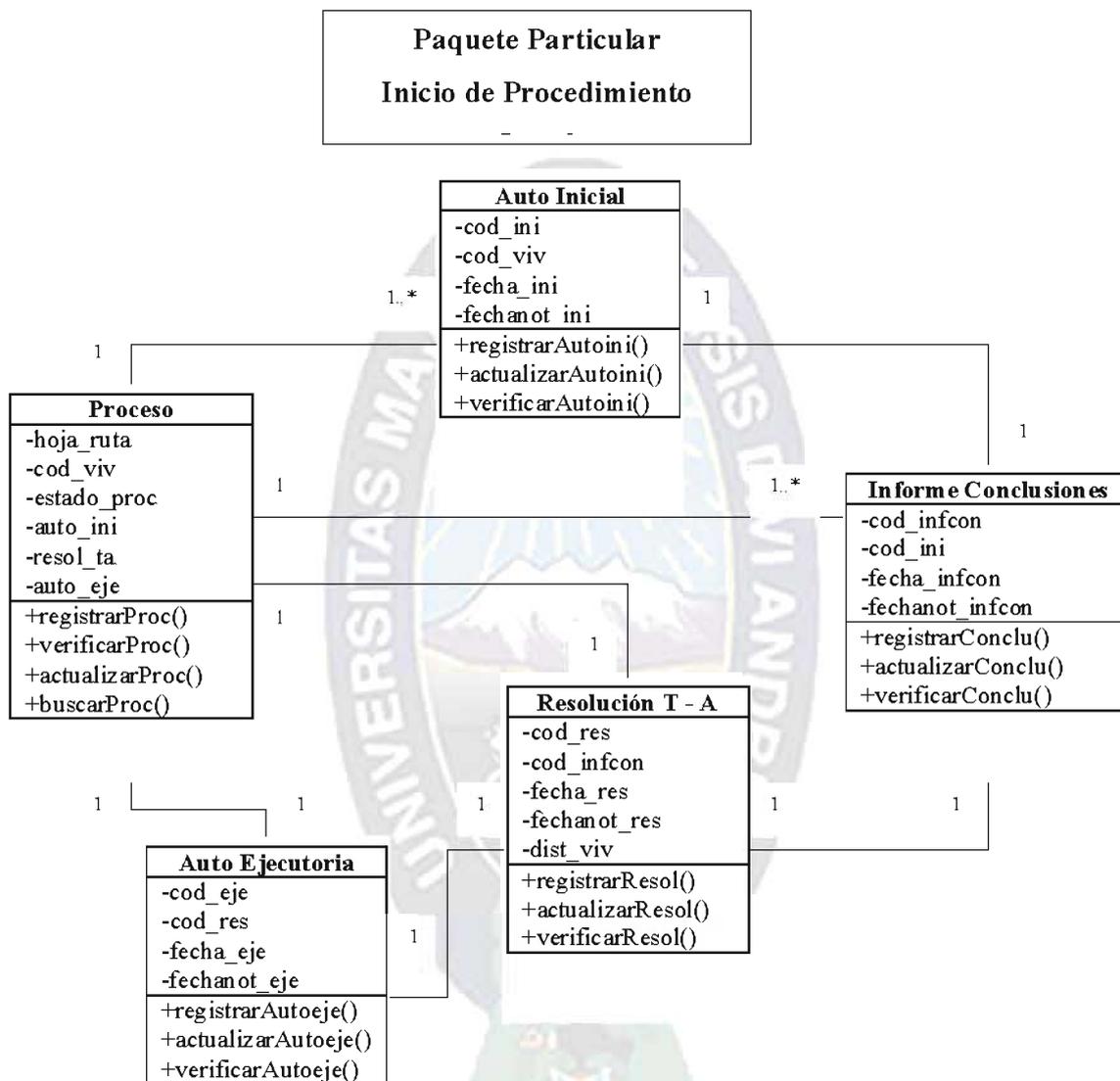


Figura 3.15 Diagrama de Clases del Proceso Particular

Fuente: [Elaboración Propia]

b) Diseño Navegacional del Sistema

Como se definió anteriormente, existen tres actores que interactúan con el sistema de control y seguimiento de los procesos de fiscalización, sin embargo, tan solo dos de ellos utilizan el sistema, esto quiere decir, que se dibujaran dos diseños navegacionales para cada perfil de actor definido. *Figura 3.16* y *Figura 3.17*

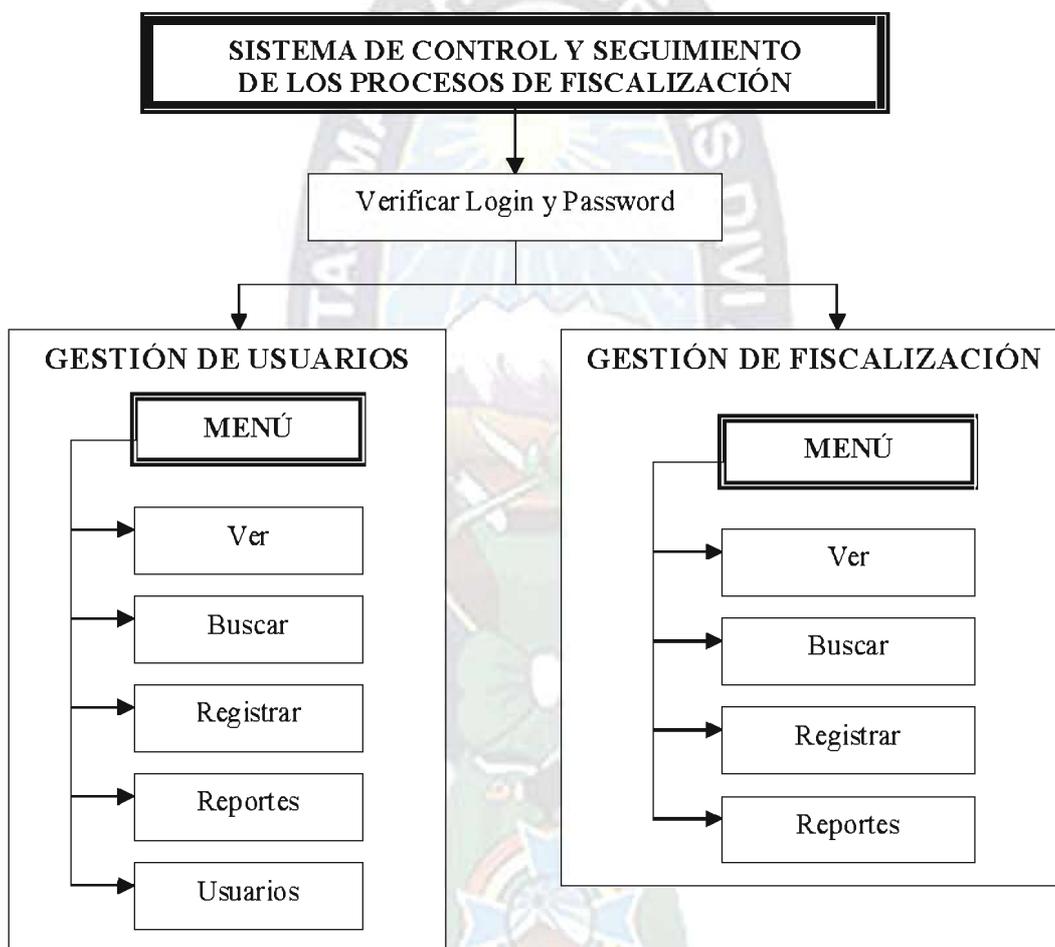


Figura 3.16 Diseño Navegacional – Administrador y/o Fiscalizador

Fuente: [Elaboración Propia]



Figura 3.17 Diseño Navegacional – Inspector

Fuente: [Elaboración Propia]

c) Diagrama de Componentes del Sistema

Cada funcionario del GMLP, puede ingresar a la red Intranet mediante nodos que se comunican con el protocolo TCP/IP de Intranet. Además, la configuración de la red utiliza el patrón habitual de tres capas: la primera capa es la del Cliente (interacciones de los funcionarios), la segunda es la funcionalidad de la Base de Datos, y en la tercera capa es para la Lógica de Aplicación.

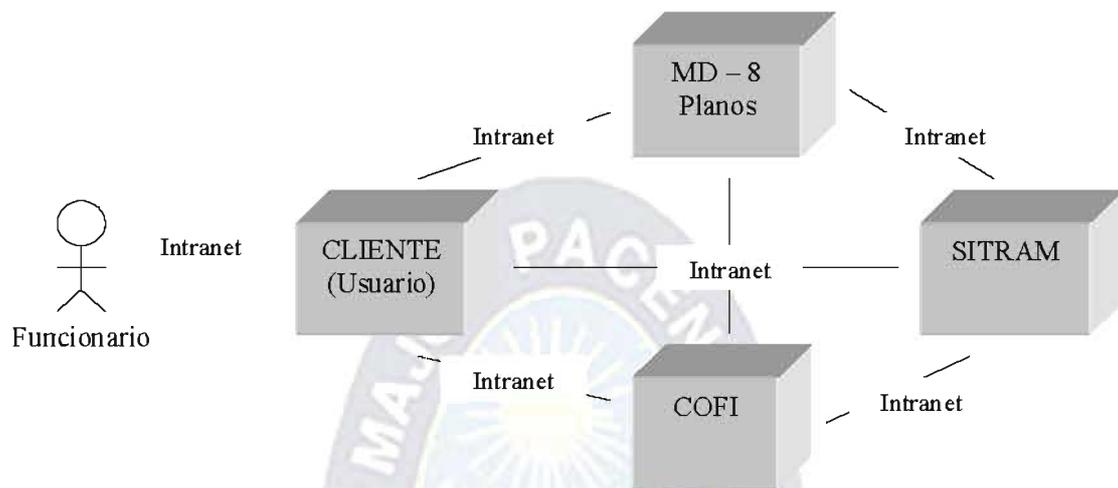


Figura 3.18 Diagrama de Componentes en general

Fuente: [Elaboración Propia]

d) Modelo de Despliegue del Sistema

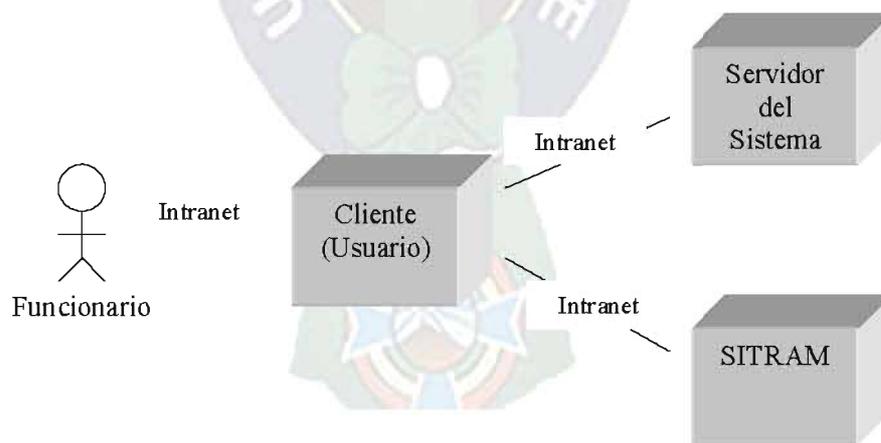


Figura 3.19 Modelo de Despliegue – Procesos de Fiscalización

Fuente: [Elaboración Propia]

3.4 FASE DE TRANSICIÓN - IMPLEMENTACIÓN

a) Modelo de Implementación

Esta actividad tiene como entradas, todos los modelos realizados precedentemente, el Modelo de Análisis, el Modelo del Diseño y el Modelo de Despliegue.

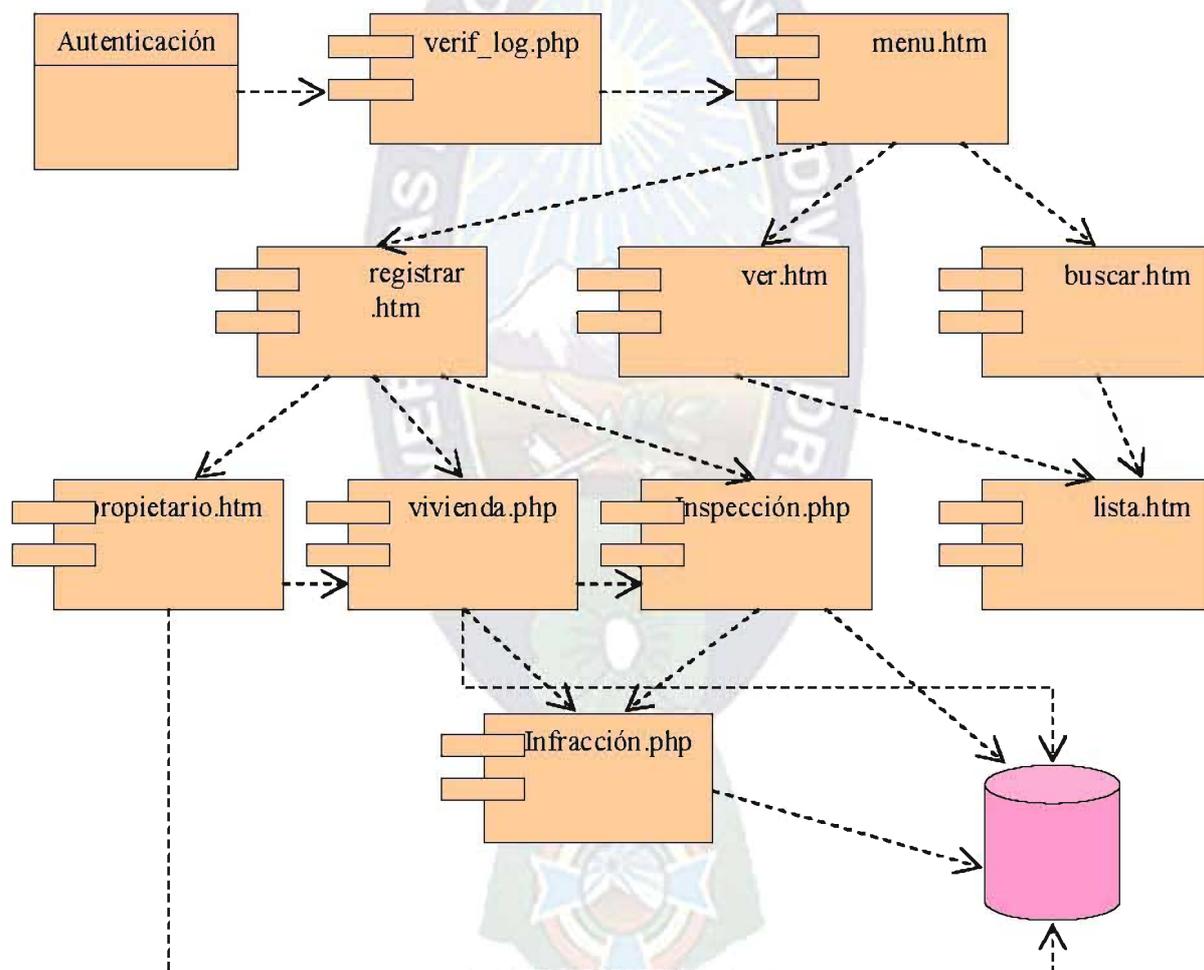


Figura 3.20 Modelo de Implementación – Control y Seguimiento de Procesos

Fuente: [Elaboración Propia]

b) Diseño e Implementación de la Base de Datos

La Base de Datos es relacional, obtenida de la traslación del diagrama de clases al modelo relacional. Contiene 10 tablas que a continuación se especifican:

NRO.	CAMPO	TIPO	DESCRIPCIÓN
T1	Usuario	5 atributos	Funcionario de la Unidad de Fiscalización
	ci_usu *	varchar (8)	Carnet de Identidad del funcionario
	nom_usu	varchar (15)	Nombres del funcionario
	ape_usu	varchar (15)	Apellidos del funcionario
	nac_usu	Date	Fecha de nacimiento del funcionario
	pas_usu	varchar (9)	Contraseña de ingreso al sistema
T2	propietario	4 atributos	Propietario de una vivienda y vecino del M-8
	ci_pro *	varchar (8)	Carnet de Identidad del propietario (dueño)
	nom_pro	varchar (15)	Nombres del propietario de la vivienda
	ape_pro	varchar (15)	Apellidos del propietario
	nac_pro	Date	Fecha de nacimiento del propietario
T3	Vivienda	5 atributos	Propiedad de un vecino del M-8
	ci_pro *	varchar (8)	Carnet de Identidad del propietario
	cod_viv *	varchar (7)	Código de la vivienda (números y letras)
	pmc_viv	varchar (15)	Código de impuestos asignado previamente
	dir_viv	varchar (30)	Dirección de la Vivienda
	dist_viv	varchar (15)	Distrito al que pertenece la Vivienda
T4	Inspección	2 atributos	Verificación de legalidad de una Vivienda
	cod_ins *	varchar (7)	Código de inspección (números y letras)
	fecha_ins	Date	Fecha en la que se realizó la inspección
T5	Infracción	5 atributos	Transgresión de normas por un Propietario
	cod_viv *	varchar (7)	Código de la vivienda (números y letras)
	cod_ins *	varchar (7)	Código de inspección (números y letras)
	tipo_inf	varchar (30)	Tipo de infracción cometida por el Propietario

	mon_inf	Float	Monto pagado por la infracción
	metdem_inf	Float	Metros de demolición estipulados
T6	Autoinicial	4 atributos	Proceso inicial legal
	cod_ini *	varchar (7)	Código de Auto Inicial (números y letras)
	cod_viv *	varchar (7)	Código de la vivienda (números y letras)
	fecha_ini	Date	Fecha en la que se inicio el Auto Inicial
	fechanot_ini	Date	Fecha en la que se realizó la notificación
T7	Informe	4 atributos	Datos de informe gradual de infracción
	cod_infcon *	varchar (7)	Código de Inf. Conclusiones (números y letras)
	cod_ini *	varchar (7)	Código de Auto Inicial (números y letras)
	fecha_infcon	Date	Fecha en la que se inicio el Inf. en Conclusiones
	fechanot_infcon	Date	Fecha en la que se realizó la notificación
T8	Resolución	4 atributos	Proceso secundario legal
	cod_res *	varchar (7)	Código de Resolución T-A (números y letras)
	cod_infcon *	varchar (7)	Código de Inf. Conclusiones (números y letras)
	fecha_res	Date	Fecha en la que se inicio la Resolución T-A
	fechanot_res	Date	Fecha en la que se realizó la notificación
T9	autoejecutoria	4 atributos	Tercer proceso legal que transfiere caso
	cod_eje *	varchar (7)	Código de Auto Ejecutoria (números y letras)
	cod_res *	varchar (7)	Código de Resolución T-A (números y letras)
	Fecha_eje	Date	Fecha en la que se inicio el Auto de Ejecutoria
	Fechanot_eje	Date	Fecha en la que se realizó la notificación
T10	Proceso	3 atributos	Datos generales de los procesos fiscales
	hoja_ruta *	varchar (7)	Número de 5 dígitos asignados por el SITRAM
	cod_viv *	varchar (7)	Código de la vivienda (números y letras)
	Estado_proc	varchar (20)	Breve explicación del estado del proceso

Tabla 3.15 Base de Datos Normalizada del Sistema

Fuente: [Elaboración Propia]

c) Herramientas de Implementación

La implementación del sistema de Control y Seguimiento, se basa en los siguientes requerimientos:

Plataforma de Trabajo: Para el funcionamiento del sistema, es necesario que en el equipo se tenga instalado el Servidor Apache en su versión XAMP, sobre el sistema operativo Windows, preferentemente actualizaciones por encima del 98.

Gestor de Base de Datos: La base de datos del sistema se elaboro en un entorno SQL, específicamente My SQL, debido a las cualidades de este gestor y su facilidad de mantenimiento.

Herramientas de Programación: La herramienta de programación que se utilizo para este proyecto, son: el lenguaje de programación PHP 5.0, el editor HTML de paginas web Dream Weaver.

d) Diseño de Interfaz de Usuario del Sistema

La interfaz del usuario esta diseñado con un menú que conlleva a enlaces que ayudan en la manipulación correcta de la información.

A continuación, se presentan los despliegues de vistas de pantalla de los principales módulos, de esta forma se podrá apreciar el modelado web.



*Figura 3.21 Pantalla de Ingreso al Sistema de Fiscalización
Fuente: [Elaboración Propia]*



*Figura 3.22 Pantalla del Menú de Opciones
Fuente: [Elaboración Propia]*

Unidad de Fiscalización Integral
Sub Alcaldía San Antonio

Sistema de Control y Seguimiento para los Procesos de Fiscalización Integral

LISTA DE PROPIETARIOS

ID	NOMBRE	DIRECCION	FECHA DE NACIMIENTO	MODIFICAR	ELIMINAR
5885456	María Elena	Poma Guaschi	1871-05-12	modificar	eliminar
3877569	Edson	Limachi Mamani	1956-04-20	modificar	eliminar
4258877	Justo Jose	Apaza Fuentes	1884-10-18	modificar	eliminar
3685412	Josefina	Venastegui Lima	1888-09-14	modificar	eliminar
4775112	Susana Lizeth	Herrera Miranda	1875-07-25	modificar	eliminar
4868883	Nieves	Roche Cortez	1880-02-13	modificar	eliminar
3546888	Manuel Marcelo	Vargas Nina	1958-11-28	modificar	eliminar
3423158	Nilo Benjamin	Valdez Mena	1853-08-15	modificar	eliminar
4745459	Marco Antonio	Garcia Mendoza	1878-03-22	modificar	eliminar

Figura 3.23 Pantalla de Listado de Propietarios

Fuente: [Elaboración Propia]

Unidad de Fiscalización Integral
Sub Alcaldía San Antonio

Sistema de Control y Seguimiento para los Procesos de Fiscalización Integral

REGISTRO GENERAL

DATOS PROPIETARIO

C.I. Propietario:

Nombre:

Apellidos:

Fecha Nac.:

Figura 3.24 Pantalla de Registro de Propietarios

Fuente: [Elaboración Propia]

3.5 SEGURIDAD DEL SISTEMA

En un sistema software, se presentan dos formas de establecer la seguridad, una es la seguridad física y la otra es la seguridad lógica.

Para lo que se refiere a la seguridad física, la Unidad de Fiscalización se encarga de establecer políticas de seguridad y confianza entre los funcionarios.

Ahora, para la seguridad lógica, la seguridad de la base de datos satisface este requerimiento. Es entonces, que se aprovecha las distintas facetas y aspectos del Gestor de Base de Datos SQL, en específico, las del My SQL, que por las opciones de seguridad que ofrece justifica la seguridad del sistema.

Por lo tanto, el acceso a la información se dirige por privilegios o niveles de acceso a los usuarios autorizados, es decir, a los usuarios registrados en el sistema como funcionarios de la unidad.

La siguiente tabla muestra los niveles de acceso al sistema ya establecidos.

GRUPO	NIVEL	PRIVILEGIOS
Administrador	1	Todos
Operador Jefe de Unidad	2	Adición, Modificación, Eliminación, Verificación
Operador Inspectores	3	Adición, Verificación

Tabla 3.16 Niveles de Seguridad de Acceso al Sistema

Fuente: [Elaboración Propia]

CAPÍTULO IV

CALIDAD DEL SISTEMA

4.1 CALIDAD DEL SOFTWARE

La calidad del software se apoya en los principios y prácticas de ingeniería del software, para la evaluación y comparación de los datos con las características de los atributos, con el fin de obtener resultados objetivos y justificables.

4.1.1 ANÁLISIS DE PUNTO FUNCIÓN

El Punto Función es una medida del tamaño de un sistema de software y del proyecto que lo construye. Es una unidad de medida así como la hora lo es para medir el tiempo o el kilómetro para medir la distancia. El Análisis de Punto Función se basa en la teoría de que las funciones de una aplicación son la mejor medida del tamaño de un sistema. El Punto Función mide el software mediante la cuantificación de la funcionalidad que el sistema le brinda al usuario basado fundamentalmente en el diseño lógico. Es independiente del lenguaje de computación, de la metodología de desarrollo, de la tecnología utilizada y de la capacidad del equipo de trabajo para desarrollar la aplicación.

Para proceder al cálculo de los puntos función de un sistema han de realizarse tres etapas:

- a) Identificación de los componentes necesarios para el cálculo.
- b) Cálculo de los Puntos Función no ajustados.
- c) Ajuste de los Puntos Función.

a) Identificación de los componentes

Se identifican y enumeran todos los componentes de cada tipo (*entradas externas, salidas externas, grupos lógicos de datos internos, grupos lógicos de datos de interfaz y consultas externas*); además de clasificarlos modularmente, es decir, se identifica a que tipo de modulo pertenece cada componente, seguidamente, se evalúa individualmente la complejidad de cada uno de ellos, utilizando unas tablas ya establecidas que proporcionan el factor de complejidad de cada componente individual.

A continuación se describen los distintos componentes que han de tenerse en cuenta para el cálculo:

Parámetros	DESCRIPCIÓN
Entradas de Usuario	1. Registro Propietario y Vivienda 2. Registro Inspección e Infracción 3. Registro Proceso 4. Registro Auto Inicial y Notificación 5. Registro Resolución T-A y Notificación 6. Registro Auto Ejecutoria y Notificación 7. Registro Usuario
Salidas de Usuario	1. Lista Propietarios y Viviendas 2. Lista Procesos 3. Reporte estadístico
Peticiones o Consultas de Usuario	1. Búsqueda por Proceso 2. Búsqueda por Propietario 3. Búsqueda por Vivienda 4. Consulta estadística
Archivos Lógicos Internos	1. Base de Datos 2. Fichero de Query
Interfaces Externas	1. Verificación Login 2. Verificación Propietario y Vivienda 3. Verificación Inspección e Infracción 4. Verificación Proceso 5. Verificación Auto Inicial 6. Verificación Resolución T-A 7. Verificación Auto Ejecutoria 8. Verificación Usuario

Tabla 4.1 Identificación de los Componentes del Sistema

Fuente: [Elaboración Propia]

b) Cálculo de los Puntos Función No Ajustados

PARÁMETROS DE MEDICIÓN	CONTEO	SENCILLA	MEDIA	COMPLEJA	TOTAL PFNA
Numero de Entradas de Usuario	7	*3	*4	*6	21
Numero de Salidas de Usuario	3	*4	*5	*7	12
Numero de Peticiones de Usuario	4	*7	*10	*15	40
Numero de Archivos	2	*5	*7	*10	10
Numero de Interfaces Externas	8	*3	*4	*6	24
TOTAL SUMA PFNA					107

Tabla 4.2 Cálculo del PF con un Factor de Complejidad

Fuente: [SALAMANCA, 2000, datos propios]

c) Cálculo de los Puntos Función Ajustados

Para Ajustar los Puntos Función, existen 14 atributos de ajuste que impactan en el desarrollo y que deben ser evaluados. A cada atributo se le asignará un valor entre 0 y 5, dependiendo del grado de influencia de éstos. Los posibles valores son:

PARÁMETRO	VALOR	SIGNIFICADO
Sin Influencia	0	El sistema no contempla este atributo.
Influencia Mínima	1	La influencia de este atributo es muy poco significativa.
Influencia Moderada	2	El sistema contempla este atributo y su influencia, aunque pequeña, ha de ser considerada.
Influencia Apreciable	3	La importancia de este atributo debe ser tomada en cuenta, aunque no es fundamental.
Influencia Significativa	4	Este atributo tiene una gran importancia para el Sistema.
Influencia Muy Fuerte	5	Este atributo es esencial para el Sistema y ha de ser tomado en cuenta a la hora del diseño.

Tabla 4.3 Escala de Parámetros de Evaluación

Fuente: [SALAMANCA, 2000]

Los 14 Atributos de Ajuste son:

ATRIBUTO DE AJUSTE		VALOR
1	Comunicación de datos	3
2	Funciones distribuidas	3
3	Prestaciones	3
4	Gran uso de la configuración	2
5	Velocidad de las transacciones	2
6	Entrada de datos en línea	2
7	Diseño para la eficiencia del usuario final	4
8	Actualización de datos en línea	1
9	Complejidad del proceso lógico interno de la aplicación	1
10	Reusabilidad del código	4
11	Facilidad de instalación	3
12	Facilidad de operación	3
13	Localizaciones múltiples	0
14	Facilidad de cambios	3
SVA		34

Tabla 4.4 Cálculo de los Valores de los Atributos de Ajuste

Fuente [SALAMANCA, 2000, datos propios]

Entonces, por la fórmula establecida:

$$\text{Factor de Ajuste FA} = 0.65 + 0.01 * \text{SVA}$$

$$\text{Factor de Ajuste FA} = 0.65 + 0.01 * 34$$

$$\text{Factor de Ajuste FA} = 0.99$$

Por último, se ajustan los Puntos Función mediante la siguiente fórmula:

$$\text{Puntos Función Ajustados PFA} = \text{PFNA} * \text{FA}$$

$$\text{Puntos Función Ajustados PFA} = 107 * 0.99$$

$$\text{Puntos Función Ajustados PFA} = 105.93$$

Donde: PFNA = Puntos Función No Ajustados

4.1.2 CÁLCULO DEL TIEMPO DE ESFUERZO

Una vez ajustados los Puntos Función, bastará multiplicar el valor calculado por los días en que se valore cada Punto Función.

En cada organización se asigna un valor en días diferente para el Punto Función. Es aconsejable que cada organización vaya utilizando su propia experiencia para variar el valor de los Puntos Función dependiendo de sus propios resultados.

Hay quien estima que, inicialmente, se asigne 1 día de esfuerzo por cada Punto Función, de manera que a medida que vayan cerrándose proyectos se vaya modificando tal valor. Otros, basándose en valores medios de la industria informática, recomiendan partir del valor siguiente:

1 Mes de esfuerzo (21 días aprox.) = 18 Puntos Función.

Entonces, tenemos: PFA = 105.93

Si: 21 días = 18 PF → X días = 105.93 PF

Así: $X = (21 \text{ días} * 105.93 \text{ PF}) / 18 \text{ PF}$

X = 123 días de esfuerzo

Este valor es aproximado y estimable.

4.2 CALIDAD EN LA APLICACIÓN WEB

La metodología Web-Site QEM brinda algunas fases para el control de calidad de sitios web, entre las que se tomarán en cuenta son:

Especificación de Requerimientos de Calidad, en otras palabras, se requiere determinar un Árbol de Requerimientos de Calidad especificado por las condiciones que

conlleva el caso de estudio. En dicho árbol de requerimientos se debe determinar las variables IE_i (preferencia de calidad) para cada atributo A_i (hojas de árbol) que forma parte del árbol, estos valores son definidos siguiendo los criterios de evaluación elemental, de este modo el resultado final elemental IE_i se puede interpretar como el grado de aceptabilidad del atributo A .

El objetivo de evaluar la calidad de la aplicación web del sistema, consiste en comprender la calidad global del Sistema de Control y Seguimiento de Procesos de Fiscalización desde el enfoque del usuario, que en este caso vienen a ser los funcionarios de la Unidad de Fiscalización.

Entonces, como ya se menciono antes, es necesario considerar un Modelo de Calidad que especifique características de calidad, esto es el Modelo Estándar ISO 9126, que muestra las siguientes características: usabilidad, funcionalidad, confiabilidad, eficiencia, portabilidad y mantenibilidad.

Luego, describimos el árbol de requerimientos de calidad, que no es más que una lista de especificaciones de las características, sub-características y atributos de calidad, todas conforme al enfoque del usuario.

1. USABILIDAD
1.1 Comprensibilidad Global del sitio
1.2 Mecanismos de Ayuda y Retroalimentación
1.2.1 Calidad de la Ayuda
1.2.2 Indicador de última actualización
1.2.3 Facilidad FAQ
1.3 Aspectos de Interfaces y Estéticos
1.3.1 Cohesividad al agrupar los objetos de control principales
1.3.2 Permanencia y estabilidad en la presentación de los controles principales
1.3.3 Aspectos de Estilo
1.3.3.1 Uniformidad en el color de enlaces
1.3.3.2 Uniformidad en el estilo global
1.3.4 Preferencias Estéticas
2. FUNCIONALIDAD

2.1 Aspectos de búsqueda y recuperación
2.1.1 Mecanismo de búsqueda en el sitio
2.1.2 Mecanismos de Recuperación
2.2 Aspectos de Navegación y Exploración
2.2.1 Navegabilidad
2.2.1.1 Orientación
2.2.1.2 Promedio de Enlaces por página
2.2.2 Objetos de Control Navegacional
2.2.2.1 Nivel de desplazamiento
2.2.2.2.1 Desplazamiento Vertical
2.2.2.2.2 Desplazamiento Horizontal
2.2.3 Predicción Navegacional
2.2.3.1 Enlace con Título (enlace con texto explicatorio)
2.2.3.2 Calidad de la frase de enlace
3. CONFIABILIDAD
3.1 No deficiencia
3.1.1 Errores de enlaces
3.1.1.1 Enlaces rotos
3.1.1.2 Enlaces inválidos
3.1.1.3 Enlaces no implementados
3.1.2 Errores o Deficiencias Varias
3.1.2.1 Deficiencias o cualidades ausentes debido a browsers
3.1.2.2 Deficiencias o resultados inesperados independiente de browsers
3.1.2.3 Nodos Destino inesperadamente en construcción
3.1.2.4 Nodos Web Muertos (sin enlaces de retorno)
4. EFICIENCIA
4.1 Performancia
4.1.1 Páginas de acceso rápido
4.2 Accesibilidad
4.2.1 Accesibilidad de Información
4.2.1.1 Legibilidad al desactivar la Imagen del Browser
4.2.1.1.1 Imagen con título
4.2.1.1.2 Legibilidad Global
4.2.2 Accesibilidad de Ventanas
4.2.2.1 Numero de visitas considerando marcos
4.2.2.2 Versión sin marcos
5. PORTABILIDAD
5.1 Independencia de Hardware
5.2 Independencia de Software

6. MANTENIBILIDAD
6.1 Expandibilidad
6.2 Estabilidad

Tabla 4.5 Árbol de Requerimientos de Calidad

Fuente: [OLSINA, 1999; datos propios]

4.2.1 EVALUACIÓN ELEMENTAL

A partir del árbol de calidad antes esquematizado, y para cada atributo cuantificable A_i se debe asociar y determinar la variable X_i , que tomara un valor real a partir de un proceso de medición. Además, para el rango de valores acordados para la variable X_i , por medio de un criterio elemental, se debe hacer corresponder en una preferencia elemental IE_i .

a) Características

Título: USABILIDAD	
Código:	1
Sub-características	1.1 Comprensibilidad Global del Sitio 1.2 Mecanismos de Ayuda 1.3 Aspectos de Interfaces y Estéticos
Definición	Es una característica de calidad de producto de alto nivel, que se la puede medir mediante cálculo a partir de métricas directas e indirectas, y representa la capacidad o potencialidad del producto para ser utilizado, comprendido y operado por los usuarios, además de ser atractivo.
Peso	Con un peso de 0.3
Operador Aritmético Lógico	C--

Título: FUNCIONALIDAD	
Código:	2
Sub-características	2.1 Aspectos de búsqueda y Recuperación 2.2 Aspectos de Navegación y Exploración
Definición	Es una característica de calidad de producto de alto nivel,

	que se la puede medir mediante cálculo a partir de métricas directas e indirectas, y representa la existencia de un conjunto de funciones y comportamientos de una aplicación que deben satisfacer un conjunto de especificaciones de requisitos en consideración del dominio y de los usuarios.
Peso	Con un peso de 0.3
Operador Aritmetico Lógico	C--

Título: CONFIABILIDAD	
Código:	3
Sub-características	3.1 No deficiencia
Definición	Es una característica de calidad de producto de alto nivel, que se la puede medir mediante cálculo a partir de métricas directas e indirectas, y representa el grado en el que una aplicación esta libre de errores y fallas y es operable bajo ciertas condiciones específicas.
Peso	Con un peso de 0.2
Operador Aritmetico Lógico	C--

Título: EFICIENCIA	
Código:	4
Sub-características	4.1 Performancia 4.2 Accesibilidad
Definición	Es una característica de calidad de producto de alto nivel, que se la puede medir mediante cálculo a partir de métricas directas e indirectas, y representa a la relación entre el grado de performance del artefacto y la cantidad de recursos (tiempo, espacio, etc.) usados bajo ciertas condiciones.
Peso	Con un peso de 0.2
Operador Aritmetico Lógico	C--

Título: PORTABILIDAD	
Código:	5
Sub-características	5.1 Independencia de Hardware 5.2 Independencia de Software

Definición	Es una característica de calidad de producto de alto nivel, que se la puede medir mediante cálculo a partir de métricas directas e indirectas, y representa la facilidad y factibilidad de transferir el artefacto de un contexto físico a otro.
Peso	Con un peso de 0.3
Operador Aritmetico Lógico	C--

Título: MANTENIBILIDAD	
Código:	6
Sub-características	6.1 Expandibilidad 6.2 Estabilidad
Definición	Es una característica de calidad de producto de alto nivel, que se la puede medir mediante cálculo a partir de métricas directas e indirectas, y principalmente representa la facilidad de modificar y testear el artefacto y el software.
Peso	Con un peso de 0.3
Operador Aritmetico Lógico	C--

b) Sub-Características

Título: COMPRENSIBILIDAD GLOBAL DEL SITIO	
Código:	1.1
Súper-característica	1. Usabilidad
Sub-características	No posee
Atributos	No posee
Definición	Es una característica que representa a todas aquellas facilidades que permiten a la audiencia, tener una rápida comprensión tanto de la estructura organizativa, como del contenido del sitio web como un todo, acceso y recorrido del mismo y sus componentes. Por tal razón, los atributos y sub-características se hallan principalmente en la página principal o en los primeros niveles del sitio.
Peso	Con un peso de 0.4
Operador Aritmetico Lógico	A

Título: ASPECTOS DE BÚSQUEDA Y RECUPERACIÓN	
Código:	2.1
Super-característica	2. Funcionalidad
Sub-características	2.1.1 Mecanismos de Búsqueda 2.1.2 Mecanismos de Recuperación
Atributos	No posee
Definición	Es una característica que modela el mecanismo que permite a la audiencia, tener un modo directo de encontrar información u ocurrencias de documentos a partir de palabras o frases claves, y operadores o filtros. Además, permite recuperarla conforme a ciertas preferencias de usuario y obtener retroalimentación de lo sucedido.
Peso	Con un peso de 0.3
Operador Aritmético Lógico	C--

c) Atributos

Título: FACILIDAD FAQ	
Código:	1.23
Característica	1. Usabilidad
Super-características	1.2 Mecanismos de Ayuda y Retroalimentación
Definición	Este atributo representa a un conjunto de preguntas que se realizan con mayor frecuencia, que están ya publicadas en el sitio con sus respectivas respuestas. A su vez, las respuestas pueden estar enlazadas a otros contenidos. Esto favorece al mecanismo de aprendizaje o ayuda, evitando potencialmente la demora de los visitantes.
Tipo de Criterio	Es un criterio multi-nivel, discreto y absoluto, definido como subconjunto; en donde si se evalúa la disponibilidad y tipo de mecanismo, entonces: 0 = No disponible, 1 = FAQ básico, permite acceder a las respuestas generalmente no enlazadas a otros contenidos, 2 = Permite acceder a las respuestas, generalmente enlazadas a otros contenidos.
Escala	

Título: ENLACES ROTOS	
Código:	3.1.1.1
Característica	3. Confiabilidad
Súper-características	Errores de Enlaces
Definición	Este atributo representa básicamente a los enlaces encontrados que conducen a nodos destino ausentes.
Tipo de Criterio	Es un criterio de variable normalizada, continuo y absoluto, donde si BL = Numero de enlaces rotos encontrados TL = Numero total de enlaces del sitio La formula para computar la variable es: $X = 100 - (BL * 100 / TL) * 10$; donde, si $X < 0$ entonces $X = 0$.
Escala	

Título: PÁGINAS DE ACCESO RÁPIDO	
Código:	4.1.1
Característica	4. Eficiencia
Súper-características	Performancia
Definición	Para este atributo, se mide el tamaño de todas las paginas estáticas del sitio web, considerando todos sus componentes gráficos, tabulares y textuales. El tamaño de cada página se especifica como una función del tiempo de espera y de la velocidad mínima establecida para una línea de comunicación dada. Se especifica un tamaño umbral aceptable, para el tamaño total de cada pagina, por ejemplo el de 35.2 kb. Una página de este tamaño requiere 20 segundos para ser bajada a una tasa de 14.400 bps. Ese es el tiempo aceptable que un usuario debe esperar, sin que se ponga impaciente.
Tipo de Criterio	Es un criterio multi-variable, continuo y absoluto.
Escala	

4.2.2 AGREGACIÓN DE PREFERENCIAS PARCIALES

Los valores obtenidos en la evaluación elemental permiten implementar la evaluación global. Aplicando un mecanismo de agregación paso a paso (método LPS). La función de agregación utilizada para obtener las sub-características y características, se basa en la función denominada media de potencia pesada:

$$IG(r) = (P_1 IE_1^r + P_2 IE_2^r + \dots + P_m IE_m^r)^{1/r} \quad (1)$$

Donde:

$$-\infty \leq r \leq +\infty ; 0 \leq IE_i \leq 1 \quad (2)$$

$$(P_1 + P_2 + \dots + P_m) = 1 ; P_i > 0 ; i = 1, \dots, m$$

$$IG(-\infty) = \text{Min} (IE_1, IE_2, \dots, IE_m);$$

$$IG(+\infty) = \text{Max} (IE_1, IE_2, \dots, IE_m);$$

Además:

Cada indicador elemental IE_i debe tener un peso asociado P_i .

La obtención de los indicadores de calidad deben estructurarse y agregarse de un modo de abajo hacia arriba, para permitir el posterior cálculo de las preferencias parciales respectivas. A su vez, repitiendo el proceso de agregación, al final puede obtenerse la estructura de agregación de todo el sistema.

Las estrategias y mecanismos para realizar estas actividades pueden ser más o menos intuitivas, conforme a la experiencia de los evaluadores y el nivel de criticidad del proyecto de evaluación. Por ejemplo, se puede valorar la relativa importancia de los elementos de entrada a las funciones de acuerdo a la intuición y experiencia, o se puede utilizar mecanismos como encuestas y establecer fórmulas de relativa importancia para computar los pesos.

A continuación, se puede ver la agregación de las preferencias parciales para el Sistema de Control y Seguimiento de Procesos de Fiscalización:

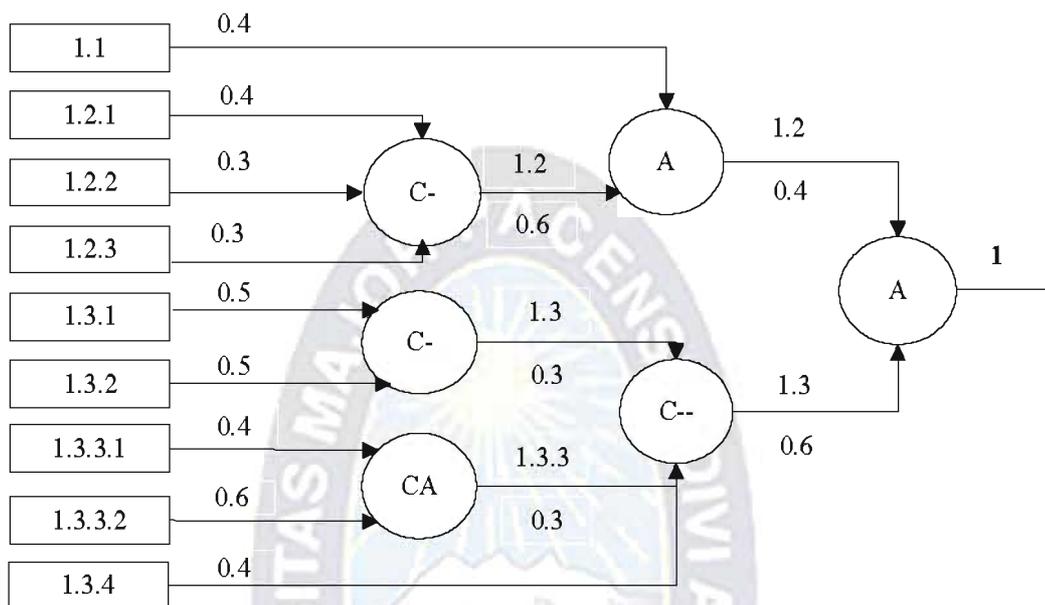


Figura 4.1 Estructura de Agregación para la característica Usabilidad

Fuente: [Elaboración Propia]

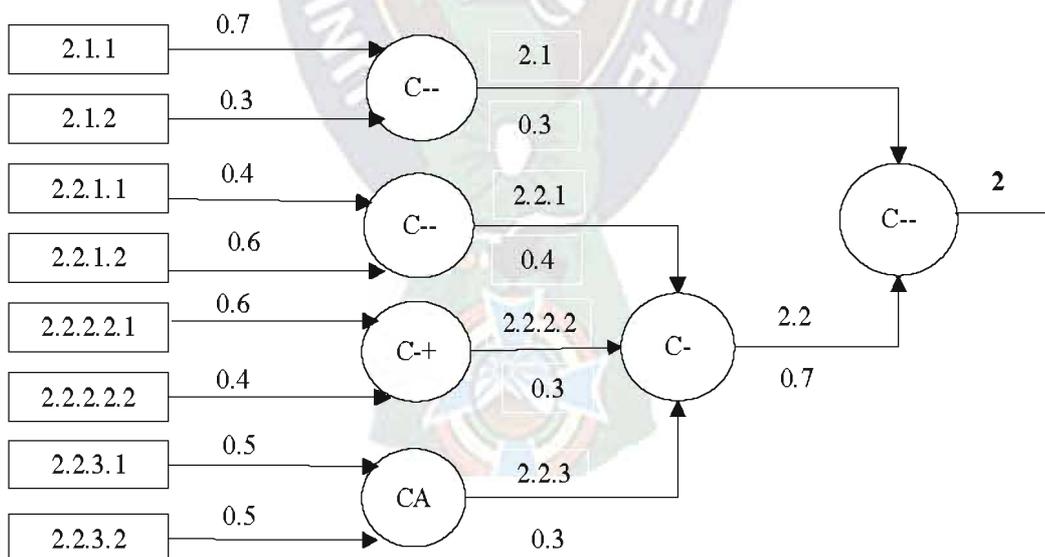


Figura 4.2 Estructura de Agregación para la característica Funcionalidad

Fuente: [Elaboración Propia]

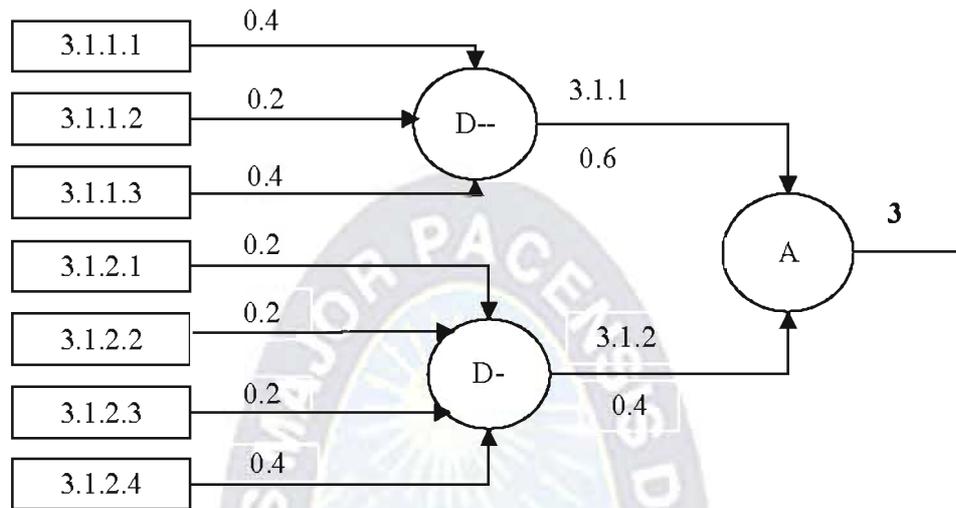


Figura 4.3 Estructura de Agregación para la característica Confianza

Fuente: [Elaboración Propia]

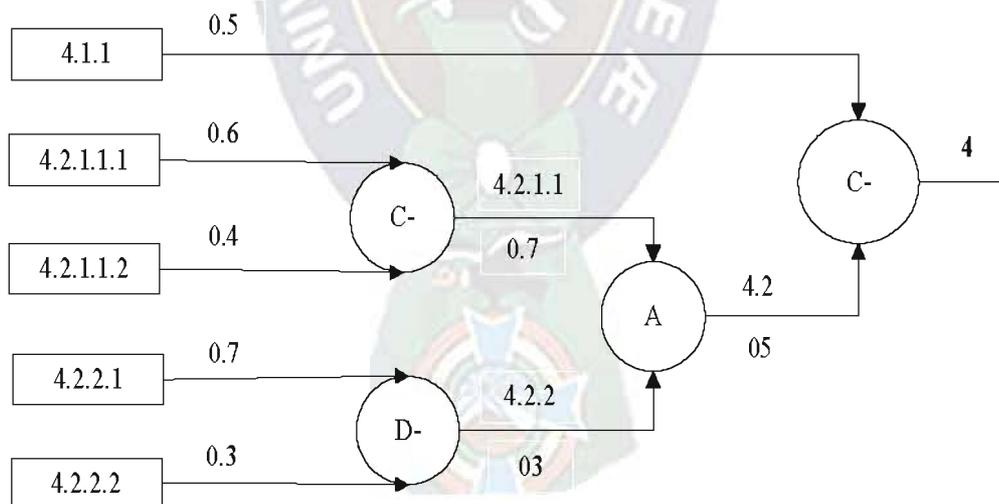


Figura 4.4 Estructura de Agregación para la característica Eficiencia

Fuente: [Elaboración Propia]

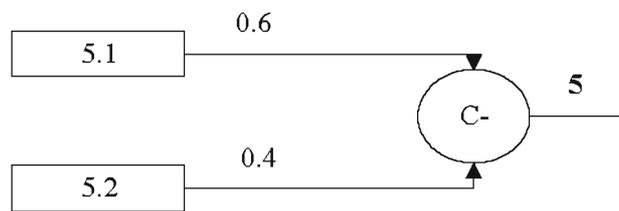


Figura 4.5 Estructura de Agregación para la característica Portabilidad

Fuente: [Elaboración Propia]

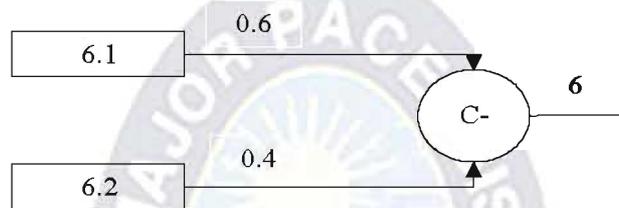


Figura 4.6 Estructura de Agregación para la característica Mantenibilidad

Fuente: [Elaboración Propia]

Luego, se muestra la estructura de agregación de las preferencias parciales de calidad, de las características de más alto nivel, esto para representar la preferencia de calidad del Sistema de Control y Seguimiento de Procesos de Fiscalización.

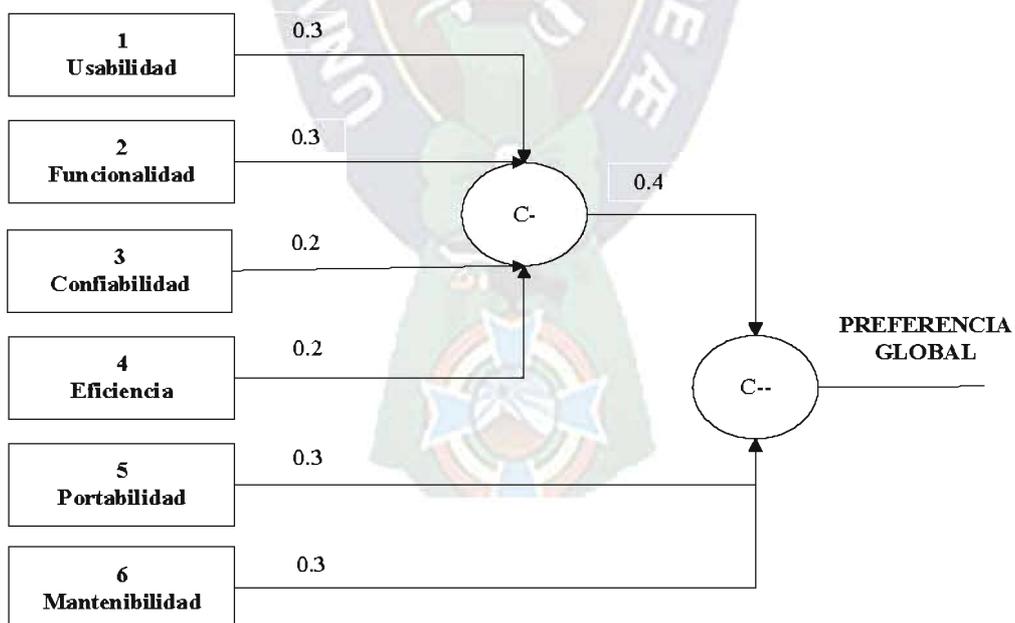


Figura 4.7 Estructura de Agregación de Preferencias Parciales y Globales

Fuente: [Elaboración Propia]

CÓDIGO	CARACTERÍSTICAS, SUB-CARACTERÍSTICAS Y ATRIBUTOS	IE*100	P	R	P*IE ^r
1	USABILIDAD	56.08		1	53.64
1.1	Comprensibilidad Global del sitio	50.5	0.4	1	20.2
1.1.1	Esquema de Organización Global	0	0	0	
1.2	Mecanismos de Ayuda	47.75	0.6	1	28.65
1.2.1	Calidad de Ayuda	55.5	0.4	0.192	0.86
1.2.2	Indicador de última actualización global	40	0.3	0.192	0.61
1.2.3.1	Directorio E-mail	0	0	0	
1.2.4	Facilidad FAQ	0	0	0	
1.2.5	Retroalimentación	0	0	0	
1.2.5.3	Comentarios, Sugerencias	0	0	0	
1.3	Aspectos de Interfaces y Estéticos.	70	0.3	0.573	3.42
1.3.1	Cohesividad al agrupar los objetos de control principales	70	0.5	0.261	1.52
1.3.2	Permanencia y estabilidad en la presentación de los controles principales	70	0.5	0.261	1.52
1.3.2.3	Estabilidad	0	0	0	
1.3.3	Aspectos de Estilo	85	0.3	0.573	3.83
1.3.3.1	Uniformidad en el color de enlaces	80	0.4	-0.720	0.01
1.3.3.2	Uniformidad en el estilo global	90	0.6	-0.720	0.02
1.3.3.3	Guía de estilo global	0	0	0	
1.3.4	Preferencias Estéticas	90	0.4	0.573	5.27
2	FUNCIONALIDAD	73.625		0.619	70.6
2.1	Aspectos de búsqueda y recuperación	70.5	0.3	0.619	4.18
2.1.1	Mecanismo de búsqueda en el sitio	75.5	0.7	0.261	2.16
2.1.2	Mecanismo de recuperación	65.5	0.3	0.261	0.89
2.2	Aspectos de Navegación y Exploración	76.75	0.7	0.192	1.61
2.2.1	Navegabilidad	80.25	0.4	0.619	6.04
2.2.1.1	Orientación	70.5	0.4	0.261	1.22
2.2.1.2	Promedio de Enlaces por página	90	0.6	0.261	1.94
2.2.2	Nivel de desplazamiento	70	0.3	-0.148	0.16
2.2.2.1	Desplazamiento Vertical	80	0.6	-0.148	0.31
2.2.2.2	Desplazamiento Horizontal	60	0.4	-0.148	0.22
2.2.3	Predicción Navegacional	80	0.3	-0.720	0.01
2.2.3.1	Enlace con Título (enlace con texto explicatorio)	70	0.5	-0.720	0.02
2.2.3.2	Calidad de la frase de enlace	90	0.5	-0.720	0.02
3	CONFIABILIDAD	76.6		1	73.45
3.1	No deficiencia	76.6	0.6	1	45.96
3.1.1	Errores de enlaces	76.66	0.6	1	45.99
3.1.1.1	Enlaces rotos	75	0.4	2.187	50.44
3.1.1.2	Enlaces inválidos	85	0.2	2.187	33.16
3.1.1.3	Enlaces no implementados	70	0.4	2.187	43.37

3.1.2	Errores o Deficiencias Varias	73.75	0.4	1	29.5
3.1.2.1	Deficiencias o cualidades ausentes debido a browsers	65	0.2	2.302	29.80
3.1.2.2	Deficiencias o resultados inesperados independiente de browsers	65	0.2	2.302	29.80
3.1.2.3	Nodos Destino inesperada/ en construcción	85	0.2	2.302	55.27
3.1.2.4	Nodos Web Muertos (sin enlaces de retorno)	80	0.4	2.302	96.15
4	EFICIENCIA	72.81		0.261	70.23
4.1	Performancia	85.5	0.5	0.261	1.59
4.1.1	Páginas de acceso rápido	85.5	0.5	0.261	1.59
4.2	Accesibilidad	60.125	0.5	1	30.06
4.2.1	Accesibilidad de Información	60	0.5	0.261	1.46
4.2.1.1	Soporte a versión sólo texto	0	0	0	
4.2.1.2	Legibilidad al desactivar la Imagen del Browser	80	0.7	0.261	2.20
4.2.1.2.1	Imagen con título	80	0.6	0.261	1.88
4.2.1.2.2	Legibilidad Global	80	0.4	0.261	1.26
4.2.2	Accesibilidad de Ventanas	60.25	0.3	2.018	11.72
4.2.2.1	Número de visitas considerando marcos (frames)	80	0.7	2.018	48.47
4.2.2.2	Versión sin marcos	40.5	0.3	2.018	52.5
5	PORTABILIDAD	80.5		0.261	79.18
5.1	Independencia de Hardware	75.5	0.6	0.261	1.85
5.2	Independencia de Software	85.5	0.4	0.261	1.28
6	MANTENIBILIDAD	90		0.261	86.18
6.1	Expandibilidad	90	0.6	0.261	1.94
6.2	Estabilidad	80	0.4	0.261	1.26

Tabla 4.6 Resultados Parciales y Globales de Calidad

Fuente: [Elaboración Propia]

Donde:

IE = Indicador Elemental

P = Peso o importancia relativa de cada entrada, según la restricción (2)

r = Valor real conforme al operador lógico y la cantidad de entradas

Luego, por la ecuación (1):

$$IG(r) = (P_1 IE^{r_1} + P_2 IE^{r_2} + \dots + P_m IE^{r_m})^{1/r}$$

Los resultados globales serán los Indicadores Globales (IG):

CÓDIGO	CARACTERÍSTICAS, SUB-CARACTERÍSTICAS Y ATRIBUTOS	IG
1	USABILIDAD	53.64
1.1	Comprensibilidad Global del sitio	50.5
1.2	Mecanismos de Ayuda	47.7
1.3	Aspectos de Interfaces y Estéticos.	70
2	FUNCIONALIDAD	70.6
2.1	Aspectos de búsqueda y recuperación	70.5
2.2	Aspectos de Navegación y Exploración	76.7
3	CONFIABILIDAD	73.4
3.1	No de eficiencia	76.6
4	EFICIENCIA	70.2
4.1	Performancia	85.5
4.2	Accesibilidad	60
5	PORTABILIDAD	79.1
5.1	Independencia de Hardware	75.5
5.2	Independencia de Software	85.5
6	MANTENIBILIDAD	86.1
6.1	Expandibilidad	90
6.2	Estabilidad	80
TOTAL		72.21

Tabla 4.7 Indicadores Globales de Calidad para el Sistema Control y Seguimiento de Procesos de Fiscalización

Fuente: [Elaboración Propia]

Un resultado de **72.21** en el Indicador Global indica, según la tabla 2.1 del Capítulo II, que el sistema presenta una calidad de la aplicación web satisfactoria. Sin embargo, es necesario mencionar que índices bajos en la *Comprensión Global del Sitio* implican una falencia en la *Tabla de Contenidos* y *Mapa del Sitio*, lo que involucra una futura mejora en el sistema, para hacerlo más entendible a primera vista para los usuarios.

CAPÍTULO V

CONCLUSIONES Y RECOMENDACIONES

5.1 CONCLUSIONES

Después de realizar el análisis, diseño, implementación y evaluación del Sistema de Control y Seguimiento de los Procesos de Fiscalización, se concluye lo siguiente:

- El presente proyecto de grado permite establecer y registrar la información de manera óptima en un 80 por ciento, por lo que permite un eficaz acceso a la Base de Datos.
- Después de identificar y analizar los procedimientos computacionales de la información se logro nuevos procesos de control y seguimiento, los mismos que mejoran el rendimiento administrativo en la unidad en un 85 por ciento.
- La implementación de un mecanismo de seguridad para la base de datos del sistema hace efectivo el resguardo y la integridad de la información, de tal forma que alcanza un 95 por ciento de aprobación para los funcionarios.
- El sistema presentado es capaz de generar reportes estadísticos, coadyuvando así a la toma de decisiones, al mismo tiempo crea un 90 por ciento de expectativas de trabajo para los propios funcionarios de la Unidad.
- Una evaluación de calidad web para un sistema intranet, por un lado subsanó los requerimientos del sistema, y por el otro, reveló un 72 por ciento de calidad web, indicando un nivel satisfactorio en la evaluación.

5.2 RECOMENDACIONES

Luego de la obtención satisfactoria de los objetivos, se recomienda:

- Para resguardar la información, el administrador del sistema debe realizar un seguimiento periódico de los usuarios autorizados, de acuerdo a los cargos que desempeñan.
- Ampliar las investigaciones concernientes a los sistemas de control y seguimiento, pues la mayoría de los sistemas están orientados a realizar dichas funciones.
- Se puede ampliar el estudio de este proyecto, puesto que las limitaciones del sistema vienen dadas por las características de los requerimientos, sin embargo, es posible incorporar nuevos módulos de función.



BIBLIOGRAFÍA

- [HANSEN, 1997] Gary W. Hansen, James V. Hansen, *“Diseño y Administración de Base de Datos”*, 2da. Edición, Ed. Prentice Hall, Madrid, 1997
- [JACOBSON, BOOCH, RUMBAUGH, 2000] Jacobson Ivar, Booch Grady, Rumbaugh James, *“El Proceso Unificado de Desarrollo de Software”*, Ed. Addison Wesley, Madrid, 2000
- [KENDALL & KENDALL, 1997] Kendall Kenneth, Kendall Julie, *“Análisis y Diseño de Sistemas”*, 3ra. Ed. Prentice Hall Hispanoamericana, 1997
- [LARMAN, 1999] Graig Larman, *“UML y Patrones – Introducción al Análisis y Diseño Orientado a Objetos”*, Ed. Prentice Hall, México, 1999
- [LOAYZA, 2004] Mario Loayza, *“Apuntes de Taller de Sistemas de Información”*, Bolivia, 2004
- [PRESSMAN, 2002] Pressman Roger S., *“Ingeniería del Software - Un Enfoque Práctico”*, 5ta. Ed. McGraw Hill, Madrid, 2002
- [TAPIA, 2006] Tapia Selaez Javier F., *“Sistema de Información para el Seguimiento y Control de Procesos de Fiscalización Tributaria de Bienes Inmuebles”*, 2006

ARTÍCULOS VIRTUALES:

- [BERZAL, 2000] Fernando Berzal, *“UML Programación Orientada a Objetos”*, 2000
- [CUEVA, 1999] Juan Manuel Cueva Lovelle, *“Introducción a UML – Lenguaje para Modelar Objetos”*, Universidad de Oviedo, España, 1999

Disponible en:

<http://gidis.ing.unlpam.edu.ar/downloads/pdfs/IntroduccionUML.PDF>

- [CUEVA, 1999] Juan Manuel Cueva Lovelle, *“Métricas de Usabilidad en la Web”*, Universidad de Oviedo, España, 1999

- [FERRÉ, SÁNCHEZ, 2001] Xavier Ferré Grau, María Isabel Sánchez Segura, *“Desarrollo Orientado a Objetos con UML”*, Facultad de Informática - UPM
Disponible en:
<http://www.itox.mx/Comunidad/Librero/umITotal.pdf>

- [OLSINA, 1999] Luis Antonio Olsina, *“Metodología Cuantitativa para la Evaluación y Comparación de la Calidad de Sitios Web”*, Universidad Nacional de La Plata, Argentina, 1999

- [SALAMANCA, 2000] Jorge Salamanca Escorial, *“Calidad del Software”*, 2000

- [SILVA & MARCERAT, 2002] D. Silva & B. Marcerat , *“Construyendo Aplicaciones Web con una Metodología de Diseño Orientado a Objetos”*, 2002

ANEXO A.

Actores y Casos de Uso de Alto Nivel.

Caso de Uso :	Control de Inspección
Actores :	Inspector, Fiscalizador
Tipo :	Primario
Descripción :	El Inspector y/o el Fiscalizador utilizan este caso de uso para realizar una organización y coordinación de las inspecciones que se ejecutaran en un cierto periodo de tiempo. El caso de uso Control de Inspección efectúa una secuencia de pasos de verificación de la información.

Tabla 3.6 Caso de Uso – Control de Inspección

Fuente: [Elaboración Propia]

Caso de Uso :	Seguimiento del Proceso
Actores :	Inspector, Fiscalizador
Tipo :	Primario
Descripción :	El Inspector y/o el Fiscalizador utilizan este caso de uso para planificar una entrevista con el Propietario, para llegar a un convenio entre partes. El caso de uso Seguimiento del Proceso efectúa una secuencia de pasos de revisión y comprobación de la información.

Tabla 3.7 Caso de Uso – Seguimiento del Proceso

Fuente: [Elaboración Propia]

Caso de Uso :	Emitir Reportes
Actores :	Inspector, Fiscalizador
Tipo :	Secundario
Descripción :	El Fiscalizador utiliza este caso de uso para realizar una lectura de la información refinada para luego tomar decisiones. El caso de uso Generar Reportes efectúa una salida con información depurada en base a parámetros establecidos.

Tabla 3.8 Caso de Uso – Emitir Reportes

Fuente: [Elaboración Propia]

ANEXO B.

Prioridad de los Casos de Uso.

Caso de Uso: Actualizar datos	Flujo de Sucesos: 1. El Fiscalizador inicia sesión en el sistema. 2. El Fiscalizador ingresa al enlace 'Ver' para escoger la información que va actualizar. 3. El Fiscalizador ingresa al registro de los datos reconocidos por la base de datos, y decide si los va a modificar o eliminar definitivamente. 4. El Fiscalizador verifica actualización, mediante el sistema. 5. El Fiscalizador cierra su inicio de sesión.
Actores: Fiscalizador	
Objetivos: Actualizar la información de la base de datos.	
Descripción: El Fiscalizador inicia sesión en el sistema para actualizar datos, ya sea, modificándolos y/o eliminándolos.	
Precondiciones: Existencia de la información necesaria en la base de datos.	Alternativas: Para el paso 1, solo el Fiscalizador (Jefe de Unidad) puede realizar este procedimiento, ingresando con su contraseña.
Postcondiciones: Cerrar sesión de usuario autorizado para la actualización de la base de datos.	

Tabla 3.11 Caso de Uso expandido - Actualizar Datos

Fuente: [Elaboración Propia]

Caso de Uso: Control de inspección	Flujo de Sucesos: 1. El Inspector revisa base de datos del sistema y clasifica casos fiscales. 2. El Fiscalizador verifica los casos fiscales, revisando la duración de cada trámite y el estado en el que se encuentra. 3. El Fiscalizador Indica fechas de operativos de inspección. 4. El Inspector toma nota de las fechas y planifica entrega de Notificaciones previas.
Actores: Inspector, Fiscalizador	
Objetivos: Coordinar inspecciones en un periodo de tiempo, basados en la información del sistema.	
Descripción: El Inspector coordina con el Fiscalizador una serie de inspecciones, todas basadas en el estado legal de cada Propietario y su vivienda.	

Precondiciones: Primera inspección realizada a la vivienda del Propietario.	Alternativas: Para el paso 2, el estado de cada caso fiscal se refiere al estado del tramite, si esta en estado de notificación, o ya ingreso al estado de proceso legal.
Postcondiciones: Decisión de iniciar la primera etapa del proceso legal.	

Tabla 3.12 Caso de Uso expandido - Control de Inspección

Fuente: [Elaboración Propia]

Caso de Uso: Seguimiento del proceso	Flujo de Sucesos: 1. El Fiscalizador inicia sesión en el sistema para buscar al Propietario indicado. 2. El Propietario responde a las cuestiones del Fiscalizador y verifica datos registrados junto con la documentación legal. 3. El Fiscalizador brinda opciones de decisión al Propietario. 4. El Propietario decide si sigue siendo infractor o si subsanara la infracción, y suscribe un compromiso. 5. Por cualquier opción se plantea una nueva inspección.
Actores: Inspector, Fiscalizador	
Objetivos: Llegar a un convenio con el Propietario, siguiendo normas legales.	
Descripción: El Fiscalizador realiza una entrevista al Propietario, identificando la infracción y la sanción a cancelar. El Propietario decide si continúa en estado de infractor o si subsanara la infracción. De cualquier forma se plantea una nueva inspección.	
Precondiciones: Segunda inspección a la vivienda, mas la Notificación al Propietario.	
Postcondiciones: Decisión de iniciar proceso legal o esperar rectificación del Propietario.	Alternativas: En el paso 3, si el Propietario decide ser infractor, el Fiscalizador inicia proceso legal, si decide subsanar infracción, el Fiscalizador establece un plazo.

Tabla 3.13 Caso de Uso expandido - Seguimiento del Proceso

Fuente: [Elaboración Propia]

Caso de Uso: Emitir reportes	Flujo de Sucesos: 1. El Inspector y/o Fiscalizador inician sesión en el sistema. 2. Ingresan en el enlace de 'Reportes' y seleccionan los datos de entrada. 3. Imprimen los reportes en formato de acuerdo al requerimiento. 4. El Inspector y el Fiscalizador verifican reportes. 5. El Inspector y/o Fiscalizador cierran sesión del sistema.
Actores: Fiscalizador	
Objetivos: Generar información precisa del trabajo realizado durante un determinado periodo de tiempo.	
Descripción: El Inspector y/o Fiscalizador imprimen informe de fiscalización, el cual contiene porcentajes de recaudaciones por sanciones a los infractores.	
Precondiciones: Existencia de la información necesaria en la base de datos, además de las formulas correspondientes para derivar dichos porcentajes.	
Postcondiciones: Impresión de los reportes.	Alternativas: En el paso 3, el formato de impresión de los reportes responde a las necesidades de los funcionarios.

Tabla 3.14 Caso de Uso expandido - Emitir Reportes

Fuente: [Elaboración Propia]

ANEXO C.

CARGO	FUNCIÓN	NRO. FUN.
Jefe de Unidad	Dirige la planificación, ejecución y control de los operativos de fiscalización que realiza esta unidad. También se encarga de coordinar actividades con las demás unidades.	1
Asesor Legal	Elabora informes legales y tributarios, tanto de los procesos iniciados como de los reclamos de los ciudadanos. Además, coopera legalmente con los demás empleados de la unidad.	1
Inspector Predial	Se encarga de la fiscalización técnica de los inmuebles o predios (casa, construcción, topología, etc.) de un determinado distrito.	4
Supervisor de Actividades Económicas	Administra el ingreso y egreso de los recursos económicos disponibles y/o asignados a dicha unidad.	1

Tabla 1.1 Tabla de Funciones de los Empleados de la Unidad de Fiscalización

Fuente: [Elaboración Propia]