

Universidad de las Ciencias Informáticas "Facultad 2"



Título: "Módulo Gestión de los Objetivos y los Responsables de un Cuadro de Mando Integral"

Trabajo de Diploma para optar por el título de
Ingeniero en Ciencias Informáticas

Autores:

Sulesky Rodríguez Estrada
Aliniuska Vazquez Tudela

Tutores:

Dr. Guillermo Ronda Pupo
Lic. Iván Hernández Ferrero

"Ciudad de la Habana"
"Julio del 2007"

DECLARACIÓN DE AUTORÍA

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo la presente a los ____ días del mes de _____ del año _____.

Aliniuska Vazquez Tudela

Sulesky Rodríguez Estrada

Firma de la autora

Firma de la autora

Dr. Guillermo Ronda Pupo

Lic. Iván Hernández Ferrero

Firma del tutor

Firma del tutor

AGRADECIMIENTOS

Autora: Sulesky Rodríguez Estrada.

Dicen que con la gloria se olvidan las memorias, yo no pienso hacer lo mismo, cuando alcanzamos las metas trazadas debemos tener en cuenta todas las personas que de una forma u otra nos ayudaron y apoyaron.

Yo quiero agradecer en primer lugar a mi padre, por ser la persona que más quiero en la vida, por todo su esfuerzo, apoyo y comprensión, por creer en mí y darme siempre fuerzas para seguir adelante, sin él no estaría hoy aquí y no sería lo que soy.

Quiero agradecer además a Grisel por ser tan especial conmigo, por todo el cariño que me ha dado, por brindarme confianza y seguridad en el éxito de este trabajo.

A mis queridas amigas Iris y Yanela, por ocupar un lugar especial durante estos últimos cinco años compartiendo conmigo los buenos y malos momentos, gracias por soportarme, a Aine por ser también otra de mis grandes amigas, por estar junto a mí desde pequeñas y por estar siempre para ayudarme.

A todos mis amigos, sin ellos la vida no es nada, gracias por su preocupación.

A mi compañera de tesis, por trabajar juntas y llegar hasta el final de este largo camino, al resto de mis compañeros de tesis del proyecto en especial a Yasser por todo su esfuerzo y sacrificio por ayudarnos.

A Michel por estar siempre cerca de mí y por darme cariño.

A mis abuelos, mis tíos y tías, todos mis primos, a mi familia en general por ser tan unida, por preocuparse por mí y por quererme como soy, a mi tía Rae que aunque esté lejos me ayudó muchísimo, a Gregory y Yordanis por su preocupación.

A Iván por haber sido el tutor de este trabajo.

A mis compañeros de clases y a todos los que me preguntaron alguna vez por mis estudios.

En fin quiero agradecer de todo corazón a todos aquellos que me han apoyado durante este largo andar.

Autora: Aliniuska Vazquez Tudela.

A mis padres porque siempre me han apoyado en todo y han confiado plenamente en mi.

A mis hermanas porque al estar en mis pensamientos me impulsan a seguir adelante.

A mi tía Zoraida y a Juan Luis por las veces que los he molestado y los favores que me han hecho.

A Yeni por ser como otra hermana y porque siempre he contado con ella en los buenos y malos momentos.

A mi primo Gustavo, a Yamilet, a Milerkys y demás amistades por los momentos agradables que he tenido junto a ellos.

A mi compañera de tesis porque la carga de trabajo ha sido menos tediosa.

A mis compañeros de proyecto en especial a Yasser porque siempre han estado dispuestos a ayudarme con cualquier duda de la tesis.

A mis compañeros de aula y de apartamentos por los buenos y malos momentos que me servirán de experiencia en el futuro.

A Rosa por acogerme como otra hija más.

A Chino, Carmen y demás familiares por las veces que compartí con todos ellos.

A Iván por haber sido el tutor de este trabajo.

A los hermanos de la iglesia que han orado para que salga bien en la tesis.

A todas esas personas que de una forma u otra me han apoyado y ayudado durante estos 5 cursos.

DEDICATORIA

Autora: Sulesky Rodríguez Estrada.

Quiero dedicar este trabajo especialmente a mi madre, que aunque ya no esté junto a mí, ha sido siempre el punto de partida de mi motivación en esta carrera, por educarme y mostrarme siempre el camino correcto.

Lo dedico también a mi padre y a Grisel, a todos mis amigos y a mi inmensa familia.

Y por supuesto me lo dedico a mí, por el placer de aprender.

Autora: Aliniuska Vazquez Tudela.

A mis padres porque son el mayor tesoro que tengo, porque gracias a ellos soy la persona que soy y los quiero mucho.

A mis hermanas porque las quiero mucho y siempre las tengo presentes en mis pensamientos.

A mis familiares que han confiado en que este sueño se pudiera realizar.

A mis amistades que me han dado su apoyo cuando los he necesitado.

Pensamiento

“En el mismo hombre suelen ir unidos un corazón pequeño y un talento grande. Pero todo hombre tiene el deber de cultivar su inteligencia, por respeto a sí mismo y al mundo. ...”

José Martí.

RESUMEN

El presente trabajo pretende diseñar un modelo para una aplicación web que permita la integración de los niveles estratégico, táctico y operativo de la gestión empresarial cubana a través de la gestión de los objetivos y los responsables de las perspectivas de un Cuadro de Mando Integral.

En estos momentos Cuba posee pocos CMI implementados, herramienta muy útil para la toma de decisiones en la dirección de las empresas, este sistema administrativo va más allá de las expectativas con la que los gerentes acostumbran evaluar la marcha de una empresa.

Se diseñará con carácter sistémico y funcional en organizaciones en el país un sistema automatizado del Cuadro de Mando Integral que además dará la posibilidad de la generalización de su aplicación en diversas organizaciones, lo cual mejorará el desempeño de la dirección estratégica al elevar el nivel de efectividad al emplearse dicho sistema.

Para desarrollar el sistema informático que mejore la consolidación de los procesos de la gestión de los objetivos y los responsables se siguió una metodología basada en las nuevas técnicas de la informática, cumpliendo con las necesidades empresariales. Se espera en primer lugar que la consolidación de los procesos de dicha gestión gane en rapidez y eficiencia. Los resultados más relevantes lo constituyen: la existencia de un producto informático que juega un papel importante en las distintas empresas, el logro de un flujo más rápido y seguro de información sensible y un producto con facilidades para migrar a otras plataformas.

ÍNDICE

INTRODUCCIÓN.....	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.....	4
1.1 Introducción.....	4
1.2 ¿Qué es Cuadro de Mando Integral?.....	4
1.2.1 Surgimiento del Cuadro de Mando Integral.....	6
1.2.2 Perspectivas del Cuadro de Mando Integral.....	7
1.2.2.1 Perspectiva financiera.....	7
1.2.2.2 Perspectiva Cliente.....	8
1.2.2.3 Perspectiva Procesos Internos.....	9
1.2.2.4 Perspectiva de Aprendizaje y Crecimiento.....	11
1.3 Sistemas automatizados existentes.....	11
1.4 Necesidades del Cuadro de Mando Integral en Cuba.....	14
1.5 Diseño metodológico de la investigación.....	15
1.6 Tendencias y Tecnologías actuales.....	15
1.6.1 Las aplicaciones Web.....	15
1.6.2 Lenguajes de programación para la Web.....	17
1.6.2.1 PHP 5.1.....	17
1.6.2.2 Java Script.....	18
1.6.3 Sistemas de Gestión de Base de Datos.....	19
1.6.3.1 MySQL 5.0.....	20
1.6.4 Servidor Web Apache 2.2.....	20
1.6.5 Modelo Vista Controlador.....	21
1.6.6 Proceso de Desarrollo.....	22
1.6.6.1 RUP.....	23
1.6.6.2 UML.....	24
1.6.7 Herramienta utilizadas.....	24
1.6.7.1 Rational Rose.....	24
1.6.7.2 Diseño de interfaz: Dreamweaver 8.0.....	25
1.6.7.3 Zend Studio 5.1.....	26
1.6.7.4 SQL Manager for MySQL 2005 versión 3.7.0.1.....	27
1.6.7.5 Framework Code Igniter 1.5.....	27
1.6.7.6 Framework AJAX: YUI.....	28
1.7 Conclusiones.....	28
CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA.....	29
2.1 Introducción.....	29
2.2 Objeto de Estudio.....	29
2.2.1 Problema y situación problemática.....	29
2.2.2 Objeto de automatización.....	29
2.2.3 Información que se maneja.....	30
2.3 Propuesta de sistema.....	30
2.4 Modelo de negocio.....	30
2.4.1 Actores del negocio.....	31
2.4.2 Trabajadores del negocio.....	31
2.4.3 Diagrama de Casos de Uso del Negocio.....	32

2.4.4 Descripción textual de los casos de uso del negocio	32
2.5 Diagrama de actividades.....	32
2.6 Diagrama de clases del modelo de objeto.....	32
2.7 Especificación de los requisitos de software	33
2.7.1 Definición de los requerimientos funcionales	33
2.7.2 Definición de los requerimientos no funcionales	34
2.8 Definición de los casos de usos.....	35
2.8.1 Definición de los actores.....	35
2.8.2 Diagrama de Casos de Uso del Sistema.....	35
2.8.3 Descripción textual de los casos de uso del sistema	36
2.9 Conclusiones.....	36
CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA	37
3.1 Introducción.....	37
3.2 Diagramas de clases del análisis.....	37
3.3 Diagramas de clases del diseño.....	39
3.4 Diagramas de interacción del diseño.....	44
3.5 Descripción de las clases.....	44
3.6 Diseño de la Base de datos	44
3.6.1 Modelo físico de datos.....	44
3.7 Descripción de las tablas de la Base de datos	46
3.8 Definiciones de diseño que se apliquen.....	46
3.8.1 Patrones.....	46
3.9 Tratamiento de errores.....	48
3.10 Seguridad.....	48
3.11 Interfaz de usuario.....	49
3.12 Concepción de la ayuda.....	49
3.13 Estándares de diseño.....	50
3.14 Conclusiones.....	52
CAPÍTULO 4. IMPLEMENTACIÓN	53
4.1 Introducción.....	53
4.2 Modelo de Despliegue	53
4.3 Diagrama de componentes.....	54
4.4 Conclusiones.....	55
CAPÍTULO 5: ESTUDIO DE FACTIBILIDAD	56
5.1 Introducción.....	56
5.2 Planificación basada en casos de uso.....	56
5.2.1 Identificar los puntos de casos de usos desajustados.....	56
5.2.2 Cálculo de UAV.....	56
5.2.3 Cálculo de UUCW.....	57
5.3 Ajustar los puntos de casos de uso.....	57
5.3.1 Cálculo de TCF.....	57
5.3.2 Cálculo de EF.....	59
5.4 Calcular esfuerzo FT implementación.....	60
5.4.1 Cálculo de CF.....	60
5.5 Esfuerzo de todo el proyecto	60

5.5 Costo del Proyecto.....	61
5.6 Beneficios tangibles e intangibles. Debe cuantificarse los tangibles.....	61
5.7 Análisis de costo.....	62
5.8 Conclusiones.....	62
CONCLUSIONES.....	63
RECOMENDACIONES.....	64
BIBLIOGRAFÍA.....	65
REFERENCIAS BIBLIOGRÁFICAS.....	69
ANEXOS.....	70
Anexo 1. Descripción textual de los casos de uso del negocio.....	70
Anexo 2. Diagramas de actividades.....	73
Anexo 3. Modelo de objetos.....	75
Anexo 4. Descripción textual de los casos de uso del sistema.....	76
Anexo 5. Diagramas de interacción del diseño.....	86
Anexo 6. Descripción de las clases.....	92
Anexo 7. Descripción de las tablas de la Base de datos.....	95
GLOSARIO DE TÉRMINOS.....	98

INTRODUCCIÓN

Dentro del entorno económico actual signado no sólo por la globalización, y los rápidos cambios tecnológicos y culturales, además de la cada vez más intensa explotación de los recursos escasos, las empresas se ven en la obligación de mejorar sus procesos día a día para afianzar su competitividad, satisfaciendo de la mejor forma a los usuarios con sus productos y servicios, y hacer un uso más eficiente de los recursos. Es por eso que éstas necesitan para monitorear el buen funcionamiento y capacidad de sus procesos, como así también sus continuos niveles de mejora, contar con un Cuadro de Mando Integral que permita no sólo hacer conocer los planes estratégicos y operativos a los diversos niveles de la organización y etapas de los procesos, sino que además permita informarse a los diversos niveles acerca de lo que está ocurriendo en el lugar de trabajo.

El Cuadro de Mando Integral (CMI) es una herramienta muy útil para la dirección de empresas en el corto y en el largo plazo. En primer lugar, porque al combinar indicadores financieros y no financieros permite adelantar tendencias y realizar una política estratégica proactiva. En segundo lugar, porque ofrece un método estructurado para seleccionar los indicadores guías que implica a la dirección de la empresa.

Un CMI se caracteriza por las cuatro perspectivas: financiera, de clientes, interna y de aprendizaje y crecimiento. Estas perspectivas son las más comunes, porque son aplicables en un gran número de empresas para organizar el modelo de negocio y estructurar los indicadores y la información.

Actualmente Cuba posee pocos CMI como herramienta para la gestión empresarial, existen ejemplos del uso del CMI en Cuba, como el GET Varadero, ETECSA y el de la Empresa de Servicios de Protección de Cienfuegos, SEPSA, en ambos casos se han obtenido resultados satisfactorios, no obstante, esto no es suficiente, pues la mayor aspiración es que todas las empresas cubanas puedan hacer uso del mismo siempre que así lo requieran. A nivel nacional no existe gran información relacionada con el uso de los CMI y por tanto existen situaciones en las que no se toman las mejores decisiones, las cuales son muy necesarias para saber el enfoque hacia donde debe estar dirigida la empresa para un mejor desarrollo, es por eso que esta investigación surge como necesidad de dar solución a esta situación por lo que el **problema científico** es *¿cómo ayudar al control de las empresas cubanas mediante el módulo de gestión de los objetivos y los responsables?*

En las empresas cubanas la gestión de los objetivos y responsables se torna algo engorrosa y provocan en ocasiones errores que dañan el buen funcionamiento de la empresa así como pérdida de la

información pues al no estar accesible la misma el personal autorizado no puede tener un mejor control de todo el proceso.

Por lo anteriormente planteado se define que el **objeto de la investigación** es *el Cuadro de Mando Integral (CMI) como herramienta de estudio en las empresas* y el **campo de acción** se enmarcaría *en los procesos de la gestión de los objetivos y responsables de las perspectivas de un Cuadro de Mando Integral en las empresas cubanas.*

El **objetivo general** es: *automatizar el módulo gestión de los objetivos y responsables de las perspectivas del Cuadro de Mando Integral para apoyar el servicio empresarial cubano.*

Para cumplir con el objetivo planteado se trazarán las siguientes **tareas de la investigación**:

- Realizar un estudio detallado de los CMI ya implementados en otros países.
- Buscar información para un mejor entendimiento y desarrollo de la gestión de los objetivos y los responsables de las perspectivas del Cuadro de Mando Integral.
- Investigar sobre las diferentes tecnologías a emplear para el desarrollo de este sistema automatizado.
- Seleccionar la metodología de Análisis y Diseño de sistemas informáticos, que facilite y garantice la creación con calidad del sistema.

Aportes desde el punto de vista práctico.

Se diseña y fundamenta un sistema automatizado de la gestión de los objetivos y los responsables de los procesos en las diferentes perspectivas del Cuadro de Mando Integral con carácter sistémico y funcional en organizaciones en el país.

Da la posibilidad de una generalización de su aplicación en diversas organizaciones, lo cual contribuirá a elevar el nivel de efectividad de la implantación de la dirección estratégica conllevando al mejoramiento del desempeño de las mismas.

Aporta una mayor rapidez en las búsquedas de información por parte del personal de apoyo disminuyendo su carga de trabajo.

Aportes desde el punto de vista metodológico.

El sistema conceptual y metodológico del modelo se emplea como material de consulta para el desarrollo de las estrategias de organizaciones, así como será empleado en la impartición de cursos en postgrados y diplomados.

Novedad Científica de la tesis.

Diseño de un modelo específico de dirección estratégica, que considera la integración de los niveles estratégico, táctico y operativo a partir de la fusión de varias herramientas de dirección.

Determinación de las regularidades, principios y premisas que rigen el proceso de implantación y ejecución de la dirección estratégica de una organización de seguridad y protección.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

1.1 Introducción.

En este capítulo se da una idea general de los temas de interés para comprender el objeto de estudio de esta investigación, como la conceptualización de un Cuadro de Mando Integral, las distintas perspectivas que posee y los procesos vinculados a estas. Además se aborda sobre los sistemas existentes vinculados al campo de acción, tendencias y tecnologías actuales seleccionadas a emplear en el desarrollo de la propuesta que son las elegidas para dar solución al objetivo propuesto de la investigación.

1.2 ¿Qué es Cuadro de Mando Integral?

El Cuadro de Mando Integral (CMI) es una herramienta muy útil para describir, aplicar y dirigir la visión y estrategia futura de una organización; así como es un sistema de medición que va más allá de la perspectiva financiera con la que los gerentes acostumbran evaluar la marcha de una empresa. Además muestra continuamente cuando una compañía y sus empleados alcanzan los resultados perseguidos por la estrategia. La misma surgió con la necesidad de beneficiar la implantación del Cuadro de Mando Integral, donde se pueden relacionar de manera definitiva la estrategia y ejecución empleando indicadores y objetivos en el corto, medio y largo plazo para la dirección de una determinada empresa; tener una herramienta de control que permita la toma de decisiones de manera ágil; comunicar la estrategia a todos los niveles de la organización consiguiendo así alinear a las personas con la estrategia; y tener una clara visión de las relaciones causa-efecto de la estrategia.

Como componentes básicos de un buen CMI están los siguientes:

- **Una cadena de relaciones de causa efecto:** Expresa el conjunto de hipótesis de la estrategia a través de objetivos estratégicos y su logro mediante indicadores de desempeño.
- **Un enlace a los resultados financieros:** Los objetivos del negocio y sus respectivos indicadores, deben reflejar la composición sistémica de la estrategia, a través de cuatro perspectivas: Financiera, Clientes, Procesos Internos, y Aprendizaje y Crecimiento. Los resultados deben traducirse finalmente en logros financieros que conlleven a la maximización del valor creado por el negocio para sus accionistas.
- **Un balance de indicadores de resultados (efecto) e indicadores guía (causa):** Además de los indicadores que reflejan el desempeño final del negocio, se requiere un conjunto de indicadores que reflejen las cosas que se necesitan "hacer bien" para cumplir con el objetivo (asociados a las

palancas de valor e indicadores guía- causa). Estos miden el progreso de las acciones que nos acercan o que propician el logro del objetivo. El propósito es canalizar acciones y esfuerzos orientados hacia la estrategia del negocio.

- **Mediciones que generen e impulsen el cambio:** Una de las premisas a las que hacen mención Kaplan y Norton es: la medición motiva determinados comportamientos, asociados tanto al logro como a la comunicación de los resultados organizacionales, de equipo e individuales. De allí que un componente fundamental es el de definir indicadores que generen los comportamientos esperados, particularmente aquellos que orienten a la organización a la adaptabilidad ante un entorno en permanente y acelerado cambio.
- **Alineación de iniciativas o proyectos con la estrategia a través de los objetivos estratégicos:** Cada proyecto que exista en la empresa debe relacionarse directamente con el apalancamiento de los logros esperados para los diversos objetivos expresado a través de sus indicadores.
- **Consenso del equipo directivo de la empresa u organización:** El Cuadro de Mando Integral es el resultado del diálogo entre los miembros del equipo directivo, para lograr reflejar la estrategia del negocio, y de un acuerdo sobre como medir y respaldar lo que es importante para el logro de dicha estrategia.

El CMI es por lo tanto un sistema de gestión estratégica de la empresa, que consiste en:

- Formular una estrategia consistente y transparente.
- Comunicar la estrategia a través de la organización.
- Coordinar los objetivos de las diversas unidades organizativas.
- Conectar los objetivos con la planificación financiera y presupuestaria.
- Identificar y coordinar las iniciativas estratégicas.
- Medir de un modo sistemático la realización, proponiendo acciones correctivas oportunas.

Ricardo Martínez Rivadeneira (profesor consultor de Gestión) plantea: “El Cuadro de Mando Integral es una forma integrada, balanceada y estratégica de medir el progreso actual y suministrar la dirección futura de la compañía que le permitirá convertir la visión en acción, por medio de un conjunto coherente de indicadores agrupados en cuatro diferentes perspectivas, a través de las cuales es posible ver el negocio en conjunto”. [1]

1.2.1 Surgimiento del Cuadro de Mando Integral.

Los orígenes del Cuadro de Mando Integral (CMI) o Balanced Scorecard (BSC), data de 1990, cuando el Instituto Nolan Norton, patrocinó un estudio de un año de duración sobre múltiples empresas: “La medición de los resultados en la empresa del futuro”. El estudio fue motivado por la creencia de que los enfoques existentes sobre la medición de la actuación, que dependían primordialmente de las valoraciones de la contabilidad financiera, se estaban volviendo obsoletos.

En los inicios del proyecto se examinaron estudios recientes sobre casos de sistemas innovadores de medición de la actuación. Una de las empresas-fabricantes describió un enfoque para medir la tasa de progreso de actividades de mejora continua. El caso mostró la forma en que esta entidad estaba utilizando un “Cuadro de mando corporativo”, de nueva creación, que además de varios indicadores financieros tradicionales, contenía mediciones de actuación relacionadas con los plazos de entrega a los clientes, la calidad y los tiempos de los ciclos de los procesos de fabricación, y la eficacia de los avances de los nuevos productos.

Durante la primera mitad del estudio se presentó una gran variedad de ideas, incluyendo el valor del accionista, mediciones de productividad y calidad, y nuevos planes de compensación, centrándose los participantes en un Cuadro de Mando multidimensional, ya que parecía ser lo más prometedor para sus necesidades.

Las discusiones del grupo condujeron a una expansión del Cuadro de Mando hasta llegar a lo que se denominó como un “Cuadro de Mando Integral”, organizado en torno a cuatro perspectivas muy precisas: la financiera, la del cliente, la interna, y la de innovación y formación. El nombre reflejaba el equilibrio entre objetivos a corto y largo plazo, entre medidas financieras y no financieras, entre indicadores provisionales e históricos, y entre perspectivas de actuación externas e internas.

Los líderes empresariales vieron al Cuadro de Mando como algo más que un sistema de mediciones. Querían utilizar el nuevo sistema para comunicar y alinear sus organizaciones con las nuevas estrategias; lejos del enfoque histórico y a corto plazo de reducción de costos y competencia a bajo precio, y hacia la generación de crecientes oportunidades, ofreciendo a los clientes productos y servicios con valor añadido y a medida.

Norton y Kaplan describieron la importancia de elegir indicadores basados en el éxito estratégico, publicado en septiembre- octubre de 1993 en el artículo “Cómo poner a trabajar al Cuadro de Mando Integral”, donde veían al CMI como un vehículo para ayudar a la empresa a traducir y poner en práctica la

estrategia. Por tanto, el Cuadro de Mando Integral había evolucionado de un sistema de indicadores mejorado, para convertirse en un sistema de gestión central.

1.2.2 Perspectivas del Cuadro de Mando Integral.

El Cuadro de Mando Integral usa un modelo de negocio basado y estructurado en indicadores y objetivos que giran en torno a cuatro perspectivas: financiera, clientes, procesos internos, aprendizaje y crecimiento. Estas perspectivas son las más comunes, porque son aplicables en un gran número de empresas. Ver Figura 1.1.

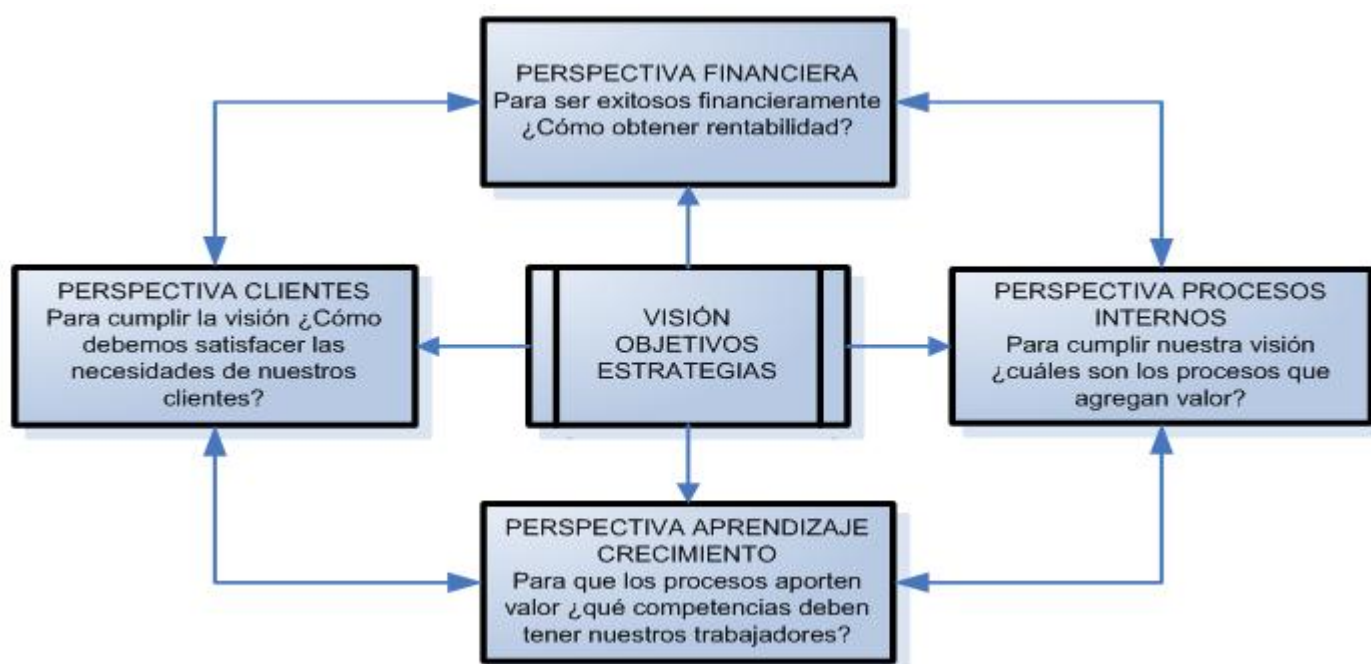


Figura 1.1 Las perspectivas del Cuadro de Mando Integral

1.2.2.1 Perspectiva financiera.

Los objetivos financieros representan los objetivos a largo plazo de la organización: proporcionar rendimientos superiores basados en el capital invertido. El Cuadro de Mando Integral puede hacer que los objetivos financieros sean explícitos, y que se adapten a la medida de las unidades de negocio, en las diferentes fases de su crecimiento y ciclo de vida. El cuadro de mando utiliza los objetivos financieros en relación al aumento e intensificación de la rentabilidad, los rendimientos de los activos y los ingresos.

Permaneciendo dentro de la perspectiva financiera, el cuadro de mando permite que la alta dirección de las unidades de negocio especifique, no solo como se evaluará el éxito de la empresa a largo plazo, sino las variables que se consideran más importantes para la creación y el impulso de los objetivos de resultados a largo plazo. Los inductores de la perspectiva financiera serán ajustados a medida del sector económico, el entorno competitivo y la estrategia de la unidad de negocio. Las empresas pueden elegir estrategias financieras entre los temas relacionados con el crecimiento de los ingresos, la reducción de costos y la utilización de los activos.

Fases del ciclo de vida de la perspectiva financiera.

Se ha descubierto que, para cada uno de los tres objetivos de crecimiento, sostenimiento y recolección, existen tres estrategias financieras que impulsan la estrategia maestra:

- Crecimiento y diversificación de los ingresos.
- Reducción de costes / mejora de la productividad.
- Utilización de los activos / estrategia de inversión.

El crecimiento y la diversificación de los ingresos implican la expansión de la oferta de producción y servicios para que se conviertan en una oferta de mayor valor añadido, y cambiar los precios de dichos productos y servicios. El objetivo de reducción de los costos y mejora de la productividad supone esfuerzos para rebajar los costos directos de la producción y los servicios, reducir los costos indirectos y compartir los recursos comunes con otras unidades de negocios. Por lo que respecta al tema de la utilización de los activos, los directivos intentan reducir los niveles de capital circulante que se necesitan para apoyar a un volumen o una diversidad del negocio dada.

1.2.2.2 Perspectiva Cliente.

Para lograr el desempeño financiero que una empresa desea, es fundamental que posea clientes leales y satisfechos, con ese objetivo en esta perspectiva se miden las relaciones con los clientes y las expectativas que los mismos tienen sobre los negocios. Además, en esta perspectiva se toman en cuenta los principales elementos que generan valor para los clientes, para poder así centrarse en los procesos que para ellos son más importantes y que más los satisfacen.

El conocimiento de los clientes y de los procesos que más valor generan es muy importante para lograr que el panorama financiero sea próspero. Sin el estudio de las peculiaridades del mercado al que está enfocada la empresa no podrá existir un desarrollo sostenible en la perspectiva financiera, ya que en gran

medida el éxito financiero proviene del aumento de las ventas, situación que es el efecto de clientes que repiten sus compras porque prefieren los productos que la empresa desarrolla teniendo en cuenta sus preferencias.

Los directivos han de identificar que clientes en los segmentos seleccionados, valoran y eligen la propuesta de valor que ellos entregaran a estos clientes. Pueden seleccionar objetivos y medidas de entre las tres clases de atributos que, si se satisfacen, permitirán que la empresa retenga y amplíe sus negocios con estos clientes seleccionados. Las tres clases de atributos son:

- Atributos de producto y servicio: funcionalidad, calidad y precio.
- Relaciones con los clientes: calidad de la experiencia de compra y relaciones personales.
- Imagen y reputación.

Al seleccionar objetivos e indicadores concretos pertenecientes a estas tres categorías, los directivos pueden encarrilar a su organización hacia la entrega a sus segmentos de clientes seleccionados de una propuesta de valor superior.

Existen cinco tipos de estrategias que son los más eficaces, según Kaplan y Norton, para la medición que encerrarían los objetivos genéricos para esta perspectiva. Estos serían:

1. La cuota de mercado.
2. El incremento de clientes.
3. La satisfacción del cliente.
4. La rentabilidad del cliente.
5. El indicador de las propiedades valor del cliente.

Estos indicadores por ser tan generales tendrán que ser adaptados a cada entidad en cuestión con respecto a las expectativas que ellas tengan hacia sus segmentos de mercado.

1.2.2.3 Perspectiva Procesos Internos.

En la perspectiva del proceso interno, los directivos identifican los procesos críticos en los que deben sobresalir con excelencia si es que quieren satisfacer los objetivos de los segmentos de accionistas y de clientes seleccionados.

Un avance reciente ha sido la incorporación del proceso de innovación como un componente vital de la perspectiva del proceso interno. El proceso de innovación subraya la importancia de la identificación de las características de los segmentos de mercado que la organización desea satisfacer con sus productos y

servicios futuros, y de diseñar y desarrollar los productos y los servicios que en el futuro satisfagan a esos segmentos seleccionados. Este enfoque permite que la organización ponga un énfasis considerable en los procesos de investigación, diseño y desarrollo que dan como resultado nuevos productos, servicios y mercados.

El proceso operativo sigue siendo importante, y las organizaciones deben identificar las características de costo, calidad, tiempo y actuación que permitirán entregar productos y servicios superiores a sus clientes existentes. El proceso de servicio posventa permite a la empresa presentar, cuando es conveniente, aspectos importantes del servicio que se ofrecen después de que el producto o servicio adquirido ha sido entregado al cliente.

Además esta perspectiva identifica aquellos pocos procesos que se espera que tengan el mayor impacto sobre la estrategia. Por ejemplo una empresa puede aumentar sus inversiones en Investigación y Desarrollo (I+D) y re estructurar sus procesos de desarrollo de producto de manera que pueda obtener productos innovadores y de alto rendimiento para sus clientes.

Cada negocio tiene un conjunto único de procesos para crear valor para los clientes y producir resultados financieros. Sin embargo, se ha descubierto que un modelo genérico de cadena de valor proporciona una plantilla que las empresas pueden hacer a su medida, al preparar su perspectiva del proceso interno. (Ver figura 1.2).

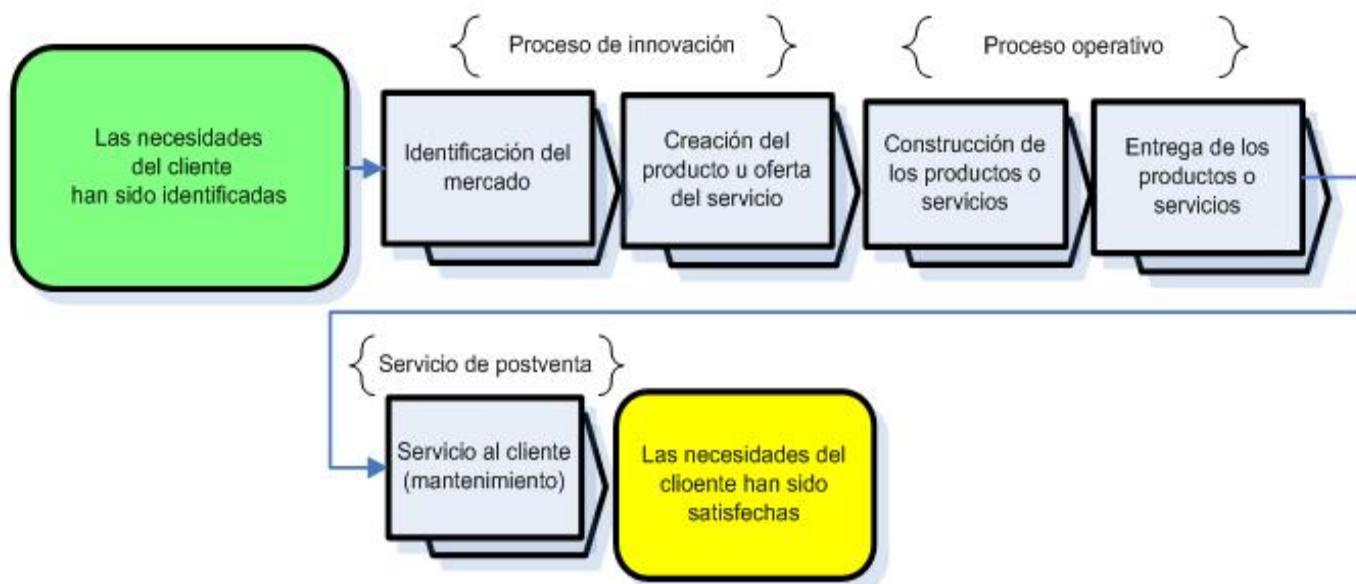


Figura 1.2. La cadena de valor de la perspectiva del proceso.

1.2.2.4 Perspectiva de Aprendizaje y Crecimiento.

La capacidad de alcanzar las ambiciosas metas de los objetivos financieros, del cliente y de los procesos internos depende de la capacidad de crecimiento y aprendizaje de la organización. Esta perspectiva es el motor impulsor de las anteriores perspectivas del cuadro de mando y refleja los conocimientos y habilidades que la empresa posee tanto para desarrollar sus productos como para cambiar y aprender. Los inductores de crecimiento y aprendizaje provienen primordialmente de tres fuentes: los empleados, los sistemas y la equiparación de la organización. Las estrategias para una actuación superior exigirán, en general, unas inversiones importantes en personal, sistemas y procesos que construyen capacidades para la organización.

Los objetivos financieros, de clientes y de procesos internos del CM I revelarán grandes vacíos entre las capacidades existentes de las personas, los sistemas y procedimientos; al mismo tiempo mostrarán que será necesario para alcanzar una actuación que represente un gran adelanto. Para llenar estos vacíos, las empresas tendrán que invertir en la recalificación de empleados, potenciar los sistemas y tecnología y coordinar los procedimientos y rutinas de la organización. La perspectiva de formación (o aprendizaje) y crecimiento permite que la empresa se asegure su capacidad de renovación a largo plazo, un requisito previo para una existencia duradera

La cuarta y última perspectiva del Cuadro de Mando Integral desarrolla objetivos e indicadores para impulsar el aprendizaje y crecimiento de la organización. Los objetivos establecidos en las perspectivas financiera, del cliente y de los procesos internos identifican los puntos en que la entidad ha de ser excelente. Los objetivos de la perspectiva de aprendizaje y crecimiento proporciona una infraestructura que permite que se alcancen objetivos ambiciosos en las restantes tres perspectivas.

1.3 Sistemas automatizados existentes

En el ámbito internacional existen varios tipos de software que ayudan a informar y evaluar la implantación del cuadro de mandos integrales en una organización. No así en nacional, y debido a esta problemática se realizó un estudio previo a distintos software, para lograr implantar el sistema que se propone en esta investigación; los cuales son los siguientes:

Decide Soft: La mayoría de los paquetes que existen en el mercado están en inglés, siendo éstos una de las pocas excepciones. Conocido como el primer software en español para la completa realización de Cuadros de Mando Integral.

Decide Soft es un producto creado, diseñado y comercializado por dos empresas compostelanas:

- Dimensiona Consultoría y Tecnológica: empresa dedicada al desarrollo de aplicaciones informáticas punteras (sistemas de traducción automática y sistemas de publicación Web).
- Ferreiro/Boullón & Asociados: empresa declarada IEBTS (de base tecnológica) dedicada a la provisión de soluciones para la toma de decisiones empresariales, especialmente en el área sanitaria.

Delphos: Es un software de Control de Gestión que permite implementar íntegramente un Cuadro de Mando Integral (Balanced ScoreCard), un Plan Estratégico, un Plan Anual Operativo o cualquier otro modelo que se requiera para controlar las operaciones de cualquier tipo de organización, e inclusive mejorar el desempeño y la productividad.

Fue implantado en Costa Rica en la década de los 90, e incluye algunos países de: Centroamérica, Suramérica y Norteamérica. Su principal beneficio es que permitir desarrollar una administración orientada a resultados, mejorando y controlando completamente el ámbito de acción de cada uno de los miembros de la organización, sea pública o privada.

Delphos integra en una sola aplicación:

- Administración estratégica (objetivos, indicadores y responsables).
- Administración de Proyectos.
- Visualización de información mediante cubos o modelos multidimensionales.
- Graficador, Reporteador, Presupuesto Gerencial.

Los cuales tienen conectividad universal, por lo que pueden ser alimentados automáticamente desde cualquier base de datos. Delphos está diseñado para profesionales NO informáticos, esto garantiza que con una pequeña capacitación de su uso y aplicación, el cliente obtendrá el máximo provecho.

Dialogy Strategy: Es el software de Cuadro de Mando Integral o Balanced Scorecard ideal para aquellas empresas que quieren crecer, que han entendido que a una administración detallada deben agregarle herramientas que les permitan medir su estrategia y desempeño de manera sencilla.

Este software permite definir una serie de indicadores, objetivos estratégicos, enlazarlos en una relación de causa-efecto, asignar las medidas apropiadas, introducir datos y comenzar a monitorizar el rendimiento de su compañía.

Además Dialog Strategy es un software gratuito para desarrollo de Cuadro de Mando Integral. Sus versiones están disponibles en MS SQL Server y Access 2000. Posee la posibilidad de ser entregado en

inglés, español o noruego, según el deseo de los clientes. Es multiusuario, multizona, multidivisión y multiproducto. Interfases intuitiva y amigable.

Requiere de estos mínimos privilegios:

- Windows 98, ME, 2000, NT or XP.
- Min. 64MB RAM.
- 25MB de espacio de disco libre.

Algunos de los ejemplos del uso de Cuadro de Mando Integral en Cuba se pueden encontrar en:

Caso GET Varadero: El diseño e implementación del Cuadro de Mando Integral en el GET Varadero, unido a la aplicación de otras herramientas complementarias, le permitió al equipo de dirección observar globalmente la marcha de los aspectos clave de la gestión de la empresa, lo que ayudó a corregir oportunamente las desviaciones con tendencia a alterar las metas u objetivos previstos, mediante un control por excepción a ciertas áreas clave seleccionadas cuidadosamente, las que por medio de señales rápidas de alerta permiten identificar las desviaciones, con sus causas y responsables, ligadas al nivel inmediato inferior de su propia función directiva. La información debe obtenerse con rapidez para poder actuar oportunamente.

La empresa de Servicios de Protección de Cienfuegos, SEPSA: En su primer año, el CMI se ha utilizado como una herramienta de control, comunicación y retroalimentación. Ha creado una auténtica intranet corporativa utilizando técnicas del Cuadro de Mando Integral, así como condiciones para efectuar el control estratégico de la misma. En esta empresa predomina el análisis holístico gracias a esta herramienta. La comunicación dentro de la organización ha mejorado ostensiblemente, al igual que la toma de decisiones. Mediante su control operativo y corto efectivo el CMI ha ayudado indiscutiblemente a impulsar la gestión del aprendizaje en SEPSA Cienfuegos.

CMI en ETECSA: La empresa telefónica ha usado el CMI en la unidad de negocio internacional (UNI). Tiene implementada una aplicación que automatiza el control de las llamadas internacionales en cuanto por ejemplo a la facturación en minutos al mes o la duración de las llamadas en total, el importe percibido, los gastos incurridos y la cantidad de llamadas.

1.4 Necesidades del Cuadro de Mando Integral en Cuba.

El Cuadro de Mando Integral se está expandiendo a las empresas cubanas, las cuales se han ido adaptando a los cambios ocurridos en el contexto internacional. Muchas de las organizaciones cubanas se han apropiado de las normas de calidad ISO 9000 y la están aplicando con resultados satisfactorios.

En Cuba se ha reflejado un despertar en este sentido. El mismo ha sido declarado en la Resolución Económica del V Congreso del Partido Comunista de Cuba, el cual plantea lo siguiente:

"El empleo de técnicas modernas de dirección empresarial, adecuadas a nuestra características y basadas en las mejores y más avanzadas prácticas contemporáneas..."[2]

Todo este proceso demuestra una brecha abierta hacia el reconocimiento de la necesidad de un cambio en el sistema empresarial cubano hacia una búsqueda de mejores desempeño.

El modelo de Perfeccionamiento Empresarial ha contribuido al desarrollo de las organizaciones. El mismo se encamina a potenciar de manera continua los niveles de eficiencia, autoridad y ejecutividad de la empresa estatal cubana como eslabón fundamental de la economía.

Este proceso hace que el Cuadro de Mando Integral sea un camino importante en el logro de los objetivos organizacionales porque presupone un cambio en la concepción de la forma y pensar que puede llevar a las organizaciones a un nuevo estado.

En la sociedad cubana se busca la competitividad de las organizaciones a través de la creación de bienes y servicios para la satisfacción siempre creciente de la sociedad cubana.

El objetivo de las organizaciones tanto lucrativas como sin fines de lucro es obtener beneficio, estableciendo un ambiente en el que las personas puedan sentirse útiles a la sociedad y contribuir con sus capacidades a la obtención de bienes y servicios que conducen al bienestar social.

Una organización, tanto con fines de lucro como no lucrativos, es un sistema coordinado conscientemente que tiene el fin de conseguir cada vez más beneficios sobre la base de aportar bienes y servicios a la sociedad con el fin de satisfacer las necesidades, deseos de un conjunto de personas que integran el mercado objetivo.

Cada vez es más creciente las entidades de la isla que muestran interés por la herramienta de gestión empresarial, dada a conocer por Kaplan y Norton en 1992.

1.5 Diseño metodológico de la investigación.

Se utilizaron los siguientes métodos teóricos porque permiten estudiar las características del objeto de investigación que no son observables directamente, facilitan la construcción de modelos de investigación y crean las condiciones para ir más allá de las características fenomenológicas y superficiales de la realidad, contribuyendo al desarrollo de las teorías científicas y para su ejecución se apoyan en el proceso de análisis y síntesis.

Analítico – sintético.

Se usa este diseño metodológico para el análisis de fuentes, donde nos ayuda en la búsqueda de la esencia, rasgos que caracterizan y distinguen a los Cuadro de Mando Integral, la caracterización del sistema de planeación de las gerencias estudiadas y demás temáticas relacionadas con el desarrollo de la investigación; así como para establecer tendencias e insuficiencias en el campo de acción y el objeto de estudio.

Modelación.

Se usa este diseño metodológico para el diseño del sistema que se propone, pues se utilizan símbolos para designar las propiedades del sistema real de un CMI y tiene la capacidad de representar las características y relaciones fundamentales de dicho sistema. De lo abstracto a lo concreto para valorar las insuficiencias de los modelos de dirección estratégica estudiados.

1.6 Tendencias y Tecnologías actuales.

Se hizo un análisis de las tendencias y tecnologías actuales posibles a emplear, considerando las situaciones de este trabajo. Las mismas se explican a continuación.

1.6.1 Las aplicaciones Web.

El desarrollo de aplicaciones Web está siendo utilizado en muchas organizaciones, esta situación va ir creciendo indefinidamente. Es por ello que día a día se requieren más programadores capacitados para desarrollos basados en el World Wide Web (WWW).

Una aplicación web es una aplicación informática que los usuarios utilizan accediendo a un servidor Web a través de Internet o de una intranet. Las aplicaciones web son populares debido a la practicidad del navegador Web como cliente ligero. La habilidad para actualizar y mantener aplicaciones web sin distribuir e instalar software en miles de potenciales clientes es otra razón de su popularidad. Aplicaciones como los

webmails, wikis, weblogs, MMORPGs (**M**assive (**l**y) **M**ultiplayer **O**nline **R**ole-**P**laying **G**ames) y tiendas en línea son ejemplos bien conocidos de aplicaciones web.

Las aplicaciones Web son una especialización de las aplicaciones cliente-servidor, o sea, su arquitectura general es la de un sistema cliente/servidor, donde tanto el cliente (el navegador) como el servidor (el servidor Web), y el protocolo mediante el que se comunican (el HTTP: HyperText Transfer Protocol) son estándar, y no han de ser creados por el desarrollador.

La parte del cliente de las aplicaciones Web está formada por el código HTML (HyperText Markup Language) que forma la página Web, con opción a código ejecutable mediante los lenguajes script de los navegadores (JavaScript, VBScript, PerlScript) o mediante pequeños programas (applets) en Java. La parte del servidor está formada por un programa o script que es ejecutado por el servidor Web, y cuya salida se envía al navegador del cliente.

Las aplicaciones Web ofrecen grandes ventajas que pueden ser aprovechadas por muchas organizaciones. Entre ellas se pueden mencionar:

- *No requieren instalación*, pues usan tecnología Web, lo cual permite el aprovechamiento de todas las características del Internet.
- *Son fáciles de usar* (no requieren conocimientos avanzados de computación).
- *Alta disponibilidad*, ya que puede realizar consultas en cualquier parte del mundo donde tenga acceso a Internet y a cualquier hora.

No obstante a la serie de ventajas que presenta tiene además algunas desventajas, las cuales son:

- *Acceso limitado*, la necesidad de conexión permanente y rápida a Internet hacen que el acceso a estas aplicaciones no esté al alcance de todos.
- *La interactividad no se produce en tiempo real*, en las aplicaciones web cada acción del usuario conlleva un tiempo de espera excesivo hasta que se obtiene la reacción del sistema.
- *Elementos de interacción muy limitados*. En comparación con el software de escritorio, las posibilidades de interacción con el usuario que ofrecen las aplicaciones web (mediante formularios principalmente) son muy escasas.
- *Diferencias de presentación entre plataformas y navegadores*. La falta de estándares ampliamente soportados dificulta el desarrollo de las aplicaciones.

1.6.2 Lenguajes de programación para la Web.

Uno de los ejes fundamentales que diferencian a Internet de otros medios de comunicación es la interacción y personalización de la información con el usuario. Esto se logra por medio de lenguajes para programación Web que existen hoy en día. Los cuales se encuentran tanto del lado del servidor como del lado del cliente.

Entre los lenguajes del lado del servidor se pueden encontrar a PERL, ASP, PHP como los más sobresalientes por el auge que han tenido. Estos se caracterizan por desarrollar la lógica de negocio dentro del Servidor, además de ser los encargados del acceso a Bases de Datos, tratamiento de la Información, entre otras funciones.

Del lado del cliente se encuentran principalmente el JavaScript (JScript) y el Visual Basic Script (VBScript), que son los encargados de aportar dinamismo a la aplicación en los navegadores. En el caso del VBScript este es prácticamente usado a la hora de programar en ASP del lado del servidor, ya que su mayor desventaja radica en que sólo es soportado por el Navegador Web de su fabricante, Microsoft.

1.6.2.1 PHP 5.1

PHP es un lenguaje de programación usado generalmente para la creación de contenido para sitios Web. Además es un lenguaje interpretado de alto nivel embebido en páginas HTML y ejecutado en el servidor.

Ventajas de PHP.

- Es un *lenguaje multiplataforma*, funciona tanto para Unix (con Apache) como para Windows (con Microsoft Internet Information Server).
- *Capacidad de conexión* con la mayoría de los manejadores de base de datos que se utilizan en la actualidad, destacando su conectividad con MySQL.
- *Capacidad de expandir su potencial* utilizando la enorme cantidad de módulos (llamados ext's o extensiones).
- Posee una *amplia documentación* en su página oficial, entre la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda.
- *Es libre*, por lo que se presenta como una alternativa de fácil acceso para todos.
- Permite las técnicas de *Programación Orientada a Objetos*.
- Permite crear los *formularios* para la Web.
- Biblioteca nativa de *funciones* sumamente amplia e incluida.

- No requiere definición de *tipos de variables* ni manejo detallado del bajo nivel.

Los diseñadores de PHP5 han realizado un cambio radical en el tratamiento de las variables objeto: en PHP5 todas las variables que nombran objetos son en realidad referencias. No hay que usar el operador '&' ni en las asignaciones, ni en el paso de parámetros que son objetos, ahorrándose con ello gran cantidad de potenciales errores. El mismo introduce tres palabras clave (`public`, `private` y `protected`) que sustituyen a `var` en la definición de variables miembro --atributos-- de la clase, y que preceden a la definición de funciones miembro --métodos--. Para el uso de este lenguaje se cuenta con un framework, llamado Code Igniter, que ya brinda todas las funcionalidades necesarias para crear la aplicación Web esperada y la posibilidad de construir en base al patrón de arquitectura MVC (Modelo Vista Controlador), lo más usual en este tipo de aplicación.

Otros lenguajes como Perl (Practical Extraction and Report Language), ASP (Active Server Pages) y JSP (Java Server Pages) tienen características similares al PHP aunque poseen rasgos que los marcan y por ello los distingue, entre ellos podemos encontrar:

- Características multiplataformas: Menos el ASP, que es solamente soportado por la plataforma Windows, los demás lenguajes están soportados en múltiples plataformas.
- Velocidad de ejecución: La velocidad es mayor en PHP, seguidos por PERL y JSP.
- Disponibilidad de recursos: Actualmente los más utilizados en la Internet son el PHP y el JSP, siendo más utilizado en la publicación de artículos y códigos de ejemplos. PHP tiene una de las comunidades más grandes en Internet, al igual que la de Java.
- Familiaridad con el lenguaje: En la universidad los lenguajes más utilizados por los programadores es el ASP y el PHP.

De acuerdo a las anteriores comparaciones, más las posibilidades multiplataforma, rapidez y su interactividad con el sistema gestor de base de datos MySQL, el PHP resultó ser un lenguaje etiquetado para el desarrollo de la aplicación, particularmente en la versión PHP 5.1.

1.6.2.2 Java Script.

JavaScript es un lenguaje interpretado, es decir, que no requiere compilación, utilizado principalmente en páginas Web, con una sintaxis semejante a la del lenguaje Java y el lenguaje C. No es un lenguaje orientado a objetos propiamente dicho, ya que no dispone de herencia, es más bien un lenguaje basado

en prototipos, ya que las nuevas clases se generan clonando las clases base (prototipos) y extendiendo su funcionalidad.

La mayoría de los navegadores interpretan el código JavaScript integrado dentro de las páginas Web, ya que es muy sencillo.

1.6.3 Sistemas de Gestión de Base de Datos.

Entre los SGBD comúnmente utilizados en el mundo tenemos Oracle, Microsoft SQL Server, PostgreSQL, MySQL, entre otros. Todos estos presentan un enfoque relacional con un buen basamento matemático centrado en el Álgebra Relacional.

Los sistemas mencionados anteriormente facilitan el trabajo con la base de datos y tienen características que los diferencian, por ejemplo:

- Oracle: Su mayor defecto es su enorme precio, que es de varios miles de euros (según versiones y licencias). Otro aspecto que ha sido criticado por algunos especialistas es la seguridad de la plataforma, y las políticas de suministro de parches de seguridad, modificadas a comienzos de 2005 y que incrementan el nivel de exposición de los usuarios.[3] Como se mencionó anteriormente, su gran desventaja consiste en adquirir licencia para utilizarlo, mientras que MySQL es gratis.
- Microsoft SQL Server: Puede ser utilizado solamente con el sistema operativo Windows que está patrocinado por la compañía Microsoft, por lo que no es multiplataforma y posee licencias con costos muy elevados. Todo lo contrario de MySQL, que es multiplataforma y gratuito.
- PostgreSQL: Es un sistema de gestión de bases de datos Objeto-Relacionales (ORDBMS) libre. PostgreSQL tiene transacciones, integridad referencial, vistas, y multitud de funcionalidades, pero es lento y pesado. Por otro lado, la velocidad de respuesta que ofrece este gestor parece un poco deficiente. También resulta engorroso las pequeñas variaciones que presenta en algunos de los tipos de datos que maneja, siendo el problema más comentado el referente al tipo "serial". Sin embargo, MySQL en la versión 5.0 soporta transacciones e integridad referencial, con la diferencia de que este último es mucho más rápido. Por tanto, es este gestor de base de datos una buena opción.

1.6.3.1 MySQL 5.0

MySQL es la base de datos de código abierto (Open Source) más popular de Internet. Es un sistema de gestión de base de datos, multihilo, multiplataforma y multiusuario con más de seis millones de instalaciones. Este es fiable, fácil de usar y muy robusto.

Este desarrolla MySQL como software libre en un esquema de licenciamiento dual. Por un lado lo ofrece bajo la GNU GPL, pero, empresas que quieran incorporarlo en productos privativos pueden comprar a la empresa una licencia que les permita ese uso.

MySQL es muy utilizado en aplicaciones Web como MediaWiki o Drupal, en plataformas (Linux/Windows-Apache-MySQL-PHP/Perl/Python), y por herramientas de seguimiento de errores como Bugzilla. Su popularidad como aplicación Web está muy ligada a PHP, que a menudo aparece en combinación con MySQL.

Las principales características de este gestor de bases de datos son las siguientes:[4]

- Aprovecha la *potencia de sistemas multiprocesador*, gracias a su implementación multihilo.
- Soporta gran *cantidad de tipos de datos* para las columnas.
- *Dispone de API's* en gran cantidad de lenguajes (C, C++, Java, PHP, etc).
- *Gran portabilidad* entre sistemas.
- Soporta hasta 32 índices por tabla.
- *Gestión de usuarios y passwords*, manteniendo un muy buen nivel de seguridad en los datos.

En aplicaciones Web hay baja concurrencia en la modificación de datos y en cambio el entorno es intensivo en lectura de datos, lo que hace a MySQL ideal para este tipo de aplicaciones, es por esto que como SGBD se seleccionó MYSQL.

1.6.4 Servidor Web Apache 2.2

Es un software de código abierto que ofrece una perfecta combinación entre estabilidad y sencillez, por ser un servidor Web potente y flexible que pueda funcionar en la más amplia variedad de plataformas y entornos. Por supuesto, se distribuye prácticamente con todas las implementaciones de Linux.

Tiene capacidad para servir páginas tanto de contenido estático, para lo que nos serviría sencillamente un viejo ordenador 486, como de contenido dinámico a través de otras herramientas soportadas que facilitan la actualización de los contenidos mediante bases de datos, ficheros u otras fuentes de información.

Apache se ha adaptado siempre a una gran variedad de entornos a través de su diseño modular. Este diseño permite a los administradores de sitios Web elegir que características van a ser incluidas en el servidor seleccionando que módulos se van a cargar, ya sea al compilar o al ejecutar el servidor.

Las principales características de Apache son:[5]

- Funcionalidad en *múltiples plataformas*.
- Elaborado índice de directorios.
- Soporte del último *protocolo http*.
- *Sencilla administración* basada en la configuración de un único archivo.
- *Soporte para CGI (Common Gateway Interface) y FastCGI*.

Beneficios.

Apache puede soportar de una forma más fácil y eficiente una amplia variedad de sistemas operativos. El servidor puede personalizarse mejor para las necesidades de cada sitio web. Además se pueden configurar funcionalidades especiales como servir diferentes hosts con diferentes identificadores de usuarios (perchild).

1.6.5 Modelo Vista Controlador.

Un propósito común en numerosos sistemas es el de tomar datos de un almacenamiento y mostrarlos al usuario. Luego que el usuario introduce modificaciones, las mismas se reflejan en el almacenamiento. Dado que el flujo de información ocurre entre el almacenamiento y la interfaz, una tentación común, un impulso espontáneo (hoy se llamaría un anti-patrón) es unir ambas piezas para reducir la cantidad de código y optimizar el rendimiento. (Ver figura 1.3).



Figura 1.3. Modelo vista controlador.

El patrón conocido como Modelo-Vista-Controlador (MVC) separa el modelado del dominio, la presentación y las acciones basadas en datos ingresados por el usuario en tres clases diferentes:

Modelo: Administra el comportamiento y los datos del dominio de aplicación, responde a requerimientos de información sobre su estado (usualmente formulados desde la vista) y responde a instrucciones de cambiar el estado (habitualmente desde el controlador).

Vista: Maneja la visualización de la información.

Controlador: Controla el flujo entre la vista y el modelo (los datos).

Tanto la vista como el controlador dependen del modelo, el cual no depende de las otras clases. Esta separación permite construir y probar el modelo, independientemente de la representación visual.

Entre las ventajas del estilo Modelo-Vista-Controlador están las siguientes:

- Soporte de múltiples vistas: Dado que la vista se haya separada del modelo y no hay dependencia directa del modelo con respecto a la vista, la interfaz de usuario puede mostrar múltiples vistas de los mismos datos simultáneamente. Por ejemplo, múltiples páginas de una aplicación Web pueden utilizar el mismo modelo de objetos mostrado de maneras diferentes.
- Adaptación al cambio: Los requerimientos de interfaz de usuario tienden a cambiar con mayor rapidez que las reglas de negocios. Los usuarios pueden preferir distintas opciones de representación, o requerir soporte para nuevos dispositivos como teléfonos celulares o PDAs. Dado que el modelo no depende de las vistas, agregar nuevas opciones de presentación generalmente no afecta al modelo.

Una desventaja que tiene este modelo es el costo de actualizaciones frecuentes: si el modelo experimenta cambios frecuentes, por ejemplo, podría desbordar las vistas con una lluvia de requerimientos de actualización.

1.6.6 Proceso de Desarrollo.

La producción de software busca adecuarse cada día más a las necesidades del usuario, esto trae como consecuencia que la misma aumente en tamaño y complejidad. Por estas razones, se hizo necesario definir la metodología de ingeniería del software que guiara el proceso de automatización de este trabajo, la cuál fue seleccionada el Proceso Unificado de Desarrollo de Software (RUP), junto con el Lenguaje Unificado de Modelado (UML).

1.6.6.1 RUP

La primera versión de RUP, fue puesta en el mercado en 1998 después de haberse realizado varias prácticas.

El Proceso Racional Unificado o RUP (Rational Unified Process), es un proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado (UML), constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos.

Está basado en componentes, lo cual quiere decir que el sistema software en construcción está formado por componentes software interconectados a través de interfaces bien definidas.

Principales características de RUP durante su ciclo de vida:

- Iterativo e incremental.
- Centrado en la arquitectura.
- Guiado por los casos de uso.

RUP divide el proceso de desarrollo en ciclos, teniendo un producto final al concluir cada ciclo; éstos se dividen en 4 fases, las cuales establecen oportunidad y alcance, identifican las entidades externas o actores con las que se trata, e identifica también los casos de uso. Así como contiene 9 flujos de trabajos: los 6 primeros son conocidos como flujos de Ingeniería y los restantes 3 como de apoyo. En la siguiente figura 1.4 se representan los mismos.

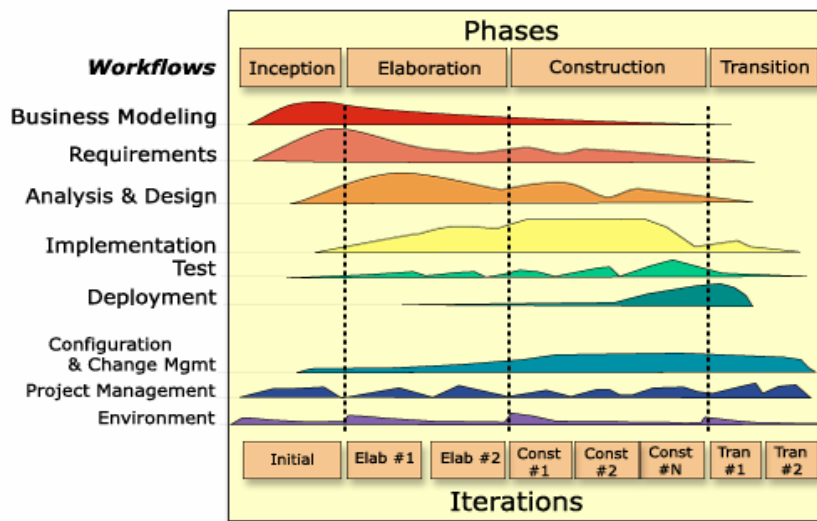


Figura 1.3. Flujos de trabajo y fases de RUP.

1.6.6.2 UML

UML (Unified Modeling Language) es un lenguaje que permite modelar, especificar, construir y documentar los elementos que forman un sistema software orientado a objetos; no un método o un proceso. Se ha convertido en el estándar de facto de la industria, debido a que fue impulsado por los autores de los tres métodos más usados de orientación a objetos: Grady Booch, Ivar Jacobson y Jim Rumbaugh.

UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocios y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes de software reutilizables.

UML cuenta con varios tipos de diagramas, los cuales muestran diferentes aspectos de las entidades representadas.

De forma general las principales características son: [6]

- *Lenguaje unificado* para la modelación de sistemas.
- *Tecnología orientada a objetos*.
- El cliente participa en todas las etapas del proyecto.
- *Corrección de errores viables* en todas las etapas.
- Aplicable para tratar asuntos de escala inherentes a sistemas complejos de misión crítica, tiempo real y cliente/servidor.

1.6.7 Herramienta utilizadas.

1.6.7.1 Rational Rose.

Se decidió que se utilizaría el Rational Rose Enterprise Edition 2003, para sustentar la documentación, como modelador visual de la notación UML (Unified Modeling Language) para la confección de los diagramas que se ilustran en este documento. Esta herramienta es muy completa y ofrece amplias potencialidades.

Rational Rose es una herramienta CASE que da soporte al modelado visual mediante UML ofreciendo distintas perspectivas del sistema. Da soporte al Proceso Unificado de Rational (RUP). Permite especificar, analizar, y diseñar el sistema antes de codificarlo.

Características Rational Rose.

- Mantiene la consistencia de los modelos del sistema software.

- Diseño centrado en casos de uso y enfocado al negocio que genera un software de mayor calidad.
- Chequeo de la sintaxis UML.
- Generación de documentación automáticamente.
- Generación de código a partir de los modelos.
- Capacidades de ingeniería inversa (crear modelo a partir código).
- Disponibilidad en múltiples plataformas.

Rational Rose es la herramienta CASE que comercializan los desarrolladores de UML y que soporta de forma completa la especificación del UML.

Esta herramienta de software propone la utilización de cuatro tipos de vistas para realizar un diseño del sistema: vista de Caso de Uso, vista Lógica, vista de Componentes y la vista de Despliegue; permitiendo crear y refinar de esta forma un modelo completo que represente el dominio del problema y el sistema de software.

1.6.7.2 Diseño de interfaz: Dreamweaver 8.0

El diseño de interfaz es uno de los elementos "clave" en la realización del programa. Podemos definir el interfaz como: "el conjunto de trabajos y pasos que seguirá el usuario, durante todo el tiempo que se relacione con el programa, detallando lo que verá y escuchará en cada momento, y las acciones que realizará, así como las respuestas que el sistema le dará". [7]

Dreamweaver es la herramienta de diseño de páginas Web más avanzada. Es un programa que se puede actualizar con componentes, que fabrica tanto Macromedia como otras compañías, para realizar otras acciones más avanzadas. Cumple perfectamente el objetivo de diseñar páginas con aspecto profesional, y soporta gran cantidad de tecnologías muy fáciles de usar:

- Hojas de estilo y capas.
- JavaScript para crear efectos e interactividades.
- Inserción de archivos multimedia.

Su última versión, Dreamweaver 8.0, es la herramienta de desarrollo Web líder del mercado y permite a sus usuarios diseñar, desarrollar y mantener de forma eficaz sitios y aplicaciones Web basadas en normas. Con esta los desarrolladores Web lo abarcan todo, desde la creación y mantenimiento de sitios Web básicos hasta aplicaciones avanzadas compatibles con las mejores prácticas y las tecnologías más recientes.

Además incluye herramientas para trabajar aplicaciones que manejan XML, así como mejoras a su manejo de hojas de estilo (CSS). Para el trabajo con CSS han simplificado la creación y manejo de diferentes estilos, promoviendo los estándares para nuevos usuarios y facilitando su aplicación para usuarios avanzados. Manifiesta el potencial que representa para la comunidad Web el desarrollo abierto basado en estándares y mejora, al mismo tiempo, las herramientas básicas que necesitan los profesionales. Con esta, los desarrolladores Web lo abarcan todo, desde la creación y mantenimiento de sitios Web básicos hasta aplicaciones avanzadas compatibles con todas las principales tecnologías de servidor como, por ejemplo, ColdFusion, PHP, ASP, ASP.NET y JSP, para que los desarrolladores, más expertos o menos expertos, puedan dar vida a sus diseños. También han buscado facilitar la difusión de Flash Video, con herramientas que permiten incluir este formato muy fácilmente en páginas Web.

1.6.7.3 Zend Studio 5.1

Se trata de un programa de la casa Zend, impulsores de la tecnología de servidor PHP, orientada a desarrollar aplicaciones Web, en lenguaje PHP. Además de servir de editor de texto para páginas PHP, proporciona una serie de ayudas que pasan desde la creación y gestión de proyectos hasta la depuración de código. Es muy útil para el trabajo en equipo porque implementa opciones de trabajo conocido como CVS (sistema de controlador de versiones). La depuración hace la vida más fácil al programador, agregando que permite corregir los errores sintácticos en tiempo real.

El programa entero está escrito en Java, lo que a veces supone que no funcione tan rápido como otras aplicaciones de uso diario. Sin embargo, esto ha permitido a Zend lanzar con relativa facilidad y rapidez versiones del producto para Windows, Linux y MacOS, aunque el desarrollo de las versiones de este último sistema se retrase un poco más.

Zend Studio consta de dos partes en las que se dividen las funcionalidades de parte del cliente y las del servidor. Las dos partes se instalan por separado, la del cliente contiene el interfaz de edición y la ayuda. Permite además hacer depuraciones simples de scripts, aunque para disfrutar de toda la potencia de la herramienta de depuración habrá que disponer de la parte del servidor, que instala Apache y el módulo PHP o, en caso de que estén instalados, los configura para trabajar juntos en depuración.

El mismo es uno de los ambientes de desarrollo integrado o Integrated Development Environment (IDE) que a través de un comprensivo conjunto de herramientas de edición, depurado, análisis, optimización y bases de datos acelera los ciclos de desarrollo y simplifica los proyectos complejos.

1.6.7.4 SQL Manager for MySQL 2005 versión 3.7.0.1

SQL Manager para MySQL 2005 es una herramienta de alto aprovechamiento para la administración y desarrollo de un servidor de Base de Datos MySQL. Soporta cualquier versión del MySQL y ofrece incluso las vistas, funciones y procedimientos almacenados, las llaves extranjeras con el motor de almacenamiento InnoDB. El Diseñador de la Base de Datos Visual está completamente rehecho. Posee facilidad de usar asistentes que realizan los servicios de MySQL y la capacidad de importar y exportar datos. Tiene poderosa administración de seguridad y da la posibilidad de acceder al servidor MySQL a través del protocolo http además de presentar múltiples conexiones a diferentes máquinas servidoras de base de datos MySQL.

1.6.7.5 Framework Code Igniter 1.5

En el desarrollo de software, un framework es una estructura de soporte definida, en la cual otro proyecto de software puede ser organizado y desarrollado. Típicamente, un framework puede incluir soporte de programas, bibliotecas y un lenguaje de scripting entre otros software para ayudar a desarrollar y unir los diferentes componentes de un proyecto.

Representa una arquitectura de software que modela las relaciones generales de las entidades del dominio. Provee una estructura y una metodología de trabajo la cual extiende o utiliza las aplicaciones del dominio.[8]

Lista de Framework en PHP.

- Symfony - Framework MVC.
- Akelos Framework - MVC multilingüe desarrollado en España. (port de Rails a PHP)
- BIF - framework hecho en PHP
- CakePHP - framework MVC (inspirado en Ruby on Rails)
- Code Igniter - framework.
- Kumbia - Framework MVC (inspirado en Ruby on Rails)
- Prado.
- Zend FrameWork.
- Zheta FrameWork - con lenguaje propio
- ChiRoTuPHP - framework MVC (realizada por chichi, Tuco y Roberto Q. de la UAGRM)

“Code Igniter es una framework no abrumador, es un potente framework, construida para los programadores de PHP que necesitan una caja de herramientas simple y elegante para crear sus propios códigos. Es muy sencillo, y fácil de adaptar a cualquier sistema que deseemos realizar.”[9]

“Está pensado para ofrecer un alto rendimiento, ser ligero y fácilmente instalable (puede usarse en un alojamiento compartido y no es necesario tener acceso a la línea de comando). Además, gracias a la documentación y los foros (en inglés ambos), el aprendizaje es muy rápido y en un par de horas se puede empezar a trabajar con él. Por supuesto, será necesario más tiempo para conocer su funcionamiento completo, pero enseguida pueden empezar a hacerse cosas muy interesantes.

Viene con varias librerías para gestionar el acceso a datos, sesiones de usuarios, formularios, la seguridad, etc.... Además la comunidad de usuarios ha creado una serie de plugins, clases y librerías para extenderlo que lo hacer aún más interesante.” [10]

1.6.7.6 Framework AJAX: YUI

La librería YUI es libre. Está bajo la licencia BSD [11] que permite comercial o no la proliferación entre los usuarios. El código YUI es usado y refinado en millones de páginas vistas cada día. YUI se ha incrementado comprensiblemente, bien documentada y soportada por la comunidad en YUI forum.

1.7 Conclusiones.

Luego de un profundo estudio de los fundamentos teóricos de las herramientas, lenguajes y tecnologías que se han abordado en este capítulo, se llegó a la conclusión que se desarrollará una aplicación Web, se planteó el gestor de base de datos y el lenguaje de programación a emplear para la realización de dicho sistema. Se determinaron las herramientas que se consideraron adecuadas para la realización del diseño de la aplicación, teniendo en cuenta muchos factores y tratando de cumplir al máximo con las exigencias que se requieren.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

2.1 Introducción.

En este capítulo se hace un análisis del negocio que se plantea automatizar. Se hace una descripción de los procesos involucrados en la gestión de los objetivos y los responsables así como una selección de los actores que se beneficiarán de la realización de los casos de uso y de los trabajadores que los ejecutarán como posibles candidatos a ser actores del sistema. También se muestran los objetos creados y los que se crean en el negocio. En resumen está dedicado a conocer a profundidad el comportamiento real del negocio objeto del proyecto. También se sigue todo el proceso de análisis de los casos de uso objeto de automatización, flujo de vital importancia para el desarrollo de la aplicación.

2.2 Objeto de Estudio.

2.2.1 Problema y situación problemática.

Actualmente muy pocas empresas cubanas cuentan con la metodología CMI. El país no tiene mucha cultura con respecto a esta herramienta y por lo tanto no siempre se toman las mejores decisiones ni se centran en los verdaderos objetivos estratégicos por lo que se aspira a que en un futuro todas las empresas cubanas puedan hacer uso de esta metodología que les permita a las organizaciones trazar un mejor camino a seguir.

La gestión de los objetivos y los responsables se realiza de forma manual, lo que conlleva a que el trabajo sea algo engorroso y que se cometan errores que dañen el buen funcionamiento de la empresa así como pérdida de la información pues al no estar accesible la misma el personal autorizado no puede tener un mejor control de todo el proceso. Con el objetivo de mejorar este proceso en la entidad, se decidió realizar una aplicación en ambiente Web, multiplataforma que permita gestionar los objetivos, los responsables y demás procesos de la unidad negocio, así como facilitar un mejor control de toda la información que se maneje.

La aplicación deberá realizar diferentes funciones como crear, modificar y eliminar además de tener un manual de usuario para el uso correcto del sistema obtenido.

2.2.2 Objeto de automatización

Los procesos que son objeto de automatización de dicho sistema son los siguientes:

- Gestionar objetivos: es un conjunto de actividades que requieren otros procesos de la entidad lo cual conlleva a que sea una de las gestiones más abarcadoras del sistema. También permite una mejor distribución de los recursos y medios disponibles a efectos de poder obtener aquellas metas trazadas.
- Gestionar responsables: permite un control de los responsables con todos los datos requeridos como es su área de trabajo y la función que realiza.

2.2.3 Información que se maneja.

La información que se maneja hace referencia al documento de la estrategia, el plan de acción y el de los objetivos que incluye los documentos anteriores para los procesos que tienen lugar en las empresas del país. En el documento de la estrategia se almacenan el nombre y la descripción de la misma, lo mismo sucede con el plan de acción. En el caso de los objetivos presenta el nombre y la descripción además de tener de forma implícita el contenido de las estrategias y los planes de acción.

2.3 Propuesta de sistema.

La propuesta como sistema es dar la posibilidad de agilizar los procesos que se realizan en cada perspectiva, ayudando a definir los objetivos trazados, estrategias, indicadores, plan de acción y responsables implicados en todas estas actividades. Además de facilitar el manejo de dicha aplicación pues su contenido es entendible para todo el personal que desee acceder a ella aunque posea pocos conocimientos informáticos.

2.4 Modelo de negocio.

Con la automatización del módulo Gestionar objetivos y responsables, se logra controlar el proceso de organización de la entidad en cuanto los indicadores, responsables, estrategias y plan de acción de las diferentes perspectivas. Muchas veces, las personas implicadas en este proceso no pueden ver como va marchando los objetivos trazados por lo que este sistema automatizado logrará que se puedan tomar nuevas medidas con mayores beneficios a la entidad.

Con el modelo de negocio se describen los procesos de un negocio (casos de uso del negocio) y su interacción con elementos externos (actores), tales como socios y clientes, es decir, describe las funciones que el negocio pretende realizar y su objetivo básico es describir cómo el negocio es utilizado por sus clientes y socios.

2.4.1 Actores del negocio

Un actor del negocio es cualquier individuo, grupo, entidad, organización, máquina o sistema de información externos con los que el negocio interactúa. Lo que se modela como actor es el rol que se juega cuando se interactúa con el negocio para beneficiarse de sus resultados.

A continuación se muestra en la tabla 2.1 los actores del negocio y su correspondiente justificación:

Actores del negocio	Justificación
Directivo	El directivo es el que inicia las acciones que tiene lugar en los procesos de la gestión de objetivos y responsables, y al mismo tiempo es el que se beneficia con el resultado de dicho proceso.

Tabla 2.1 Actores del negocio

2.4.2 Trabajadores del negocio

Un trabajador del negocio es una abstracción de una persona (o grupo de personas), una máquina o un sistema automatizado; que actúa en el negocio realizando una o varias actividades, interactuando con otros trabajadores del negocio y manipulando entidades del negocio. Representa un rol.

A continuación se muestra en la tabla 2.2 los trabajadores del negocio y su correspondiente justificación:

Trabajadores del negocio	Justificación
Jefe del Área	Es el que lleva a cabo todas las actividades correspondientes a la gestión de los objetivos y los responsables. Toma las decisiones que cree necesaria y no se beneficia en ningún momento de las acciones realizadas en los procesos del negocio que tienen lugar sino que se limita a ejecutar dichas acciones.

Tabla 2.2 Trabajadores del negocio

2.4.3 Diagrama de Casos de Uso del Negocio

Un diagrama de casos de uso del negocio representa gráficamente a los procesos del negocio y su interacción con los actores del negocio. En la figura 2.1 se muestra el correspondiente a este trabajo:

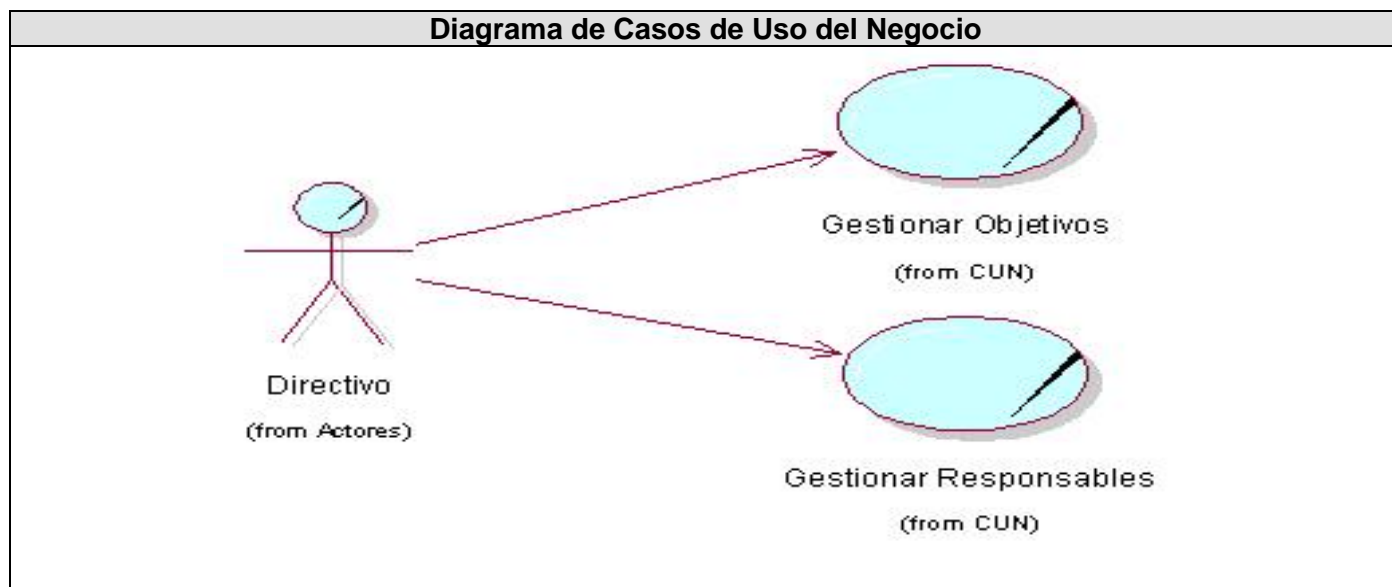


Figura 2.1 Diagrama del CUN

2.4.4 Descripción textual de los casos de uso del negocio.

(Ver Anexo 1)

2.5 Diagrama de actividades

El diagrama de actividades es un grafo que contiene estados en que puede hallarse una actividad. Además, describe un proceso que explora el orden de las tareas o actividades que logran los objetivos del negocio. Así como se divide en varias calles, donde cada una de ella representa el actor y trabajador que está llevando a cabo en el proceso y muestra cómo se utilizan las entidades del negocio. (Ver Anexo 2)

2.6 Diagrama de clases del modelo de objeto

El diagrama de clases, como artefacto que se construye para describir el modelo de objetos del negocio, muestra la participación de los trabajadores y entidades del negocio y la relación entre ellos. (Ver Anexo 3).

2.7 Especificación de los requisitos de software

Los requisitos del software son condiciones o capacidades que necesita un usuario para resolver un problema o lograr un objetivo, dichas capacidades o condiciones tienen que ser alcanzada o poseída por un sistema o componente de un sistema para satisfacer un contrato, estándar, u otro documento impuesto formalmente.

2.7.1 Definición de los requerimientos funcionales

Los requerimientos funcionales son capacidades o condiciones que el sistema debe cumplir. Estos no alteran la funcionalidad del producto, esto quiere decir que los requerimientos funcionales se mantienen invariables sin importarle con que propiedades o cualidades se relacionen.

Para cumplir con los objetivos propuestos se previó que el sistema tuviese las siguientes funcionalidades:

R.1 Gestionar Objetivos

- R.1.1 Crear objetivo
- R.1.2 Modificar objetivo
- R.1.3 Eliminar objetivo
- R.1.4 Seleccionar responsables
- R.1.5 Seleccionar indicadores
- R.1.6 Seleccionar estrategia
- R.1.7 Seleccionar plan de acción
- R.1.8 Seleccionar perspectiva
- R.1.9 Seleccionar negocio

R.2 Gestionar Responsables

- R.2.1 Crear responsables
- R.2.2 Modificar responsables
- R.2.3 Eliminar responsables
- R.2.4 Seleccionar categoría
- R.2.5 Seleccionar rol

R.3 Gestionar Categoría

- R.3.1 Crear categoría
- R.3.2 Modificar categoría
- R.3.3 Eliminar categoría

R.4 Gestionar Rol

- R.4.1 Crear rol
- R.4.2 Modificar rol
- R.4.3 Eliminar rol

2.7.2 Definición de los requerimientos no funcionales

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable.

Existen múltiples categorías para clasificar a los requerimientos no funcionales, siendo las siguientes representativas de un conjunto de aspectos que se deben tener en cuenta:

- *Apariencia o interfaz externa:* Un ambiente Web con banner que represente el tema del software. Colores serios y pocos. La mayor parte de la interfaz debe tener apariencia de aplicación de escritorio.
- *Usabilidad:* El sistema debe permitir al usuario asimilar el uso del sistema de una manera rápida e intuitiva, sin necesidad de cursos ni consultas previas. Para asegurar el entendimiento, capacitar al software de una ayuda en conceptos y guía de instrucciones en los servicios automatizados. El usuario puede no tener ninguna experiencia con sistemas informáticos.
- *Rendimiento:* El sistema debe dar posibilidades de usarse en computadores de mediano y bajo rendimiento por las características del hardware que posee Cuba. Procesar la información lo más rápida posible.
- *Soporte:* Minimizar las necesidades de soporte. En caso de posible trabajo de mantenimiento, la Universidad de las Ciencias Informáticas dará este tipo de aseguramiento.
- *Software:* Como la aplicación tiene la arquitectura cliente/servidor es necesario que en la máquina servidor esté disponible:
 - El Apache Web Server 2.x
 - Un servidor de MySQL 5.0 o superior.
 - Un navegador para visualizar e interactuar con la aplicación. Puede ser Internet Explorer desde la versión 5. Otro candidato es el Mozilla Firefox.
- *Portabilidad:* El sistema debe brindar posibilidades de uso en varias plataformas, como mínimas en Windows 98, 2000, 2003, XP y las distintas distribuciones de Linux.
- *Seguridad:* El control de la entrada de datos será regulada a través de usuarios autenticados con roles determinados. Otro tipo de usuario solo tendrá acceso leer el contenido. Además el sistema será puesto en redes locales, solo teniendo acceso los usuarios de la red.

- *Políticos-culturales:* La aplicación servirá a los intereses de la Universidad de las Ciencias Informáticas y de Cuba.
- *Legales:* No poseer compromiso de pago por el uso de alguna reutilización de código empleado, no perteneciente a la Universidad de las Ciencias Informáticas.
- *Confiabilidad:* La herramienta de implementación a utilizar tiene soporte para recuperación ante fallos y errores.
- *Ayuda:* Implementar un manual de usuario competente.

2.8 Definición de los casos de usos.

Los casos de uso son artefactos narrativos que describen, bajo la forma de acciones y reacciones, el comportamiento del sistema desde el punto de vista del usuario. Por lo tanto, establece un acuerdo entre clientes y desarrolladores sobre las condiciones y posibilidades (requisitos) que debe cumplir el sistema.

Los casos de uso candidatos también se encuentran entre las actividades a automatizar. Esto no significa que una actividad se convierta en un caso de uso porque un caso de uso es un proceso que da un resultado de valor para un actor determinado y una secuencia de actividades a automatizar puede implicar pasos dentro de un caso de uso.

2.8.1 Definición de los actores

Cada trabajador del negocio (inclusive si fuera un sistema ya existente) que tiene actividades a automatizar es un candidato a actor del sistema. Si algún actor del negocio va a interactuar con el sistema, entonces también será un actor del sistema.

Actores del Sistema	Justificación
Directivo	Es el actor del sistema que se va a beneficiar con las distintas actividades que se van a automatizar en el sistema, como son: gestionar objetivos, gestionar responsables, gestionar categoría y gestionar rol.

Tabla 2.3 Descripción de los actores del sistema.

2.8.2 Diagrama de Casos de Uso del Sistema

Un diagrama de casos de uso del sistema representa gráficamente a los procesos y su interacción con los actores.

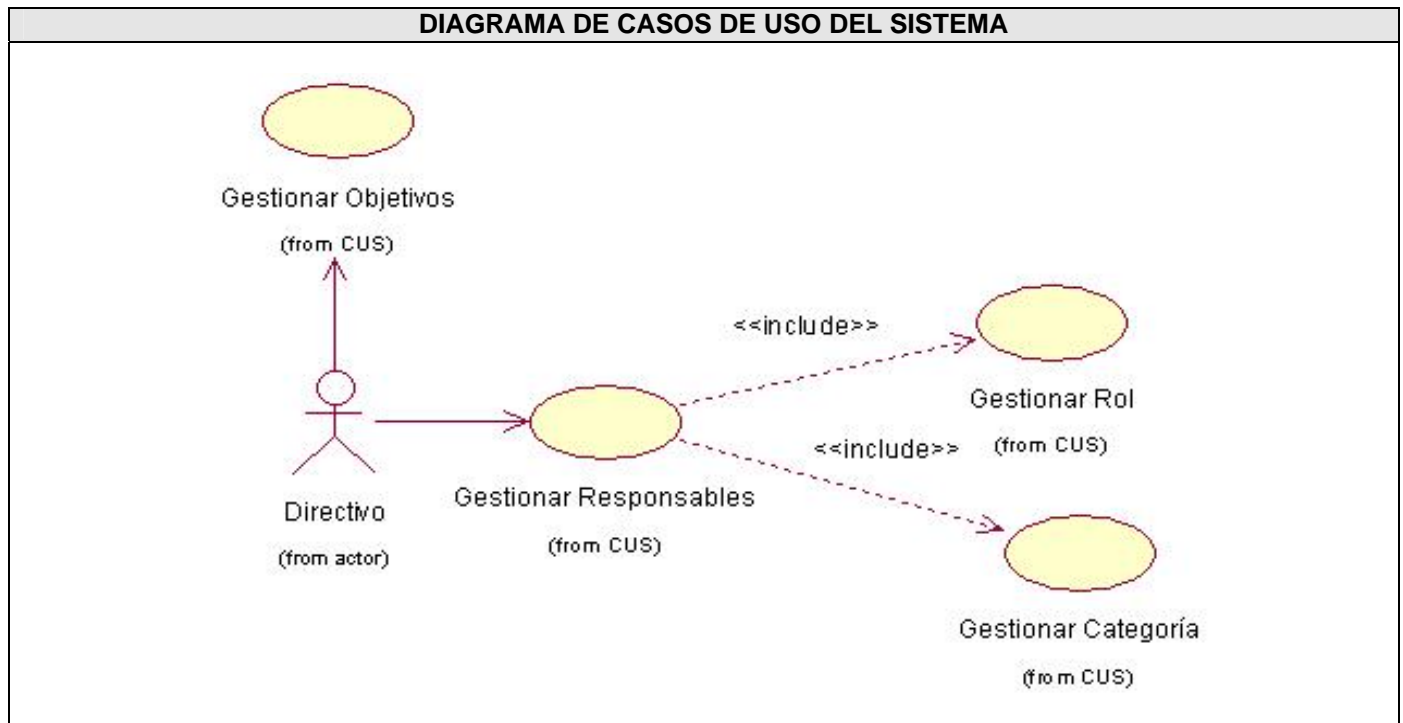


Figura 2.2 Diagrama del CUS

2.8.3 Descripción textual de los casos de uso del sistema.

(Ver Anexo 4)

2.9 Conclusiones.

En este capítulo se comenzó a desarrollar la propuesta de solución. A partir del análisis del negocio, se obtuvieron las funciones que debe tener el sistema, representadas por medio de Diagrama de Casos de Uso y las características que debe poseer mediante los requerimientos funcionales y no funcionales. Se describieron cada una de las acciones de los actores del sistema y la respuesta de los casos de uso con que interactúan. En estos momentos se está en condiciones de empezar a construir el sistema, cumpliendo con los requerimientos capturados y las observaciones realizadas a lo largo de este capítulo.

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

3.1 Introducción.

En este capítulo se harán las consideraciones necesarias para la construcción de la solución propuesta anteriormente, se modelarán los artefactos necesarios para cumplir con las tareas trazadas. Se tratarán los principales patrones empleados para el desarrollo de la aplicación, muy necesarios y útiles, al igual que los mecanismos que fueron necesarios emplear a lo largo del proceso de Análisis y Diseño de la aplicación web. Al terminar, se tendrá un modelo de Datos que satisfaga los requerimientos del sistema para la gestión de los objetivos y los responsables.

3.2 Diagramas de clases del análisis.

Un Diagrama de clases del análisis es un artefacto en el que se representan los conceptos en un dominio del problema. Representa las cosas del mundo real, no de la implementación automatizada de estas cosas. Esboza cómo llevar a cabo la funcionalidad dentro del sistema incluida la funcionalidad significativa para la arquitectura; sirve como una primera aproximación del diseño.

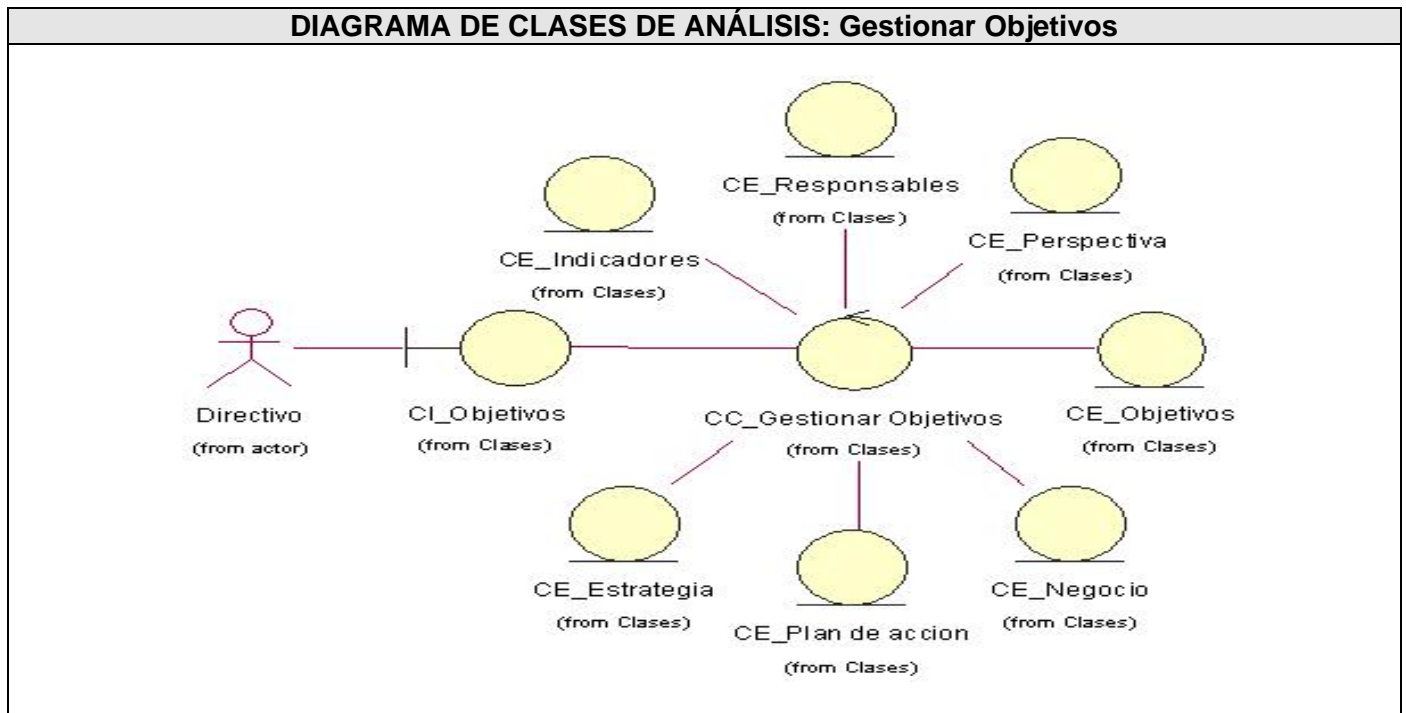


Figura 3.1 Diagrama de clases del análisis: Gestionar Objetivos

DIAGRAMA DE CLASES DE ANÁLISIS: Gestionar Responsables

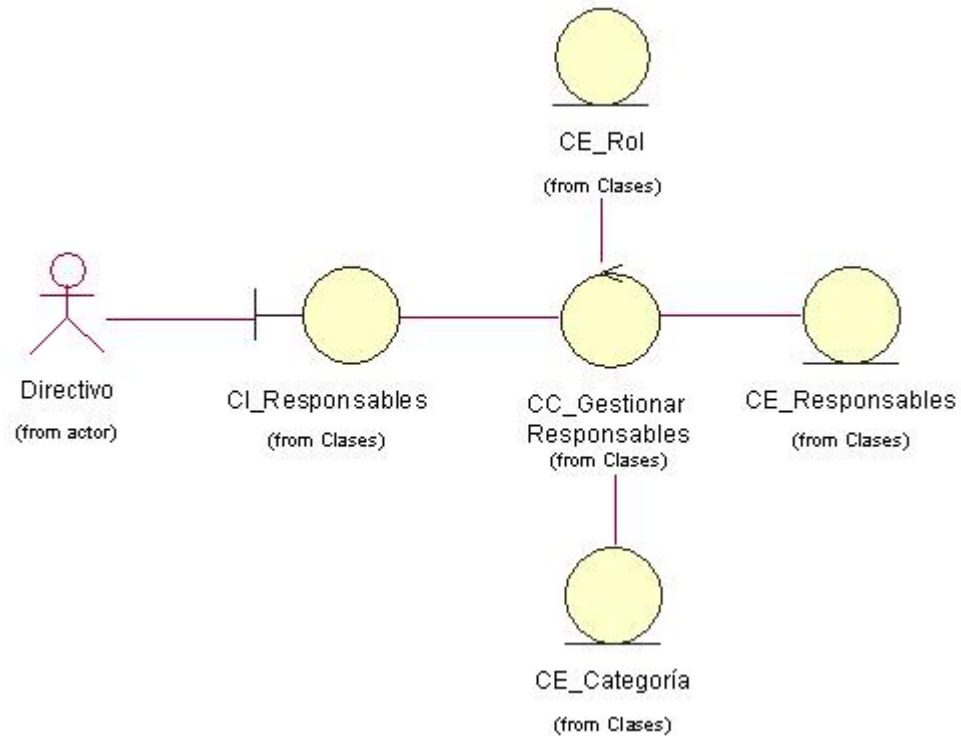


Figura 3.2 Diagrama de clases del análisis: Gestionar responsables.

DIAGRAMA DE CLASES DE ANÁLISIS: Gestionar Categoría

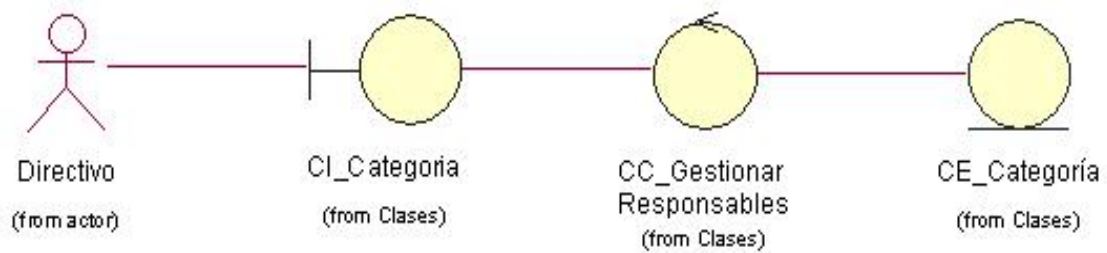


Figura 3.3 Diagrama de clases del análisis: Gestionar Categoría.

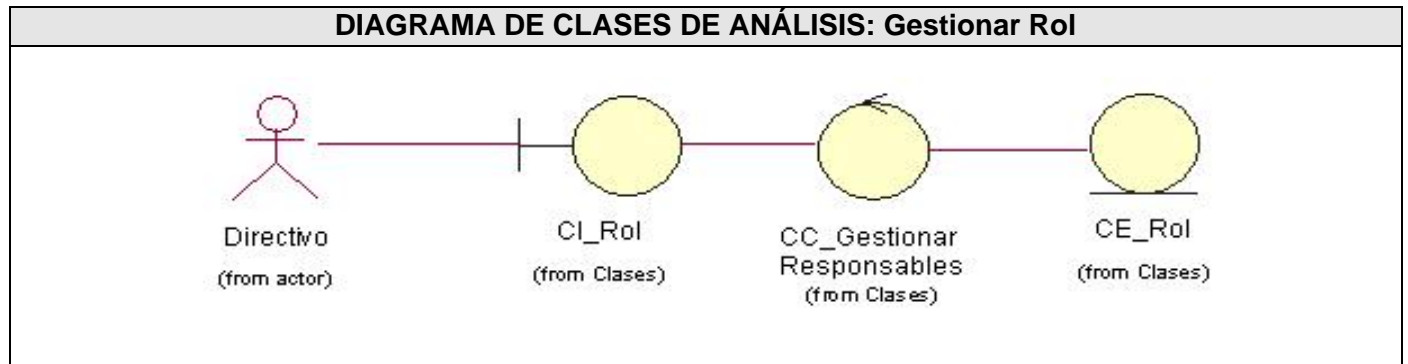


Figura 3.4 Diagrama de clases del análisis: Gestionar Rol.

3.3 Diagramas de clases del diseño.

Un diagrama de clases del diseño es una representación más concreta que el diagrama de clases del análisis, representa la parte estática del sistema, las clases y sus relaciones.

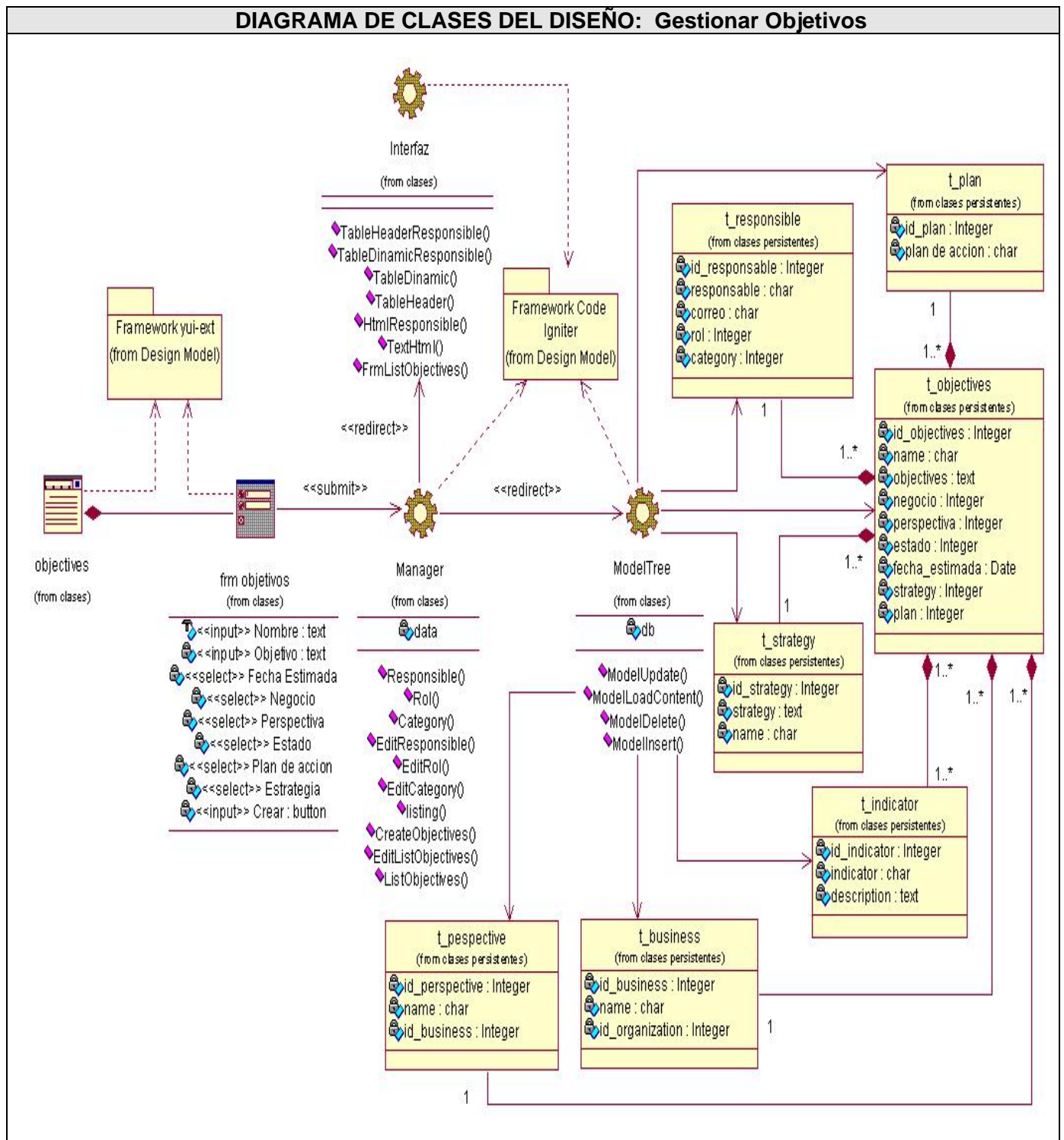


Figura 3.5 Diagrama de clases del diseño: Gestionar Objetivos.

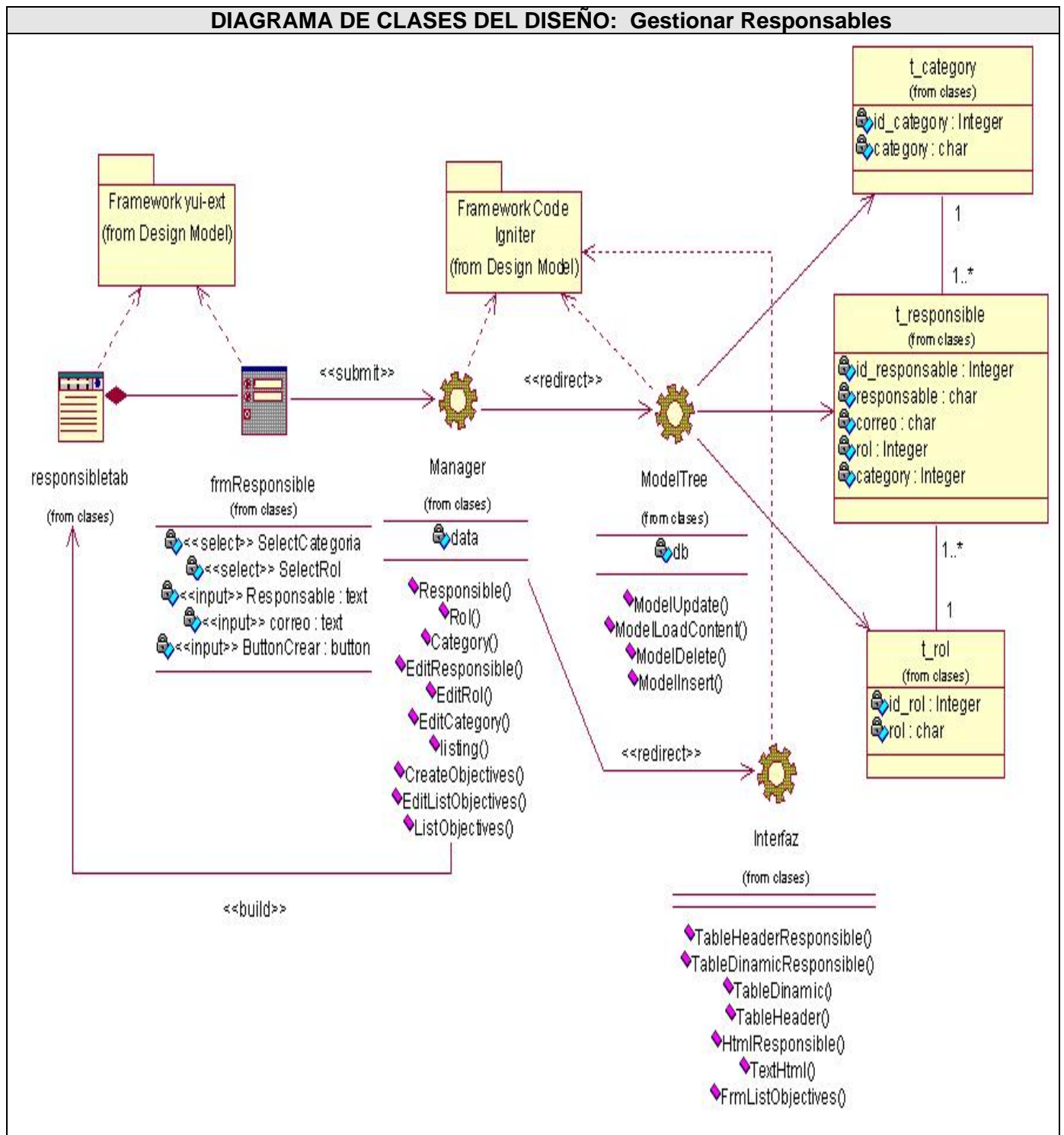


Figura 3.6 Diagrama de clases del diseño: Gestionar Responsables.

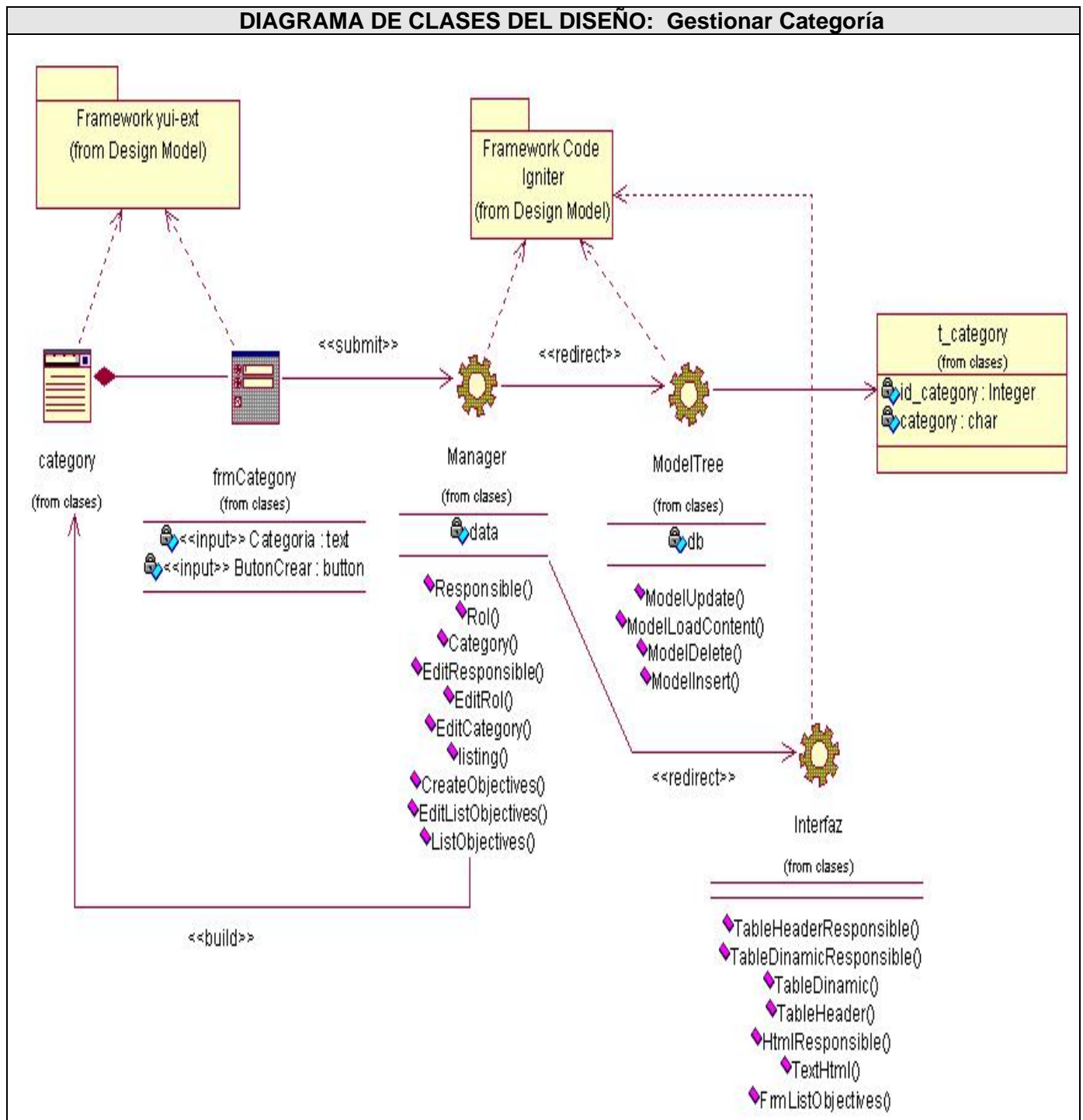


Figura 3.7 Diagrama de clases del diseño: Gestionar Categoría.

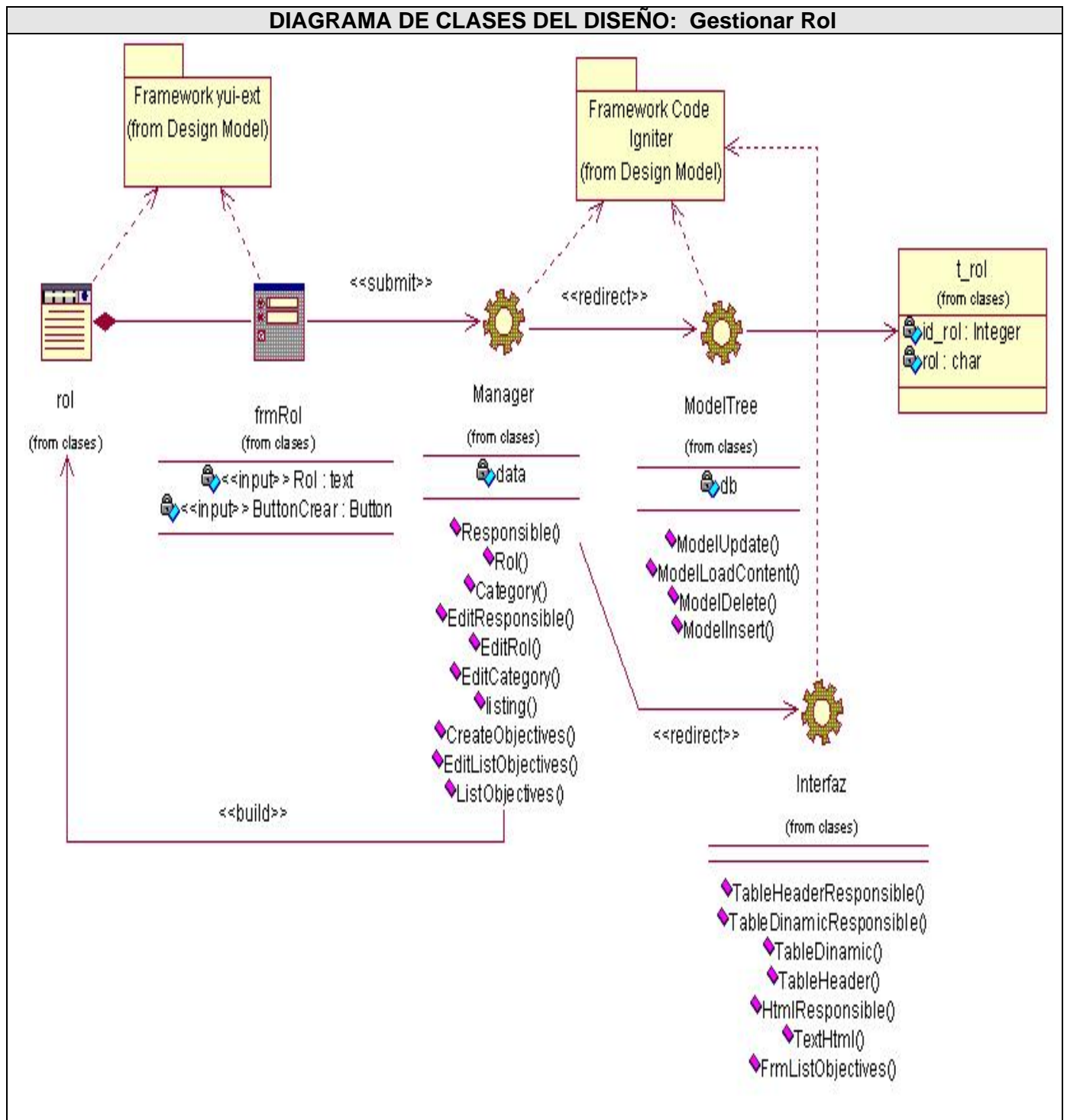


Figura 3.8 Diagrama de clases del diseño: Gestionar Rol.

3.4 Diagramas de interacción del diseño.

Los Diagramas de Interacción muestran una interacción concreta: un conjunto de objetos y sus relaciones, junto con los mensajes que se envían entre ellos; modelan el comportamiento dinámico del sistema, el flujo de control en una operación; describen la interacción entre objetos, los objetos interactúan a través de mensajes para cumplir ciertas tareas; las interacciones provee un “comportamiento” y típicamente implementan un Caso de Uso.

Estos diagramas sirven para mostrar la relación entre los distintos objetos que participan en un escenario y la relación se establece mediante el paso de mensajes.

Un diagrama de colaboración es una forma alternativa al diagrama de secuencia de mostrar un escenario. Este tipo de diagrama muestra las interacciones entre objetos organizadas entorno a los objetos y los enlaces entre ellos. Es un diagrama de interacción que enfatiza la organización estructural de los objetos que participan en una interacción. **(Ver Anexo 5)**

3.5 Descripción de las clases.

(Ver Anexo 6).

3.6 Diseño de la Base de datos

3.6.1 Modelo físico de datos

El modelo de los datos describe la representación lógica y física de datos persistentes en el sistema.

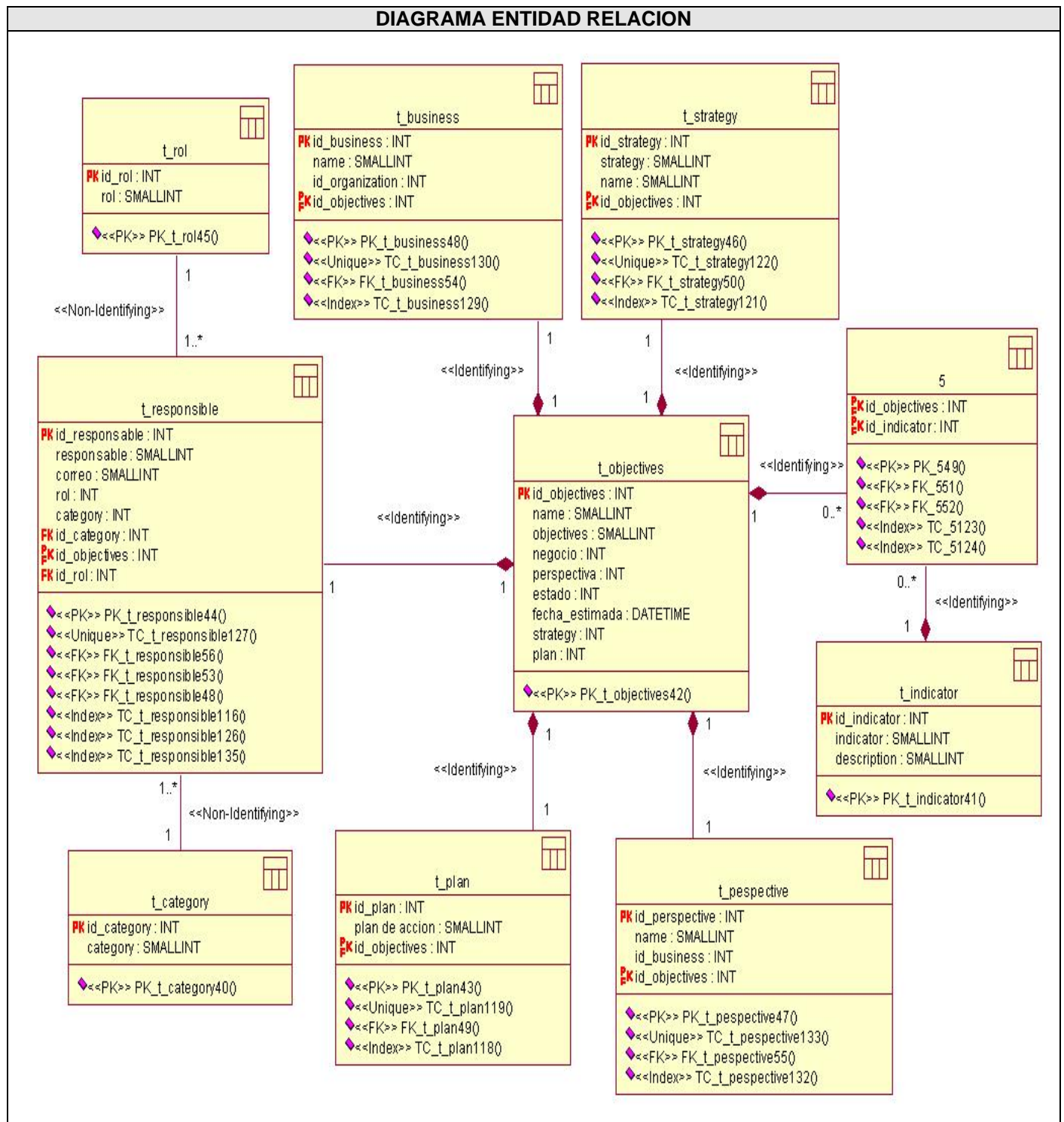


Figura 3.9 Diagrama entidad relación del módulo Gestionar Objetivos y Responsables.

3.7 Descripción de las tablas de la Base de datos

(Ver Anexo 7)

3.8 Definiciones de diseño que se apliquen.

3.8.1 Patrones

La utilización de los patrones ha tomado un gran auge a partir del desarrollo del modelo orientado a objetos. Una arquitectura orientada a objetos bien estructurada está llena de patrones. La calidad de un sistema orientado a objetos se mide por la atención que los diseñadores han prestado a las colaboraciones entre sus objetos. Los patrones conducen a arquitecturas más pequeñas, más simples y más comprensibles. Un sistema orientado a objetos se compone de objetos que envían mensajes a otros objetos para que lleven a cabo las operaciones.

Patrones GRASP:

Los patrones GRASP describen los principios fundamentales de la asignación de responsabilidades a objetos, expresados en forma de patrones.

GRASP es un acrónimo que significa General Responsibility Assignment Software Patterns (patrones generales de software para asignar responsabilidades) El nombre se eligió para indicar la importancia de captar (grasping) estos principios, si se quiere diseñar eficazmente el software orientado a objetos.

- *Experto:* Asignar una responsabilidad al experto en información: la clase que cuenta con la información necesaria para cumplir la responsabilidad. Un ejemplo en el sistema que se propone son las clases interfaz e Interfazhelp, quienes tienen delegadas las responsabilidades de generar el código HTML de formularios.
- *Creador:* Asignarle a la clase **B** la responsabilidad de crear una instancia de la clase **A**. La clase Manager es en el caso del sistema descrito, quien se encarga de crear los objetos como pueden ser, lo fabricados por la clase Modeltree.
- *Bajo Acoplamiento:* Este patrón plantea que las clases deben comunicarse con un pequeño grupo de clases para disminuir las dependencias de las clases y sea más fácil el mantenimiento de las mismas.
- *Patrón Alta Cohesión:* Una clase tiene responsabilidades moderadas en un área funcional y colabora con las otras para llevar a cabo las tareas.

- *Controlador*: Este patrón describe el uso de algunas clases dedicadas a capturar los mensajes de los eventos del sistema y aconseja el uso de una misma clase controladora para controlar los eventos del sistema de un mismo caso de uso. En el sistema propuesto existen varias clases controladoras. Ellas son las clases Manager y Chart.

El patrón MVC conocido como Modelo Vista Controlador es utilizado para definir la arquitectura del sistema. Este patrón es muy utilizado en las aplicaciones Web. Para implementar este patrón fue muy fácil gracias al uso del framework de PHP que está concebido para trabajar bajo esta arquitectura.

Patrones de diseño Web:

- *Alternating Row Colors*: Este patrón es usado para mostrar datos en una tabla con muchos campos y filas y permite darle más legibilidad al lector gracias al uso de colores alternados.
- *Action Button*: Es usado para diferenciar los vínculos de las páginas a los eventos más importantes al usuario como crear perspectivas.
- *Date Selector*: Este patrón es utilizado cuando es necesario en un formulario darle la posibilidad a un usuario, de seleccionar una fecha, de inicio o de fin, o ambas inclusive. Este recurso es usado en varios eventos, uno de ellos es para crear un objetivo estratégico.
- *Footer Bar*: Se utiliza para dar a conocer las condiciones bajo las cuales se puede usar la aplicación. Este patrón está expresado en la parte inferior de la aplicación, donde se explica los derechos de copyright que se atribuye la universidad sobre este software.
- *Tabs*: El uso de los tabs es para tener que llevar una navegación horizontal, agrupar la información por categoría y mostrar parcialmente el contenido por secciones. El uso de los tabs en la aplicación es muy extendido, por lo que este patrón ha facilitado mucho la realización de esta aplicación.
- *Image Menu*: Muy usado para ahorrar espacio en el menú y puede dar la posibilidad de enfatizar el icono seleccionado. Este tipo de diseño permitió crear una barra de herramienta para ubicar las funcionalidades de la aplicación.
- *Topic Pages*: Este patrón describe el uso de encabezados de documentos para ayudar a comprender al usuario el tema de documento expuesto más abajo. En el caso de la aplicación se usan los encabezados en la cabecera de las ventanas para explicar a que se dedica la ventana levantada.
- *Retractable Menu*: Este tipo de menú es muy útil en aquellos que ocupan mucho espacio, por ende la posibilidad de encoger el menú es una opción muy buena a la hora de optimizar el espacio

dedicado a la navegación o al tema principal de la aplicación. En la aplicación, se encuentra una navegación hecha por un árbol que pudiera robar mucho espacio, por tanto el uso de este patrón ha dado solución al problema del espacio de la navegación.

- *Split Navigation*: Este patrón describe que la navegación puede ser a través de una estructura jerárquica o por niveles. Este tipo de navegación permite ir más directo a la información que se busca, sin tener que pasar por otras páginas primero. El uso de la navegación en la aplicación a través de un árbol permite las facilidades de la estructura jerárquica enunciadas por el split navigation.
- *Customizable Windows*: Este patrón se refiere a la posibilidad que da al usuario de personalizar las ventanas desplegadas por el software. Este patrón se encuentra manifiesto en todas las ventanas del Cuadro de Mando Integral al dar la posibilidad de redimensionar las ventanas al tamaño que estime el usuario.

3.9 Tratamiento de errores

Para evitar los males provocados por defectos y errores del sistema, estos males se minimizan y se evitan a través de una serie de estrategias implementadas y otras que ya brinda el framework de PHP Code Igniter.

Para evitar la presencia de errores, cada vez que se termina la programación en cada caso de uso, se le somete a determinados posibles escenarios de entrada de datos y se evalúan las posibles respuestas del sistema o las que le faltarían.

Los mensajes de errores en los valores de entrada de interfaz se muestran con claridad y sencillez para los usuarios.

3.10 Seguridad

La seguridad fue tomada muy en cuenta a la hora de la entrada de los datos y en la selección del framework ya que este aporta una estructura e implementa políticas de seguridad para el desarrollo de la aplicación. Antes que los datos sean guardados en la base de datos es pasado por un filtrado para evitar posibles errores o ataques intencionales del usuario. Son validadas las entradas vacías y se filtra las posibles inyecciones SQL y XSS, este último filtrado lo proporciona el framework de una manera automática.

3.11 Interfaz de usuario.

La calidad de la interfaz de usuario puede ser uno de los motivos que conduzca a un sistema al éxito o al fracaso, es por eso que uno de los aspectos más relevantes de la usabilidad de un sistema es la consistencia de su interfaz de usuario.

Para el desarrollo de la interfaz se tuvo en cuenta los siguientes aspectos:

- Sencillez.
- Atractiva.
- Flexibilidad.
- Apariencia entorno de escritorio para dar más característica de una aplicación y semejanza con otros sistemas que también implementan el Cuadro de Mando Integral y sirva para trasladar la experiencia a este sistema.
- Colores serios.
- Mensajes de errores certeros.

Se tuvieron además las siguientes características:

- Utilizar una misma tipografía, forma y estilo en todas las páginas.
- La facilidad del usuario de poder navegar desde cualquier punto a otro dentro de la aplicación.
- Se tuvo presente siempre el ancho de banda y por ello se utilizaron formato de imágenes de compresión favorables.
- La simplicidad y consistencia, favoreciendo la usabilidad de la aplicación.
- Navegación simple en todas las páginas de la aplicación, de forma tal que siempre sea accesible por el usuario.
- Estabilidad y uniformidad del diseño, para así poder ubicar al usuario dentro del mismo y hacerlo sentir parte de él.

3.12 Concepción de la ayuda.

La ayuda está siempre accesible ya que se encuentra en la barra de herramienta brindada por el sistema. Además del contenido que trata sobre la gestión de los reportes y la Gestión del Sistema, el usuario puede encontrar aspectos teóricos que le ayuden a comprender el uso del sistema y la importancia de la gestión de informes. La ayuda está caracterizada para que el usuario no necesite más ningún tipo de preparación o instrucción previa.

3.13 Estándares de diseño.

La plantilla de la aplicación se concibe con apariencia de aplicación de escritorio. Compuesta por un banner que identifique el sitio con un logo y revele el propósito de la aplicación Web.

Se trabaja con tonalidades del color azul. Con pocos contrastes. Íconos que identifiquen los eventos o funcionalidades a cambio de carteles o mensajes.

Estándares de codificación

Apariencia de clases	Primera letra en mayúscula	Los nombres de las clases y las instancias de las mismas deben comenzar con la primera letra en mayúscula y el resto en minúscula, en caso de que sea un nombre compuesto se empleará notación PascalCasing*. Ejemplo: MiClase ().
Nombre de clases	Relacionados al propósito	El nombre empleado, debe permitir que con sólo leerlo se conozca el propósito de la clase o instancia de la misma.
Apariencia de atributos	Primera letra en minúscula	El nombre que se le da a los atributos de las clases debe comenzar con la primera letra en minúscula, en caso de que sea un nombre compuesto se empleará notación CamelCasing**.
Nombre de atributos	Nemotécnicos	El nombre empleado, debe permitir que con sólo leerlo se conozca el propósito del mismo dentro de la clase.
Apariencia de las funciones	Primera letra en mayúscula	Los nombres de las funciones deben comenzar con la primera letra en mayúscula y el resto en minúscula, en caso de que sea un nombre compuesto se empleará notación PascalCasing*.
Nombre de las funciones	Nemotécnicos	El nombre empleado, debe permitir que con sólo leerlo se conozca el propósito de la misma dentro de la clase.
Identación		
Objetivo: Lograr una estructura uniforme para los bloques de código así como para los diferentes niveles de anidamiento.		
2 espacios en blanco desde la izquierda en	Function Define	Se dejarán dos espacios en blanco desde la izquierda en las instrucciones antes mencionadas.
2 espacios en	Inicio y fin de bloque	Se recomienda dejar dos espacios en blanco desde la

blanco desde la referencia en		instrucción anterior para el inicio y fin de bloque {}. Lo mismo sucede para el caso de las instrucciones If, else, For, While, Do While, Switch, Foreach.
Niveles de anidación	Hasta 5 niveles	Se recomienda emplear hasta 5 niveles de anidación en instrucciones If, For, While.
Ejemplo de indentación		
<pre> <? require ('class/Interface.php'); class MiClase { function BuscaUnidad(\$nTabla, \$nFields, \$kIndice) { if (\$nTabla) { ... } for (...) { ... } } } ?> </pre>		
Comentarios, separadores, líneas y espacios en blanco		
Objetivo: Establecer un modo común para comentar el código de forma tal que sea comprensible con sólo leerlo una vez.		
Ubicación de comentarios	Al inicio de cada clase o función y al final de cada bloque de código.	Se recomienda comentar al inicio de la clase o función especificando el objetivo de la misma así como los parámetros que usa (especificar tipos de dato, y objetivo del parámetro) entre otras cosas. Y se comenta también cuando se cierran los ciclos, clases, instrucciones if y otras.
Separador de instrucciones	Se emplea el punto y coma.	Se recomienda usar el separador al final de cada instrucción y no en la línea de abajo. Ejemplo: define ("CONSTANT", "value1");
Líneas en	Se emplean antes de	Se recomienda dejar una línea en blanco antes de la

blanco	cada función.	definición de cada función para dar claridad al código.
Espacios en blanco	Entre operadores lógicos y aritméticos.	Se recomienda usar espacios en blanco entre estos operadores para lograr una mayor legibilidad en el código. Ejemplo: \$nTabla = 'nomproducto'; if ((\$nTabla) && (\$nFields))

Tabla 3.1 Estándares de codificación.

3.14 Conclusiones.

Al finalizar este capítulo quedaron desarrollados los diferentes procesos que son de gran utilidad para la implementación de la gestión de los objetivos y los responsables, facilitando una vista física y lógica de los artefactos empleados en el modelado. Quedaron fijados los estándares a emplear en cada una de las etapas de diseño y la correspondiente explicación de su funcionamiento. Se mostró de forma detallada mediante los distintos diagramas la funcionalidad del sistema para un mejor entendimiento por parte del usuario a la hora de interactuar con la aplicación.

CAPÍTULO 4. IMPLEMENTACIÓN

4.1 Introducción.

Este capítulo está referido al flujo de trabajo de implementación, el cual describe cómo los elementos del modelo del diseño se implementan en términos de componentes y cómo estos se organizan de acuerdo a los nodos específicos en el modelo de despliegue. Al terminar este capítulo se tendrá un modelo de despliegue y diagramas de componentes que esbozarán las diferentes situaciones en que se implantará la aplicación.

4.2 Modelo de Despliegue

El modelo de despliegue 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. El modelo de despliegue se utiliza como entrada fundamental en las actividades de diseño e implementación debido a que la distribución del sistema tiene una influencia principal en su diseño.

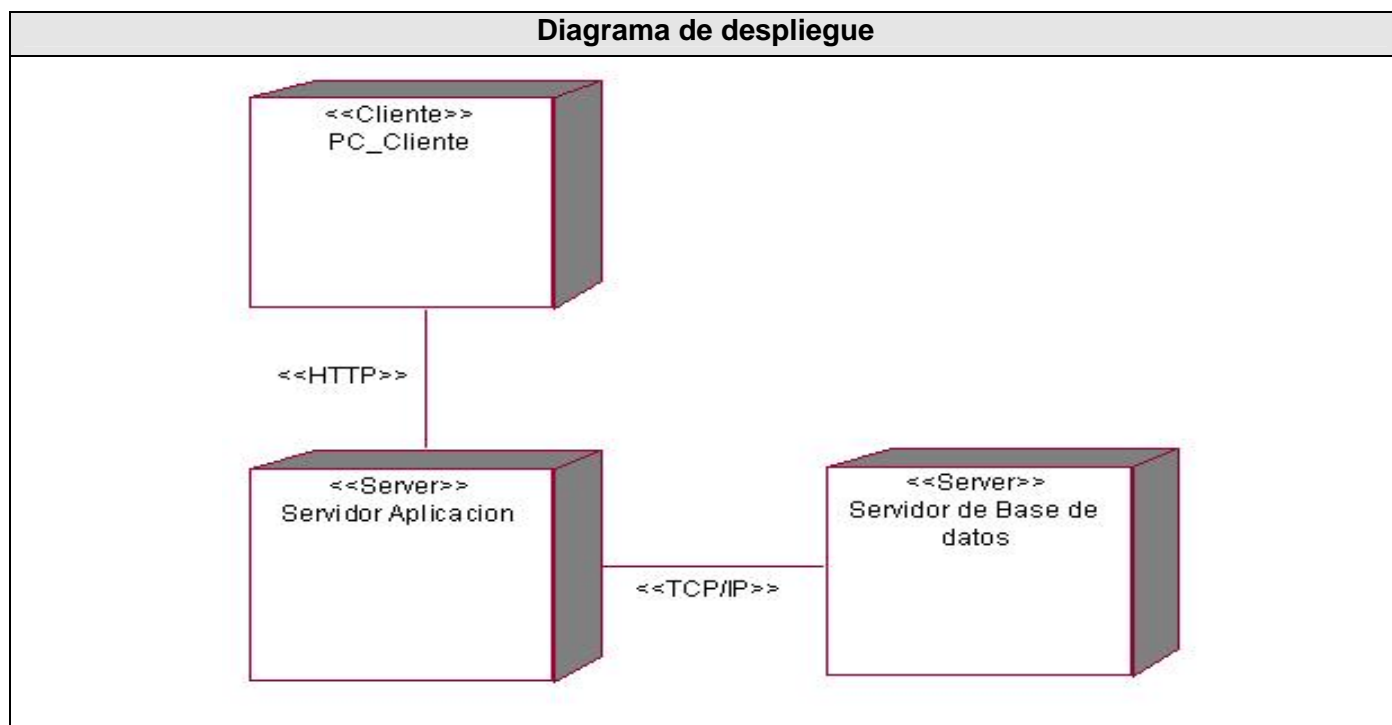


Figura 4.1. Modelo de despliegue.

4.3 Diagrama de componentes.

Es un diagrama que muestra un conjunto de elementos del modelo tales como componentes, subsistemas de implementación y sus relaciones. Se utilizan para modelar la vista estática de un sistema. Muestra la organización y las dependencias lógicas entre un conjunto de componentes software, sean éstos componentes de código fuente, librerías, binarios o ejecutables.

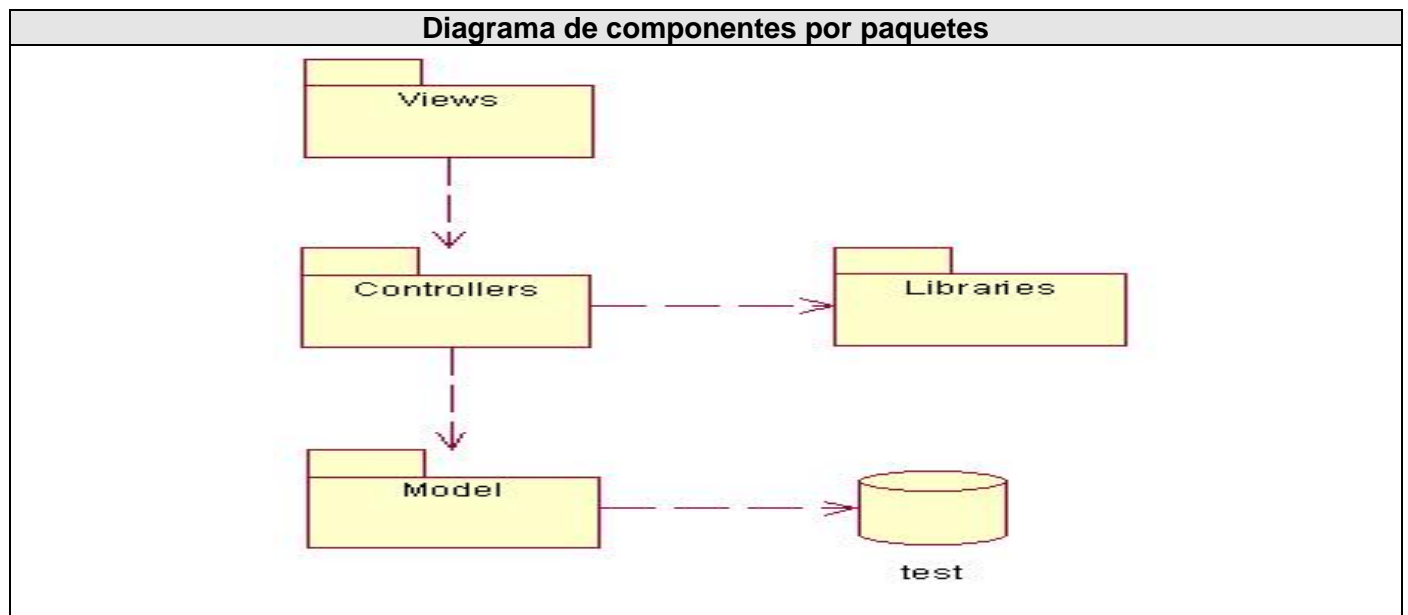


Figura 4.2. Diagrama de componentes por paquetes.

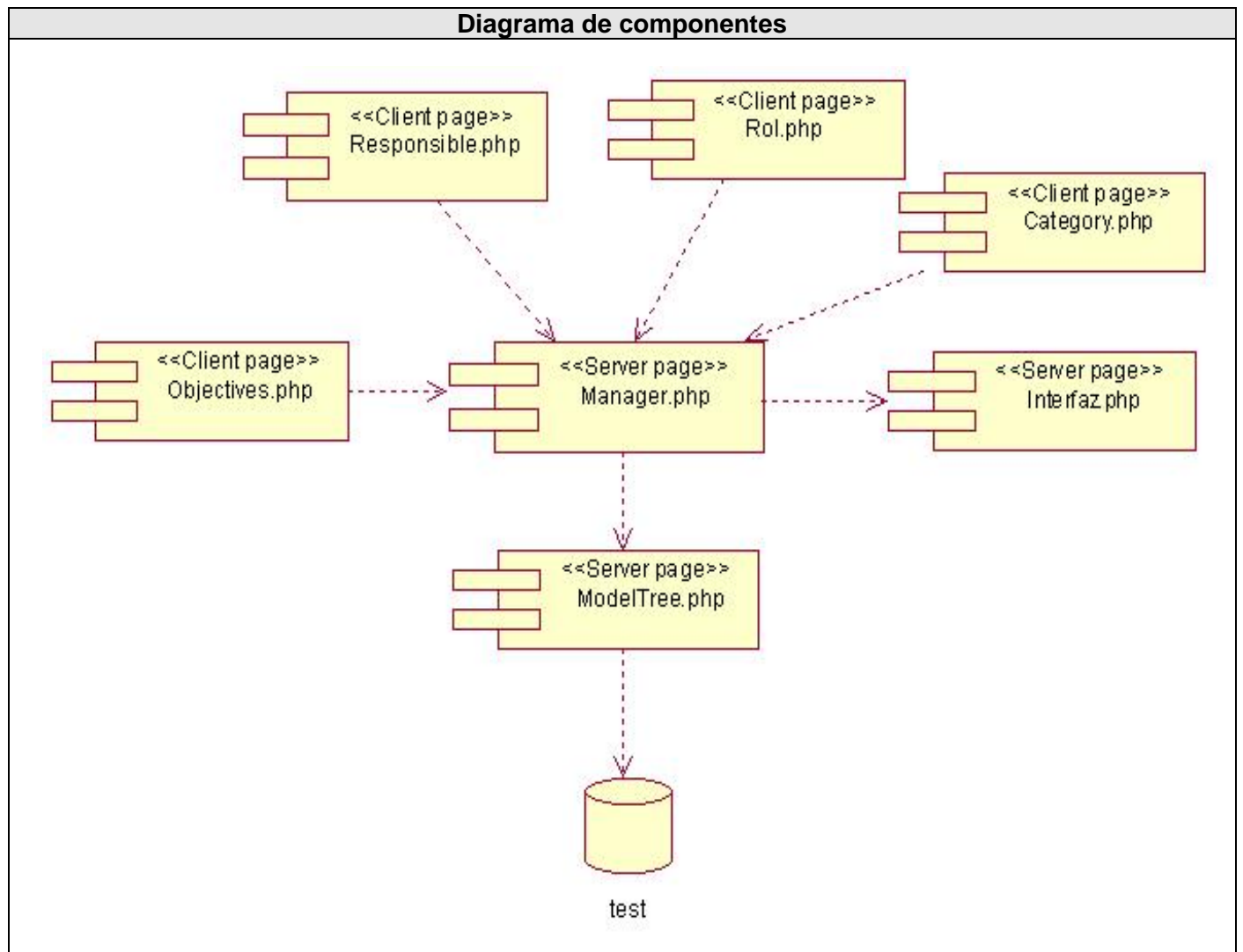


Figura 4.3. Diagrama de componentes.

4.4 Conclusiones.

En este capítulo se analizaron y desarrollaron los distintos diagramas correspondientes al flujo de implementación como son los diagramas de componentes y el de despliegue acorde al diseño e implementación del software trabajado, mostrando así, este último, la estructura del hardware sobre el que se ejecuta el sistema.

CAPÍTULO 5: ESTUDIO DE FACTIBILIDAD

5.1 Introducción.

Para la realización de este capítulo se tuvo en cuenta que a la hora de concebir un proyecto de cualquier tipo, con un tiempo de desarrollo y una fecha de entrega, se debe tener siempre presente los costos que se asumirán con su desarrollo conjuntamente con los beneficios que reportará. Como resultado de este análisis se obtiene el tiempo de desarrollo en meses y costo que se necesitan para desarrollar el proyecto. Además se realiza un análisis de los beneficios tangibles e intangibles que reporta el desarrollo de este sistema y con esta información se puede determinar la viabilidad.

5.2 Planificación basada en casos de uso.

5.2.1 Identificar los puntos de casos de usos desajustados.

$$UUCP = UAW + UUCW$$

Donde:

UUCP: Puntos de casos de uso sin ajustar.

UAV: Factor de peso de los actores sin ajustar.

UUCW: Factor de peso de los casos de uso sin ajustar.

5.2.2 Cálculo de UAV

Tipo	Descripción	Peso	Cant * peso
Simple	Otro sistema que interactúa con el sistema a desarrollar mediante una interfaz de programación (API, Application Programming Interface).	1	0*1
Medio	Otro sistema que interactúa con el sistema a desarrollar mediante un protocolo o una interfaz basada en texto.	2	0*2
Complejo	Una persona que interactúa con el sistema mediante una interfaz gráfica.	3	1*3
Total			3

Tabla 5.1 Factor de peso de los actores sin ajustar.

5.2.3 Cálculo de UUCW

Tipo	Descripción	Peso	Cant * peso
Simple	El Caso de Uso contiene de 1 a 3 transacciones.	5	0*5
Medio	El Caso de Uso contiene de 4 a 7 transacciones.	10	0*10
Complejo	El Caso de Uso contiene más de 8 transacciones.	15	4*15
		Total	60

Tabla 5.2 Factor de peso de los casos de uso sin ajustar.

$$UUCP = UAV + UUCW$$

$$UUCP = 3 + 60$$

$$UUCP = 63$$

5.3 Ajustar los puntos de casos de uso.

$$UCP = UUCP * TCF * EF$$

Donde:

UCP: Puntos de Casos de Uso ajustados.

UUCP: Puntos de Casos de Uso sin ajustar.

TCF: Factor de complejidad técnica.

EF: Factor de ambiente.

5.3.1 Cálculo de TCF

$$TCF = 0.6 + 0.01 * \sum (Peso_i * Valor_i)$$

Donde:

0: No presente o sin influencia.

1: Influencia incidental o presencia incidental.

2: Influencia moderada o presencia moderada.

3: Influencia media o presencia media.

4: Influencia significativa o presencia significativa.

5: Fuerte influencia o fuerte presencia.

$$TCF = 0.6 + 0.01 * \sum (Peso_i * Valor_i)$$

$$TCF = 0.6 + 0.01 * 38$$

TCF = 1.01

Factor	Descripción	Peso	Valor	Comentario	$\Sigma (\text{Peso}_i * \text{Valor}_i)$
T1	Sistema distribuido.	2	0	El sistema es centralizado.	0
T2	Objetivos de performance o tiempo de respuesta.	1	4	El tiempo de respuesta es bastante rápido.	4
T3	Eficiencia del usuario final.	1	4	Es necesario que el usuario final conozca el sistema.	4
T4	Procesamiento interno complejo.	1	4	Presenta cálculos complejos.	4
T5	El código debe ser reutilizable.	1	5	Se requiere que el código sea reutilizable.	5
T6	Facilidad de instalación.	0.5	3	Posee algunos requerimientos de instalación.	1.5
T7	Facilidad de uso.	0.5	5	Fácil de usar.	2.5
T8	Portabilidad.	2	5		10
T9	Facilidad de cambio.	1	4		4
T10	Concurrencia.	1	0	No hay concurrencia.	0
T11	Incluye objetivos especiales de seguridad.	1	1		1
T12	Provee acceso directo a terceras partes.	1	0	Los usuarios web no tienen acceso.	0
T13	Se requieren facilidades especiales de entrenamiento a los usuarios.	1	2	Pocos usuarios internos, sistema fácil de usar.	2
Total					38

Tabla 5.3 Factor de complejidad técnica

5.3.2 Cálculo de EF

$$EF = 1.4 - 0.03 * \sum (\text{Peso}_i * \text{Valor}_i)$$

Factor	Descripción	Peso	Valor	Comentario	$\sum (\text{Peso}_i * \text{Valor}_i)$
E1	Familiaridad con el modelo de proyecto utilizado.	1.5	3	El grupo está un poco familiarizado con el modelo.	4.5
E2	Experiencia en la aplicación.	0.5	1	La mayoría del grupo no tiene mucha experiencia en la aplicación.	0.5
E3	Experiencia en orientación a objetos.	1	4	Grupo programa en objetos.	4
E4	Capacidad del analista líder.	0.5	0	No se contrató a un especialista.	0
E5	Motivación.	1	4	El grupo está motivado.	4
E6	Estabilidad de los requerimientos.	2	2	Se esperan cambios.	4
E7	Personal part-time.	-1	4	El grupo todo el tiempo no es full-time.	-4
E8	Dificultad del lenguaje de programación.	-1	2	Se usará lenguaje PHP.	-2
Total					11

Tabla 5.4 Factor de ambiente

$$EF = 1.4 - 0.03 * \sum (\text{Peso}_i * \text{Valor}_i)$$

$$EF = 1.4 - 0.03 * 11$$

$$EF = 1.07$$

Luego

$$UCP = UUCP * TCF * EF$$

$$UCP = 63 * 0.98 * 1.07$$

$$UCP = 66.06$$

5.4 Calcular esfuerzo FT implementación.

$$E = UCP * CF$$

Donde:

E: esfuerzo estimado en horas-hombre

UCP: Puntos de Casos de Uso ajustados

CF: factor de conversión

5.4.1 Cálculo de CF

CF = 20 horas-hombre (si Total_{EF} ≤ 2)

CF = 28 horas-hombre (si Total_{EF} = 3 ó Total_{EF} = 4)

CF = abandonar o cambiar proyecto (si Total_{EF} ≥ 5)

Total_{EF} = Cant EF < 3 (entre E1, E6) + Cant EF > 3 (entre E7, E8)

Total_{EF} = 2 + 0

Total_{EF} = 2

CF = 20 horas-hombre (porque Total_{EF} ≤ 2)

Luego

$$E = UCP * CF$$

$$E = 66.06 * 20 \text{ horas hombre}$$

$$E = 1312.2 \text{ horas-hombre}$$

5.5 Esfuerzo de todo el proyecto

Actividad	% esfuerzo	Valor esfuerzo
Análisis	10%	328.05 horas-hombre
Diseño	20%	656.1 horas-hombre
Implementación	40%	1312.2 horas-hombre
Prueba	15%	492.08 horas-hombre
Sobrecarga	15%	492.08 horas-hombre
Total	100%	3280.5 horas-hombre

Tabla 5.5 Esfuerzo de todo el proyecto por flujos de trabajo.

Suponiendo que se trabaja 8 horas por día, y un mes tiene como promedio 30 días; la cantidad de horas que puede trabajar una persona en 1 mes es 240 horas.

Si $E_T = 3280.5$ horas-hombre y por cada 240 horas se tiene 1 mes, eso daría un $E_T = 13.67$ mes-hombre.

Es decir una persona puede terminar el proyecto en aproximadamente 14 meses.

5.5 Costo del Proyecto.

Se asume como salario promedio mensual \$50.00

CH: Cantidad de hombres.

Tiempo: Tiempo total del proyecto.

$CH = 2$ hombres

$CHM = CH * \text{Salario Promedio}$

$CHM = 2 * 50$

$CHM = 100.00$ \$/mes

$\text{Costo} = CHM * ET$

$\text{Costo} = 100 * 13.67$

$\text{Costo} = \$ 1367$

$\text{Tiempo} = ET / CH$

$\text{Tiempo} = 13.67 / 2$

$\text{Tiempo} = 6.84$ meses

De los resultados obtenidos se interpreta que con 2 hombres trabajando en el proyecto el mismo se desarrolla en aproximadamente 7 meses y su costo total se estima que sea \$ 1367.

5.6 Beneficios tangibles e intangibles. Debe cuantificarse los tangibles.

La implantación del sistema automatizado para la gestión de los objetivos y los responsables, sin lugar a dudas trae consigo una gran cantidad de beneficios, tanto tangibles como intangibles que se manifestarán a corto y largo plazo.

La gestación de este proyecto tiene como principal objetivo el mejoramiento de las actividades que se realizan en las empresas cubanas. Los beneficios que reportará a corto plazo son principalmente de carácter intangibles, entre los que se pueden citar:

- Permite contar con una aplicación flexible y de interfaz agradable.

- Permite obtener los listados de los objetivos y de los responsables de una forma más eficiente.
- Brinda la posibilidad de trabajar con facilidad logrando así mayor eficiencia y rapidez en el flujo de trabajo.
- Beneficiará la seguridad e integridad de los datos que se obtengan.

En cuanto a los beneficios tangibles, se puede asegurar que mejorará el rendimiento y la rentabilidad de las empresas cubanas, pues la realización de la gestión de los objetivos y los responsables, ganará en rapidez y eficiencia.

5.7 Análisis de costo.

En cualquier país desarrollar un producto informático cuesta dinero, recursos y esfuerzo. Por eso, para justificar su desarrollo depende de los beneficios que reportaría su implantación y utilización. Los beneficios pueden ser económicos y de orden social, estos últimos son de tanta importancia como los primeros. El sistema que se propone está dirigido a las empresas cubanas, una vez implantado el sistema éste contribuirá a aumentar la eficiencia de las actividades que se realizan en las mismas, al disminuir el tiempo necesario a emplear fundamentalmente en la gestión de los objetivos y los responsables.

La tecnología utilizada para el desarrollo del sistema es totalmente libre, por tanto no hay que incurrir en gastos en el pago de licencias de uso. El sistema es portable por lo que un cambio de plataforma para la implantación del mismo es viable y factible, y no hay que incurrir en muchos cambios debido a la estructuración en capas de los procesos del negocio que se diseñaron.

Realizando un análisis de los costos del proyecto, los numerosos beneficios que reporta su implantación, detallados con anterioridad, se puede concluir que su implementación es factible.

5.8 Conclusiones

En este capítulo se describió el estudio de factibilidad correspondiente al sistema propuesto, teniendo en cuenta el costo estimado y los beneficios que reportará al ser implantado. Se llegó a la conclusión que es factible implementar la herramienta propuesta por los beneficios que reporta, los bajos costos que conlleva y por contribuir a mejorar el proceso de gestión de los objetivos y los responsables de las actividades en las empresas cubanas.

CONCLUSIONES

Los problemas existentes en las empresas cubanas están siendo tratados cada vez más por las nuevas tecnologías para darle solución. Con este trabajo se presentó una aplicación web para gestionar los objetivos y los responsables de las perspectivas de un Cuadro de Mando Integral. Con ello se logrará una mayor rapidez en el procesamiento de la información correspondiente a dicha gestión, disminuirán los errores que se cometen actualmente en las empresas cubanas, se logrará agilizar el flujo de trabajo y en consecuencia habrá una mayor productividad y control de los recursos con que cuenta la entidad.

El sistema desarrollado empleó la metodología RUP para el modelado de todas las fases del proyecto. Se realizó al mismo tiempo una ardua labor en cuanto al tratamiento de errores, logrando evitar la entrada de datos incoherentes. Se solucionó el problema de realizar análisis de la información de manera sistemática y tener en cada momento detalles sobre el control de la empresas; lográndose así, que en lo adelante se pueda responder a instancias superiores con brevedad y veracidad y contar con información requerida para la toma de decisiones.

El desarrollo de este software propuesto dio cumplimiento al objetivo trazado en este trabajo pues se automatizó el módulo gestión de los objetivos y responsables de las perspectivas del Cuadro de Mando Integral (CMI) para apoyar el servicio empresarial cubano, ya que los requerimientos soportaron al sistema, y los casos de uso solucionaron las necesidades funcionales del mismo, reafirmando así la utilidad y validez de emplear las tecnologías informáticas para apoyar las labores que se desarrollan en cualquier tipo de esfera. Además, cumple con las exigencias de usabilidad, disponibilidad, agradable interfaz, facilidad para aprender a usarlo y garantiza su sostenibilidad y fácil mantenimiento.

RECOMENDACIONES

Se recomienda:

- Continuar el estudio del Cuadro de Mando Integral con el objetivo de añadir nuevas funcionalidades al sistema.
- Implementar un módulo referido a la administración del sistema, así como realizar el mapa estratégico que debe llevar un CMI.
- Proponer, tras aprobar un desempeño exitoso, la utilización y generalización de este sistema en las diferentes empresas cubanas que lleva a cabo el CMI.

BIBLIOGRAFÍA

- Achour, Mehdi. Manual de PHP, 2006. [Disponible en: <http://www.php.net/manual/es/>] [Fecha de consulta: Febrero 2007]
- Acrónimo recursivo, 2007. [Disponible en: http://es.wikipedia.org/wiki/Acr%C3%B3nimo_recursivo] [Fecha de consulta: Mayo 2007]
- Álvarez, Miguel Angel. Evaluando Zend Studio, 2003. [Disponible en: <http://www.maestrosdelweb.com/editorial/zendstudio/>] [Fecha de consulta: Febrero 2007]
- Aplicaciones web a la medida. [Disponible en: http://www.intellia.com.mx/esp/servicios/aplicaciones_web_a_la_medida.php] [Fecha de consulta: Febrero 2007]
- Aplicación web, 2007. [Disponible en: http://es.wikipedia.org/wiki/Aplicaci%C3%B3n_web] [Fecha de consulta: Febrero 2007]
- Applet, 2007. [Disponible en: <http://es.wikipedia.org/wiki/Applet>] [Fecha de consulta: Mayo 2007]
- Archivo informático, 2007. [Disponible en: <http://es.wikipedia.org/wiki/Fichero>] [Fecha de consulta: Mayo 2007]
- Balanced Scorecard Software: Dialog Strategy: Características, 2006. [Disponible en: <http://www.cacitgroup.com/productos/faqs%20nuevo%20dialog.htm>] [Fecha de consulta: Enero 2007]
- Banner, 2007. [Disponible en: <http://es.wikipedia.org/wiki/Banner>] [Fecha de consulta: Mayo 2007]
- Características del Software Strategic, 2006. [Disponible en: http://www.estrategiaempresarial.com/sw_caracteristica.asp] [Fecha de consulta: Enero 2007]
- Cliente liviano, 2007. [Disponible en: http://es.wikipedia.org/wiki/Cliente_ligero] [Fecha de consulta: Mayo 2007]
- Código abierto, 2007. [Disponible en: http://es.wikipedia.org/wiki/C%C3%B3digo_abierto] [Fecha de consulta: Mayo 2007]
- Conde Pagan, Elda. El Cuadro de Mando Integral como herramienta del Control de Gestión, 2007 [Disponible en: http://www.wikilearning.com/el_cuadro_de_mando_integral_como_herramienta_del_control_de_gestion_estrategica-wkccp-12627-4.htm] [Fecha de consulta: Enero 2007]

- Conde Pagan, Elda. El Cuadro de Mando Integral en organizaciones no lucrativas, 1997. [Disponible en: <http://www.monografias.com/trabajos31/cuadro-mando-integral/cuadro-mando-integral.shtml#cuadro>] [Fecha de consulta: Enero 2007]
- Dávila, Antonio. Nuevas herramientas de control: El Cuadro de Mando Integral, 1999. [Disponible en: <http://www.ee-iese.com/pdf/afondo4.pdf>] [Fecha de consulta: Enero 2007]
- Decide Soft. [Disponible en: http://www.decidesoft.net/esp/info_c.htm] [Fecha de consulta: Enero 2007]
- Delphos, 2005. [Disponible en: <http://www.businesssoft.com.ni/delphos.htm>] [Fecha de consulta: Enero 2007]
- Dreamweaver. [Disponible en: <http://www.desarrolloweb.com/articulos/332.php>] [Fecha de consulta: Febrero 2007]
- Drupal, 2007. [Disponible en: <http://es.wikipedia.org/wiki/Drupal>] [Fecha de consulta: Mayo 2007]
- El servidor web Apache. [Disponible en: <http://es.tldp.org/Tutoriales/doc-servir-web-escuela/doc-servir-web-escuela-html/apache.html#AEN384>] [Fecha de consulta: Febrero 2007]
- EMS SQL Manager for MySQL, 2007. [Disponible en: <http://sqlmanager.net/en/products/mysql/manager/features>] [Fecha de consulta: Febrero 2007]
- Estilos y Patrones en la Estrategia de Arquitectura de Microsoft, 2004. [Disponible en: http://www.microsoft.com/spanish/msdn/arquitectura/roadmap_arq/style.asp] [Fecha de consulta: Febrero 2007]
- Estrategia y Cuadro de Mando Integral en la práctica. [Disponible en: http://www.improven-consultores.com/paginas/documentos_gratuitos/estrategiaCMI.php] [Fecha de consulta: Enero 2007]
- Ferrer Castañedo, Marta y León Toirac, Roxana. El Cuadro de Mando Integral, 2007 [Disponible en: <http://www.gestiopolis.com/canales3/ger/cmi.htm>] [Fecha de consulta: Enero 2007]
- Framework, 2007. [Disponible en: <http://es.wikipedia.org/wiki/Framework>] [Fecha de consulta: Mayo 2007]
- Hardware, 2007. [Disponible en: <http://es.wikipedia.org/wiki/Hardware>] [Fecha de consulta: Mayo 2007]
- Hilo de ejecución, 2007. [Disponible en: http://es.wikipedia.org/wiki/Hilo_en_sistemas_operativos] [Fecha de consulta: Mayo 2007]

- Holismo, 2007. [Disponible en: <http://es.wikipedia.org/wiki/Hol%C3%ADstico>] [Fecha de consulta: Mayo 2007]
- Host, 2007. [Disponible en: <http://es.wikipedia.org/wiki/Host>] [Fecha de consulta: Mayo 2007]
- Iframe, 2007. [Disponible en: <http://es.wikipedia.org/wiki/Iframe>] [Fecha de consulta: Mayo 2007]
- Inteligencia empresarial y Cuadro de Mando Integral, 2005. [Disponible en: <http://www.fordes.co.cu/xtras/eventos/SemanasTec/memorias/Primera/lsemana.htm>] [Fecha de consulta: Enero 2007]
- JavaScript, 2007. [Disponible en: http://64.233.167.104/search?q=cache:X2PM-r6bm0J:es.wikipedia.org/wiki/JavaScript+Java+Script&hl=es&ct=clnk&cd=1&gl=cu&lr=lang_es] [Fecha de consulta: Febrero 2007]
- Lenguaje interpretado, 2007. [Disponible en: http://es.wikipedia.org/wiki/Lenguaje_interpretado] [Fecha de consulta: Mayo 2007]
- Licencia dual, 2007. [Disponible en: http://es.wikipedia.org/wiki/Licencia_dual] [Fecha de consulta: Mayo 2007]
- Logotipo, 2007. [Disponible en: <http://es.wikipedia.org/wiki/Logotipo>] [Fecha de consulta: Mayo 2007]
- MMORPG, 2007. [Disponible en: <http://es.wikipedia.org/wiki/MMORPG>] [Fecha de consulta: Mayo 2007]
- Modelo Vista Controlador, 2007. [Disponible en: http://es.wikipedia.org/wiki/Modelo_Vista_Controlador] [Fecha de consulta: Febrero 2007]
- Multimedia, 2007. [Disponible en: <http://es.wikipedia.org/wiki/Multimedia>] [Fecha de consulta: Mayo 2007]
- MySQL, 2007. [Disponible en: <http://es.wikipedia.org/wiki/MySQL>] [Fecha de consulta: Febrero 2007]
- Navegador web, 2007. [Disponible en: http://es.wikipedia.org/wiki/Navegador_web] [Fecha de consulta: Mayo 2007]
- Parche (software), 2006. [Disponible en: http://es.wikipedia.org/wiki/Parches_software] [Fecha de consulta: Mayo 2007]
- Pecos, Daniel. PostGreSQL vs. MySQL. [Disponible en: http://www.netpecos.org/docs/mysql_postgres/index.html] [Fecha de consulta: Febrero 2007]

- Plugin, 2007. [Disponible en: <http://es.wikipedia.org/wiki/Plugin>] [Fecha de consulta: Mayo 2007]
- Proceso Unificado de Rational, 2007. [Disponible en: http://es.wikipedia.org/wiki/Rational_Unified_Process] [Fecha de consulta: Febrero 2007]
- Script, 2007. [Disponible en: <http://es.wikipedia.org/wiki/Script>] [Fecha de consulta: Mayo 2007]
- Servidor, 2007. [Disponible en: <http://es.wikipedia.org/wiki/Servidor>] [Fecha de consulta: Mayo 2007]
- Servidor Web apache. [Disponible en: <http://www.linalco.com/apache.html>] [Fecha de consulta: Febrero 2007]
- Sistema de gestión de base de datos, 2007. [Disponible en: http://es.wikipedia.org/wiki/Sistema_de_gesti%C3%B3n_de_base_de_datos] [Fecha de consulta: Febrero 2007]
- Van Der Henst S, Christian. Las novedades de Dreamweaver 8, 2005. [Disponible en: <http://www.maestrosdelweb.com/actualidad/2334/>] [Fecha de consulta: Febrero 2007]
- Ventaja e inconvenientes de las aplicaciones web, 2005. [Disponible en: <http://www.avidos.net/blogold/aplicaciones-web/>] [Fecha de consulta: Febrero 2007]
- Vizcaíno, Aurora y Caballero, Ismael. Una Herramienta CASE para ADOO: Rational Rose. [Disponible en: http://alarcos.inf-cr.uclm.es/per/fgarcia/isoftware/doc/LabTr1_RationalRose.pdf] [Fecha de consulta: Febrero 2007]
- Webmail, 2007. [Disponible en: <http://es.wikipedia.org/wiki/Webmail>] [Fecha de consulta: Mayo 2007]
- Wiki, 2007. [Disponible en: <http://es.wikipedia.org/wiki/Wiki>] [Fecha de consulta: Mayo 2007]
- YUI FAQ, 2007. [Disponible en: <http://developer.yahoo.com/yui/articles/faq/>] [Fecha de consulta: Febrero 2007]

REFERENCIAS BIBLIOGRÁFICAS

- [1]. Rivadeneira, R.M., Trabajo en el VII Congreso Nacional de Control Interno en las entidades del estado, 2001, p. 12.
- [2]. Resolución Económica del V Congreso del Partido Comunista de Cuba. 1998: La Habana, Cuba. p. 25
- [3]. Oracle Corporation, 2007. [Disponible en: http://en.wikipedia.org/wiki/Oracle_Corporation]
- [4]. Pecos, Daniel. MySQL, 2002. [Disponible en: http://www.netpecos.org/docs/mysql_postgres/x57.html#AEN71]
- [5]. Linalco. Servidor Apache, 2005. [Disponible en: <http://www.linalco.com/apache.html>]
- [6]. Schuller, J., *Aprendiendo UML en 24 horas*. Prentice Hall ed. 2000.
- [7]. Dorado Perea, Carles. Aprender a Aprender Estrategias y Técnicas, 1996. [Disponible en: <http://www.xtec.cat/~cdorado/cdora1/esp/disseny.htm>]
- [8]. Framework, 2007. [Disponible en: <http://es.wikipedia.org/wiki/Framework>]
- [9]. [Disponible en: <http://php.cujae.edu.cu/modules.php?op=modload&name=Reviews&file=index&req=showcontent&id=9>]
- [10]. Nideaderedes. Code Igniter-framework PHP, 2007. [Disponible en: <http://nideaderedes.urlansoft.com/2007/02/27/code-igniter-framework-php/>]
- [11]. Microsoft, Software License Agreement (BSD License).

ANEXOS

Anexo 1. Descripción textual de los casos de uso del negocio.

CUN Gestionar Objetivos.

Nombre del Caso de Uso	Gestionar Objetivos.	
Actores	Directivo	
Propósito	Identificar el proceso de la gestión de los objetivos según las necesidades de la unidad de negocio.	
Resumen	El caso de uso se inicia cuando el directivo solicita la creación de los objetivos, luego el trabajador procede a crear dicho objetivo llevando a cabo un conjunto de actividades, además se toman decisiones hasta dar un resultado de dicho proceso y de esta forma finaliza el caso de uso.	
Curso Normal de los eventos		
Acciones del Actor	Respuestas del proceso del negocio	
1- El Directivo solicita la creación de un nuevo objetivo.		
	2- El Jefe del Área recibe la solicitud.	
	3- El Jefe del Área crea el objetivo.	
	4- El Jefe del Área selecciona los responsables.	
	5- El Jefe del Área selecciona los indicadores.	
	6- El Jefe del Área selecciona el plan de acción.	
	7- El Jefe del Área selecciona las estrategias.	
	8- El Jefe del Área muestra la lista de los objetivos.	
9- El Directivo recibe la lista.		
10- El Directivo conserva la información.		
Curso Alternativo de los eventos		
Acción 10.1	Si desea hacer algún cambio el Directivo solicita dicha modificación.	
	El Jefe del Área recibe la solicitud del cambio.	
	El Jefe del Área actualiza los objetivos.	

	El Jefe del Área muestra nueva lista de objetivos.
	El Directivo recibe dicha lista y se termina el caso de uso.
Prioridad	Crítico
Mejoras	
Otros	

CUN Gestionar Responsables.

Nombre del Caso de Uso	Gestionar Responsables.	
Actores	Directivo	
Propósito	Identificar el proceso de la gestión de los responsables según las necesidades de la unidad de negocio.	
Resumen	El caso de uso se inicia cuando el directivo solicita la lista de los responsables, luego el trabajador procede a gestionar dicha lista llevando a cabo un conjunto de actividades, además se toman decisiones hasta dar un resultado de dicho proceso y de esta forma finaliza el caso de uso.	
Curso Normal de los eventos		
Acciones del Actor	Respuestas del proceso del negocio	
1- El Directivo solicita la lista de responsables.		
	2- El Jefe del Área recibe la solicitud.	
	3- El Jefe del Área verifica si hay roles y categorías.	
	4- El Jefe del Área crea los responsables.	
	5- El Jefe del Área selecciona los roles y categorías.	
	6- El Jefe del Área muestra la lista de los responsables.	
7- Recibe la lista.		
8- Conserva la información.		
Curso Alternativo de los eventos		
Acción 3.1	Si no existen roles y categorías el Jefe del Área crea los roles y las categorías y	

	procede a la acción 4.
Acción 7.1	Si se desea hacer algún cambio el Directivo solicita la modificación.
	El Jefe del Área recibe dicha solicitud.
	El Jefe del Área actualiza los responsables.
	El Jefe del Área muestra nueva lista de responsables.
	El Directivo recibe dicha lista y se termina el caso de uso.
Prioridad	Crítico
Mejoras	
Otros	

Anexo 2. Diagramas de actividades.

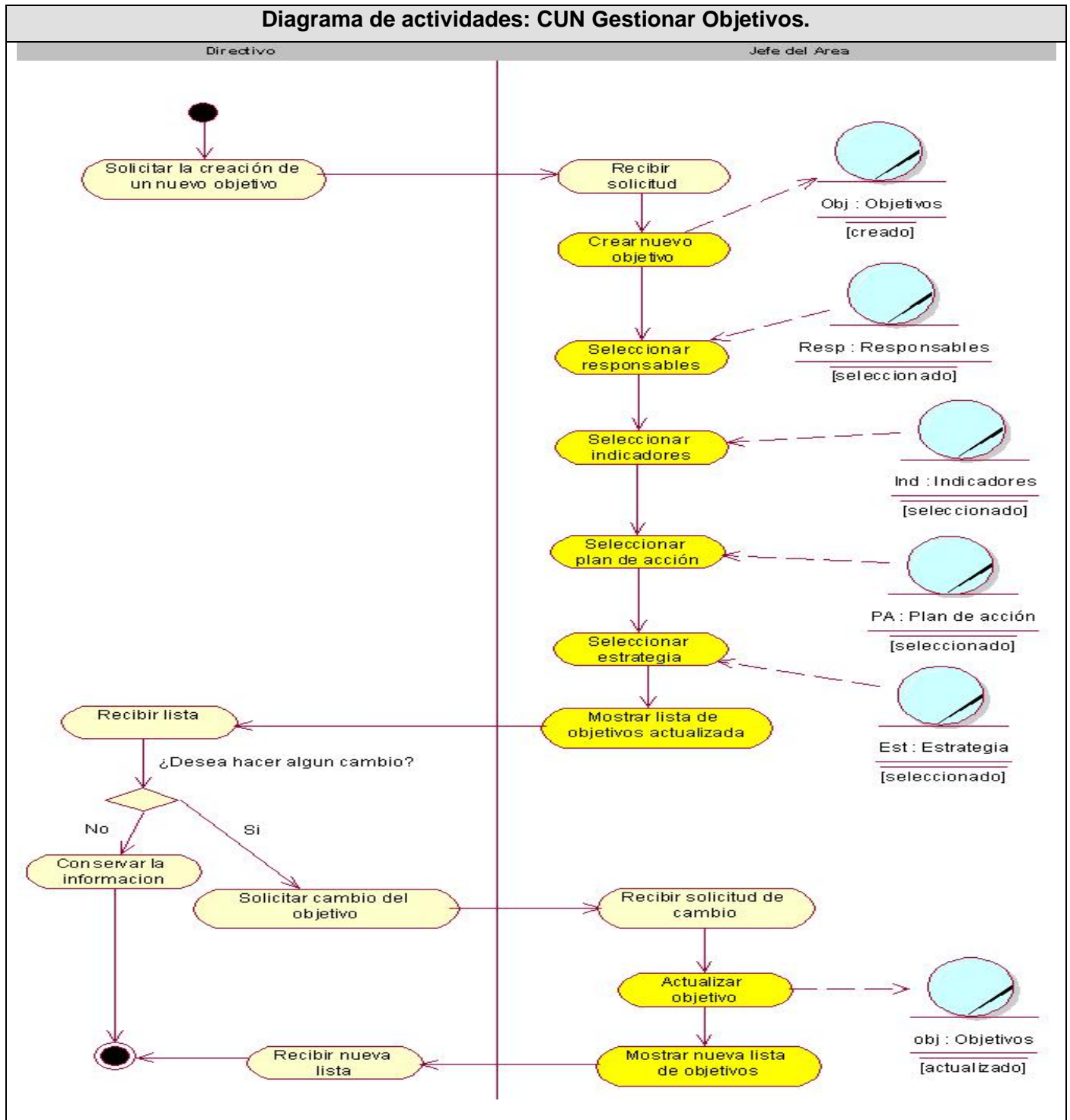
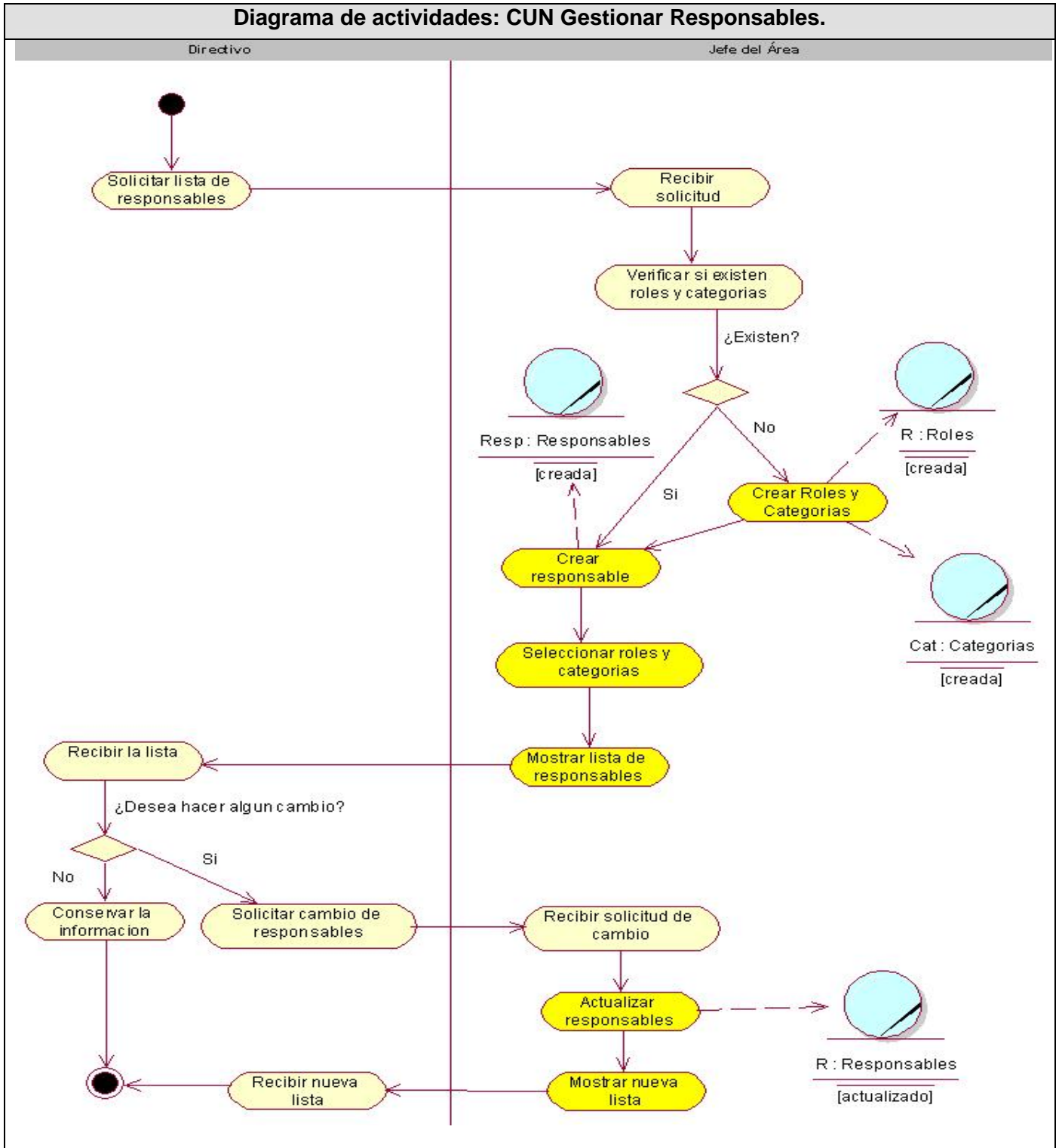
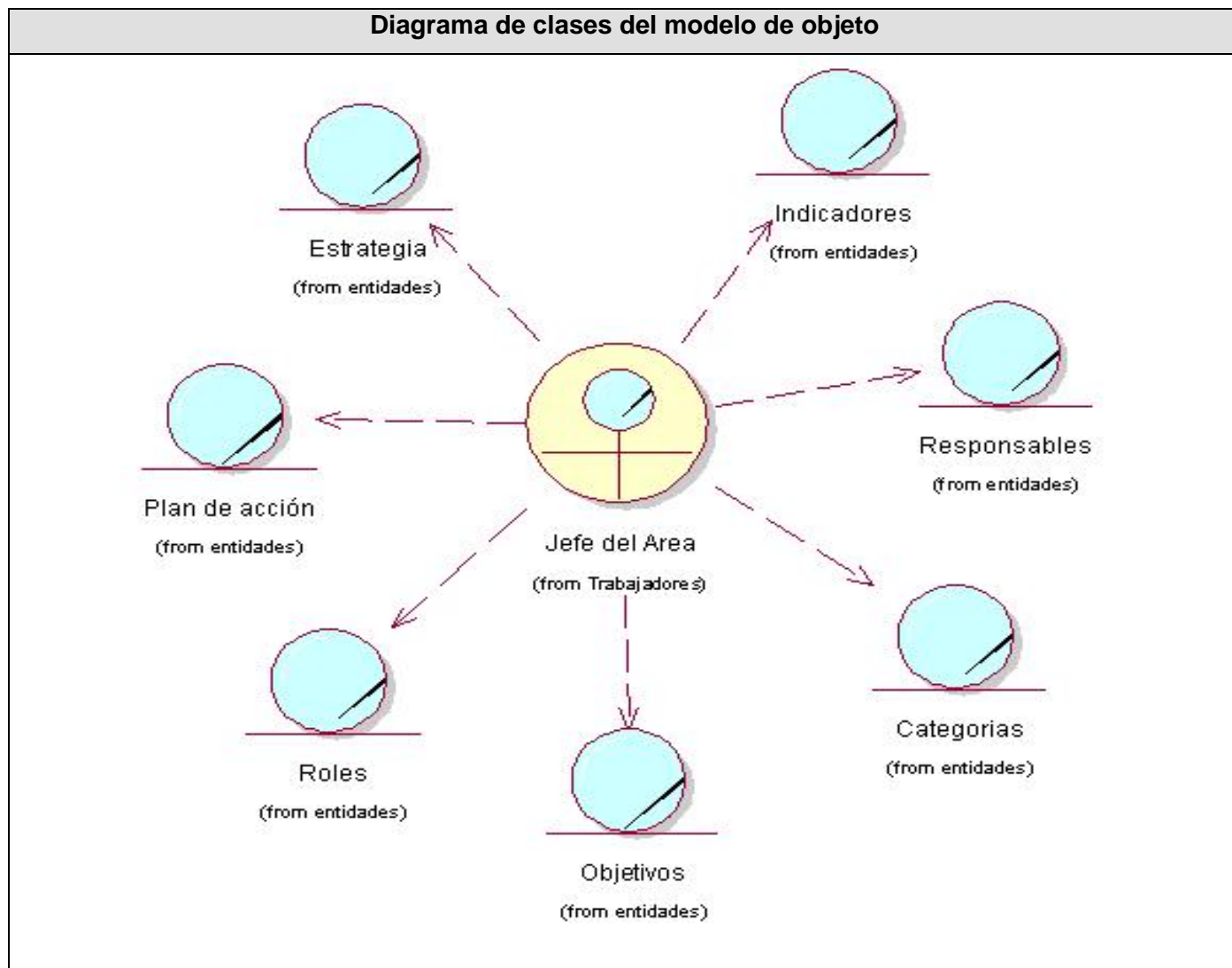


Diagrama de actividades: CUN Gestionar Responsables.



Anexo 3. Modelo de objetos



Anexo 4. Descripción textual de los casos de uso del sistema

CUS Gestionar Objetivos

Nombre del Caso de Uso	Gestionar Objetivos
Actores	Directivo (inicia)
Propósito	Permitir al directivo gestionar (crear, modificar, eliminar o seleccionar) los objetivos.
Resumen	El caso de uso se inicia cuando el directivo desea gestionar los objetivos, el sistema muestra las opciones que puede realizar en dicho caso de uso, el directivo selecciona la opción que desea: crear, modificar, eliminar o seleccionar, luego el sistema lleva a cabo la acción solicitada y termina el caso de uso del sistema.
Referencias	R.1.1, R.1.2, R.1.3 , R.1.4, R.1.5, R.1.6, R.1.7, R.1.8, R.1.9
Precondiciones	Para las siguientes acciones: Modificar: Debe existir algún objetivo registrado en el sistema. Eliminar: Debe existir algún objetivo registrado en el sistema. Seleccionar: Deben existir al menos un responsable, una estrategia, un plan de acción, un indicador, una perspectiva y un negocio.
Poscondiciones	Para las siguientes acciones: Crear: Se crea un objetivo. Modificar: Se modifica un objetivo. Eliminar: Se elimina totalmente un objetivo.
Prototipo	

Curso Normal de los eventos

Acciones del Actor	Respuestas del sistema
1. El directivo selecciona la opción gestionar objetivos.	1.1 El sistema muestra las opciones para crear, modificar, eliminar objetivos y seleccionar indicadores, responsables, estrategia y plan de acción, negocio y perspectiva.

Sección 1: Crear objetivos

1. El directivo desea crear un objetivo.	1.1 El sistema muestra una interfaz para entrar los datos referentes al objetivo.
2. El directivo teclea los datos del objetivo que desea crear, incluyendo nombre del objetivo, el planteamiento de dicho objetivo, selecciona la fecha estimada, a que tipo de negocio se están refiriendo, la perspectiva al cual pertenece, estrategia, plan de acción y el estado.	2.1 El sistema valida los datos y crea el objetivo. El Caso de Uso comienza nuevamente.

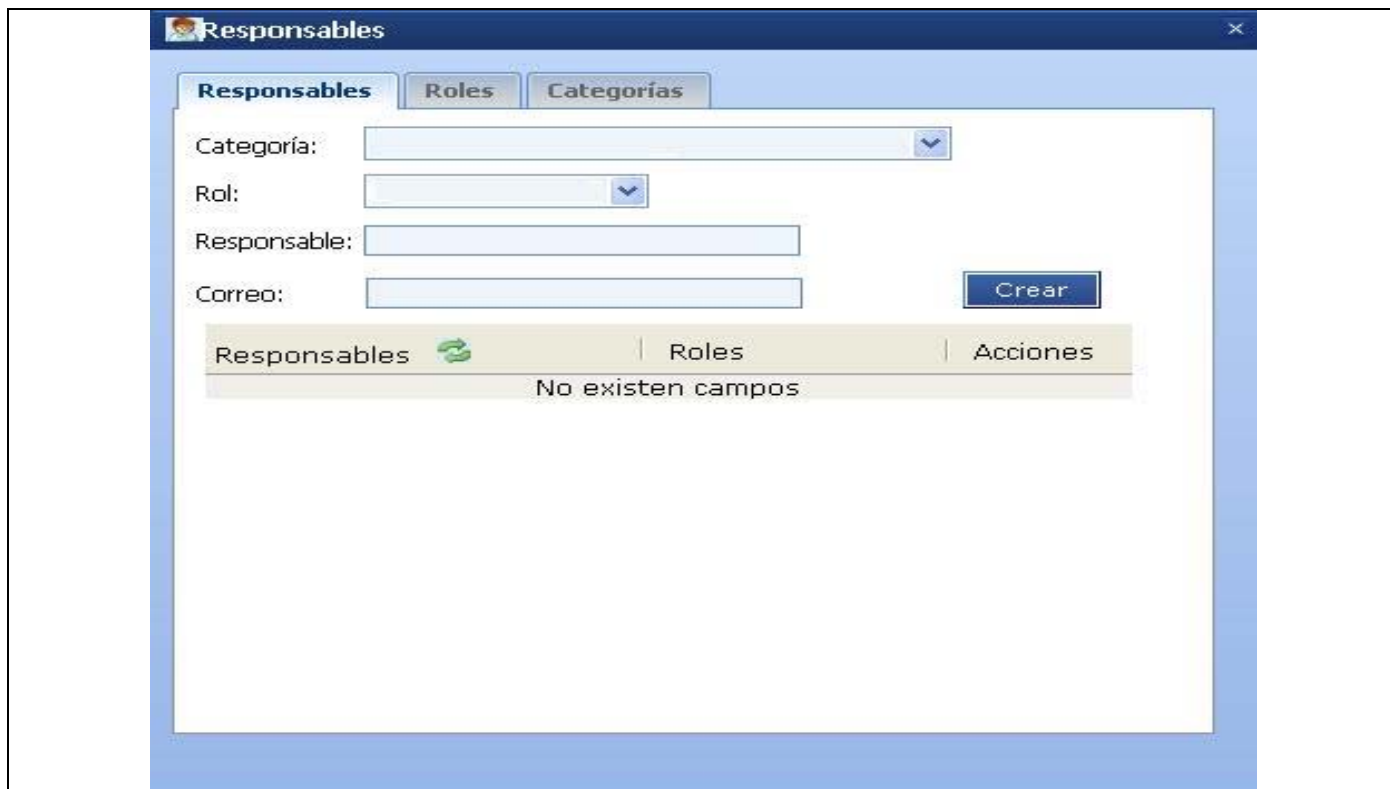
Curso alternativo de los eventos

Acción 2.1	El Directivo puede seleccionar la opción de responsables, luego el sistema valida los
------------	---

	datos y crea el objetivo. El Caso de Uso comienza nuevamente.	
	El Directivo puede seleccionar la opción de indicadores, luego el sistema valida los datos y crea el objetivo. El Caso de Uso comienza nuevamente.	
Sección 2: Modificar objetivo		
1. El directivo desea modificar un objetivo.	1.1 El sistema muestra el listado de objetivos que existen.	
2. El directivo selecciona el objetivo que desea modificar y activa esta opción.	2.1 El sistema le muestra una interfaz con los datos del objetivo para modificar.	
3. El directivo realiza los cambios necesarios al objetivo.	3.1 El sistema valida los datos y modifica el objetivo, mostrando el listado de los objetivos actualizado. El Caso de Uso comienza nuevamente.	
Curso alternativo de los eventos		
Acción 1.1	Si no existe ningún objetivo en la base de datos, no se podrá realizar la acción 2 en adelante y el caso de uso concluye automáticamente.	
Sección 3: Eliminar objetivo		
1. El directivo desea eliminar un objetivo.	1.1 El sistema muestra los objetivos que existen.	
2 El directivo selecciona el objetivo que desea eliminar y activa esta opción.	2.1 El sistema elimina el objetivo seleccionado, mostrando un mensaje para asegurar la acción y luego muestra el listado de los objetivos actualizado. El Caso de Uso comienza nuevamente.	
Curso alternativo de los eventos		
Acción 1.1	Si no existe ningún objetivo en la base de datos, no se podrá realizar la acción 2 en adelante y el caso de uso concluye automáticamente.	
Prioridad	Crítico	

CUS Gestionar Responsables

Nombre del Caso de Uso	Gestionar Responsables
Actores	Directivo (inicia)
Propósito	Permitir al directivo gestionar (crear, modificar, eliminar) los responsables.
Resumen	El caso de uso se inicia cuando el directivo desea gestionar los responsables, el sistema muestra las opciones que puede realizar en dicho caso de uso, el directivo selecciona la opción que desea: crear, modificar o eliminar, luego el sistema lleva a cabo la acción solicitada y termina el caso de uso del sistema.
Referencias	R.2.1, R.2.2, R.2.3, R.2.4, R.2.5
Precondiciones	Para las siguientes acciones: Crear: Debe existir al menos un rol y una categoría registrado en el sistema. Modificar: Debe existir algún responsable registrado en el sistema. Eliminar: Debe existir algún responsable registrado en el sistema.
Poscondiciones	Para las siguientes acciones: Crear: Se crea un responsable. Modificar: Se modifica un responsable. Eliminar: Se elimina totalmente un responsable.
Prototipo	



Curso Normal de los eventos	
Acciones del Actor	Respuestas del sistema
1. El directivo selecciona la opción gestionar responsable.	1.1 El sistema muestra las opciones para crear, modificar y eliminar responsable.
Sección 1: Crear responsable	
1. El directivo desea crear un responsable.	1.1 El sistema muestra una interfaz para entrar los datos referentes al responsable.
2. El directivo teclea los datos del responsable que desea crear, incluyendo nombre de la persona, correo electrónico y selecciona la categoría y el rol al que pertenece dicho responsable.	2.1 El sistema valida los datos y crea el responsable. El Caso de Uso comienza nuevamente.
Curso alternativo de los eventos	
Acción 1.1	Si no existe al menos una categoría y un rol no se podrá efectuar la acción 2 en adelante y el caso de uso termina.

Sección 2: Modificar responsable	
1. El directivo desea modificar un responsable.	1.1 El sistema muestra el listado de responsables que existen.
2. El directivo selecciona el responsable que desea modificar y activa esta opción.	2.1 El sistema le muestra una interfaz con los datos del responsable para modificar.
3. El directivo realiza los cambios necesarios al responsable.	3.1 El sistema valida los datos y modifica el responsable, mostrando la lista de los responsables actualizada. El Caso de Uso comienza nuevamente.
Curso alternativo de los eventos	
Acción 1.1	Si no existe ningún responsable en la base de datos, no se podrá realizar la acción 2 en adelante y el caso de uso concluye automáticamente.
Sección 3: Eliminar responsable	
1. El directivo desea eliminar un responsable.	1.1 El sistema muestra los responsables que existen.
2 El directivo selecciona el responsable que desea eliminar y activa esta opción.	2.1 El sistema elimina el responsable seleccionado mostrando un mensaje para asegurar la acción y luego muestra el listado de los responsables actualizado. El Caso de Uso comienza nuevamente.
Curso alternativo de los eventos	
Acción 1.1	Si no existe ningún responsable en la base de datos, no se podrá realizar la acción 2 en adelante y el caso de uso concluye automáticamente.
Prioridad	Crítico

CUS Gestionar Categoría

Nombre del Caso de Uso	Gestionar Categoría
Actores	Directivo (inicia)
Propósito	Permitir al directivo gestionar (crear, modificar, eliminar) las categorías.
Resumen	El caso de uso se inicia cuando el directivo desea gestionar las categorías, el sistema muestra las opciones que puede realizar en dicho caso de uso, el directivo selecciona la

	opción que desea: crear, modificar o eliminar, luego el sistema lleva a cabo la acción solicitada y termina el caso de uso del sistema.
Referencias	R.3.1, R.3.2, R.3.3
Precondiciones	Para las siguientes acciones: Modificar: Debe existir alguna categoría registrada en el sistema. Eliminar: Debe existir alguna categoría registrada en el sistema.
Poscondiciones	Para las siguientes acciones: Crear: Se crea una categoría. Modificar: Se modifica una categoría. Eliminar: Se elimina totalmente una categoría.

Prototipo



Curso Normal de los eventos

Acciones del Actor	Respuestas del sistema
1. El directivo selecciona la opción gestionar categoría.	1.1 El sistema muestra las opciones para crear, modificar y eliminar categorías.
Sección 1: Crear categoría	
1. El directivo desea crear una categoría.	1.1 El sistema muestra una interfaz para entrar los

		datos referentes a la categoría.
2. El directivo teclea los datos de la categoría que desea crear, incluyendo nombre de la categoría.		2.1 El sistema valida los datos y crea la categoría. El Caso de Uso comienza nuevamente.
Curso alternativo de los eventos		
Sección 2: Modificar categoría		
1. El directivo desea modificar una categoría.		1.1 El sistema muestra el listado de categorías que existen.
2. El directivo selecciona la categoría que desea modificar y activa esta opción.		2.1 El sistema le muestra una interfaz con los datos de la categoría para modificar.
3. El directivo realiza los cambios necesarios a la categoría.		3.1 El sistema valida los datos y modifica la categoría, mostrando la lista de las categorías actualizada. El Caso de Uso comienza nuevamente.
Curso alternativo de los eventos		
Acción 1.1	Si no existe ninguna categoría en la base de datos, no se podrá realizar la acción 2 en adelante y el caso de uso concluye automáticamente.	
Sección 3: Eliminar categoría		
1. El directivo desea eliminar una categoría.		1.1 El sistema muestra las categorías que existen.
2 El directivo selecciona la categoría que desea eliminar y activa esta opción.		2.1 El sistema elimina la categoría seleccionada mostrando un mensaje para asegurar la acción y luego muestra el listado de las categorías actualizado. El Caso de Uso comienza nuevamente.
Curso alternativo de los eventos		
Acción 1.1	Si no existe ninguna categoría en la base de datos, no se podrá realizar la acción 2 en adelante y el caso de uso concluye automáticamente.	
Prioridad	Crítico	

CUS Gestionar Rol

Nombre del Caso de Uso	Gestionar Rol
Actores	Directivo (inicia)
Propósito	Permitir al directivo gestionar (crear, modificar, eliminar) los roles.
Resumen	El caso de uso se inicia cuando el directivo desea gestionar los roles, el sistema muestra las opciones que puede realizar en dicho caso de uso, el directivo selecciona la opción que desea: crear, modificar o eliminar, luego el sistema lleva a cabo la acción solicitada y termina el caso de uso del sistema.
Referencias	R.4.1, R.4.2, R.4.3
Precondiciones	Para las siguientes acciones: Modificar: Debe existir algún rol registrado en el sistema. Eliminar: Debe existir algún rol registrado en el sistema.
Poscondiciones	Para las siguientes acciones: Crear: Se crea un rol. Modificar: Se modifica un rol. Eliminar: Se elimina totalmente un rol.

Prototipo

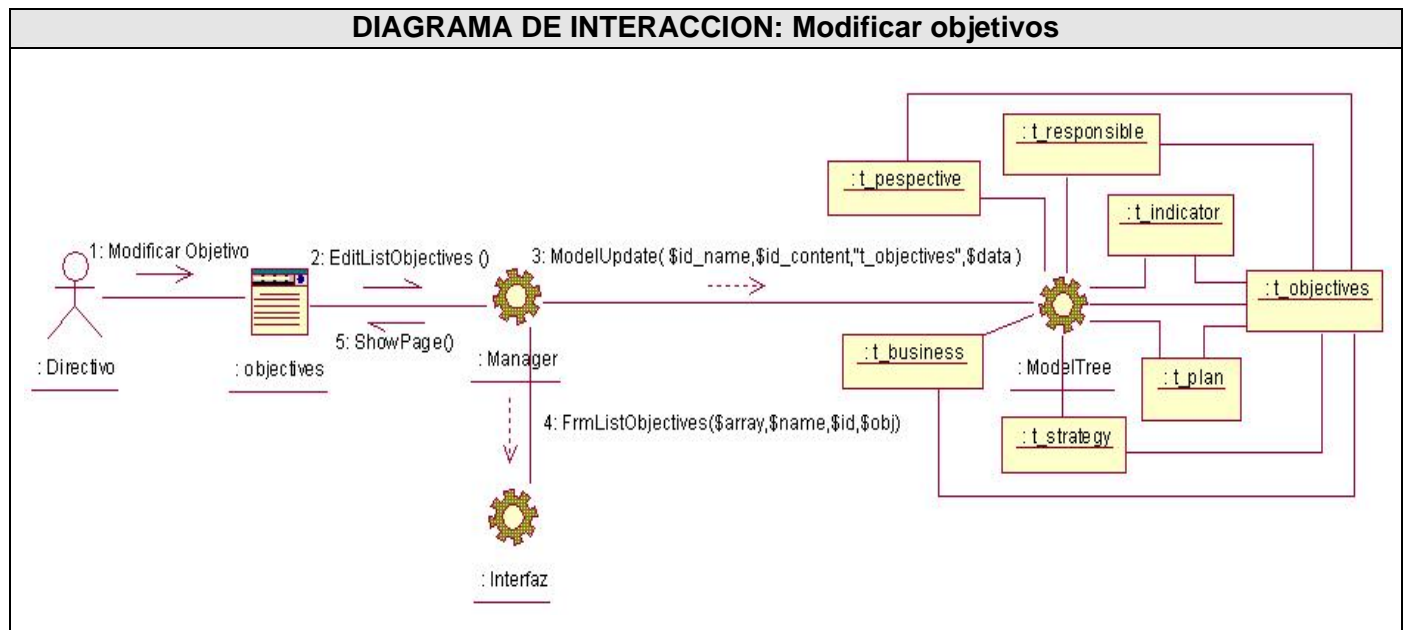
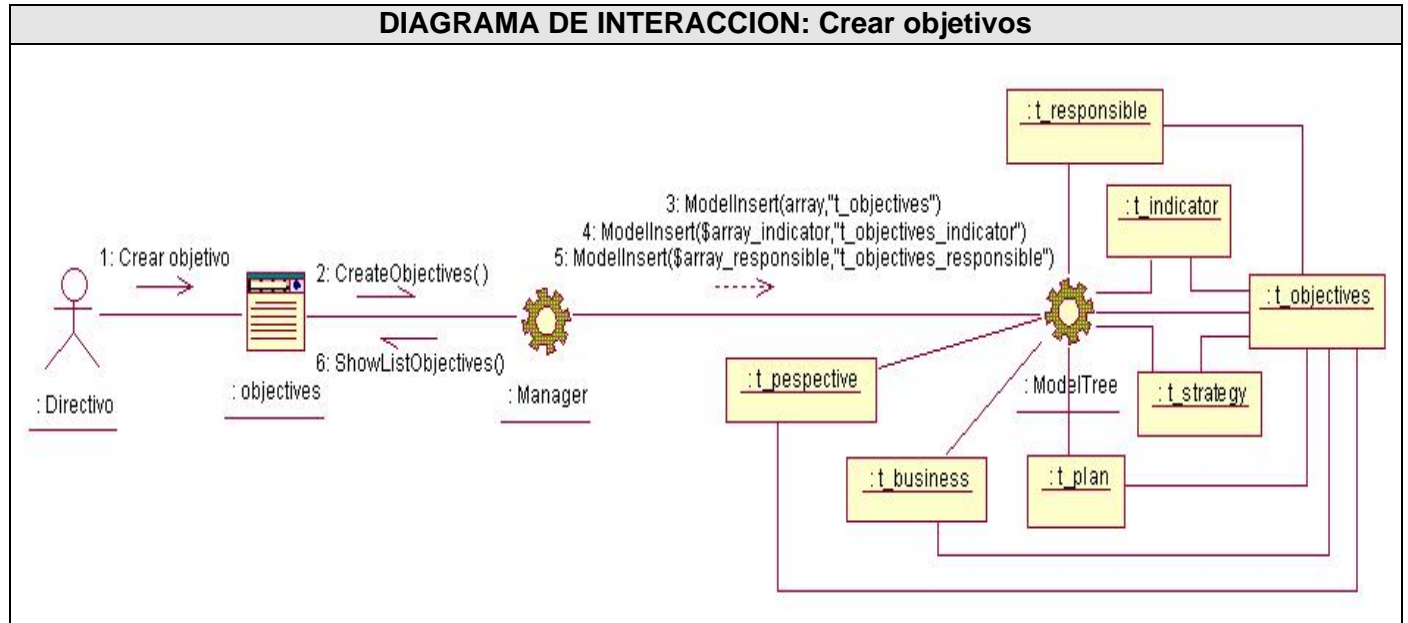


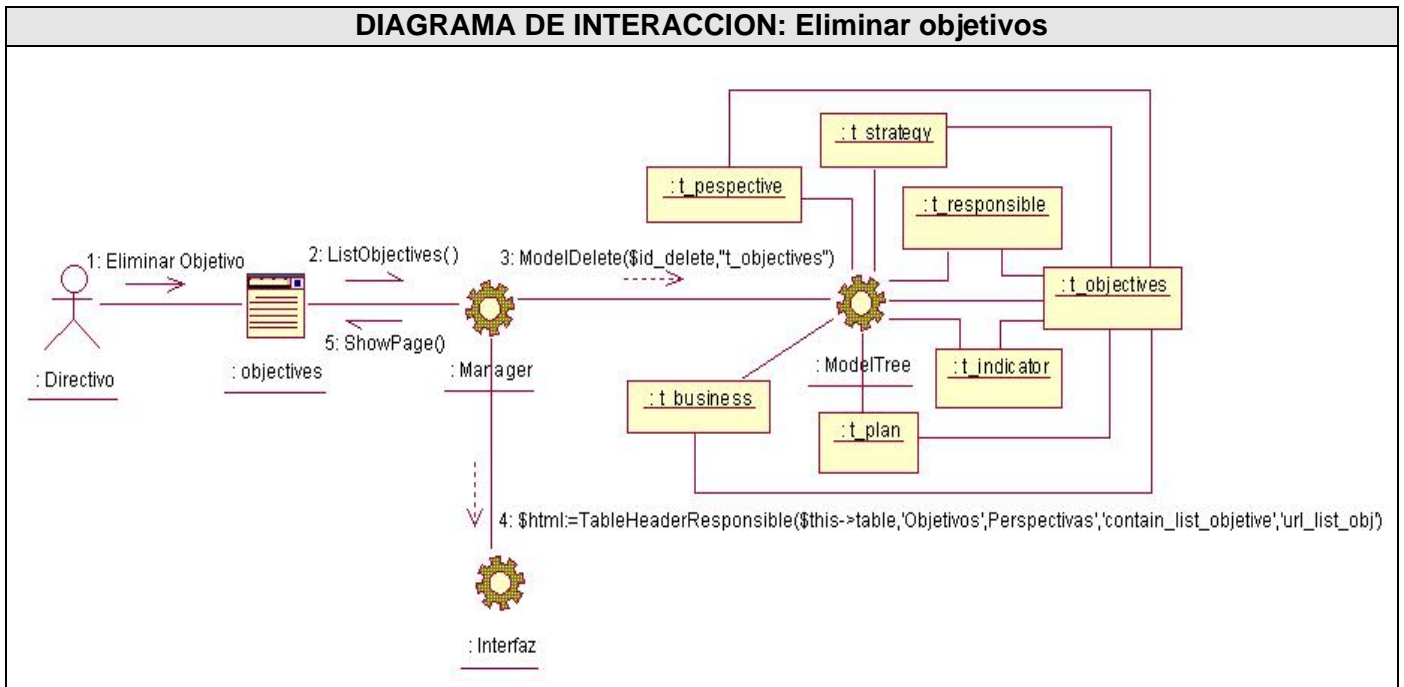
Curso Normal de los eventos

Acciones del Actor		Respuestas del sistema
1. El directivo selecciona la opción gestionar rol.		1.1 El sistema muestra las opciones para crear, modificar y eliminar roles.
Sección 1: Crear rol		
1. El directivo desea crear un rol.		1.1 El sistema muestra una interfaz para entrar los datos referentes al rol.
2. El directivo teclea los datos del rol que desea crear, incluyendo nombre del rol.		2.1 El sistema valida los datos y crea el rol. El Caso de Uso comienza nuevamente.
Curso alternativo de los eventos		
Sección 2: Modificar rol		
1. El directivo desea modificar un rol.		1.1 El sistema muestra el listado de rol que existen.
2. El directivo selecciona el rol que desea modificar y activa esta opción.		2.1 El sistema le muestra una interfaz con los datos del rol para modificar.
3. El directivo realiza los cambios necesarios al rol.		3.1 El sistema valida los datos y modifica el rol, mostrando la lista de los roles actualizada. El Caso de Uso comienza nuevamente.
Curso alternativo de los eventos		
Acción 1.1	Si no existe ningún rol en la base de datos, no se podrá realizar la acción 2 en adelante y el caso de uso concluye automáticamente.	
Sección 3: Eliminar rol		
1. El directivo desea eliminar un rol.		1.1 El sistema muestra los roles que existen.
2 El directivo selecciona el rol que desea eliminar y activa esta opción.		2.1 El sistema elimina el rol seleccionado mostrando un mensaje para asegurar la acción y luego muestra el listado de los roles actualizado. El Caso de Uso comienza nuevamente.
Curso alternativo de los eventos		
Acción 1.1	Si no existe ningún rol en la base de datos, no se podrá realizar la acción 2 en adelante y el caso de uso concluye automáticamente.	
Prioridad	Crítico	

Anexo 5. Diagramas de interacción del diseño

CUS Gestionar Objetivos





CUS Gestionar Responsables

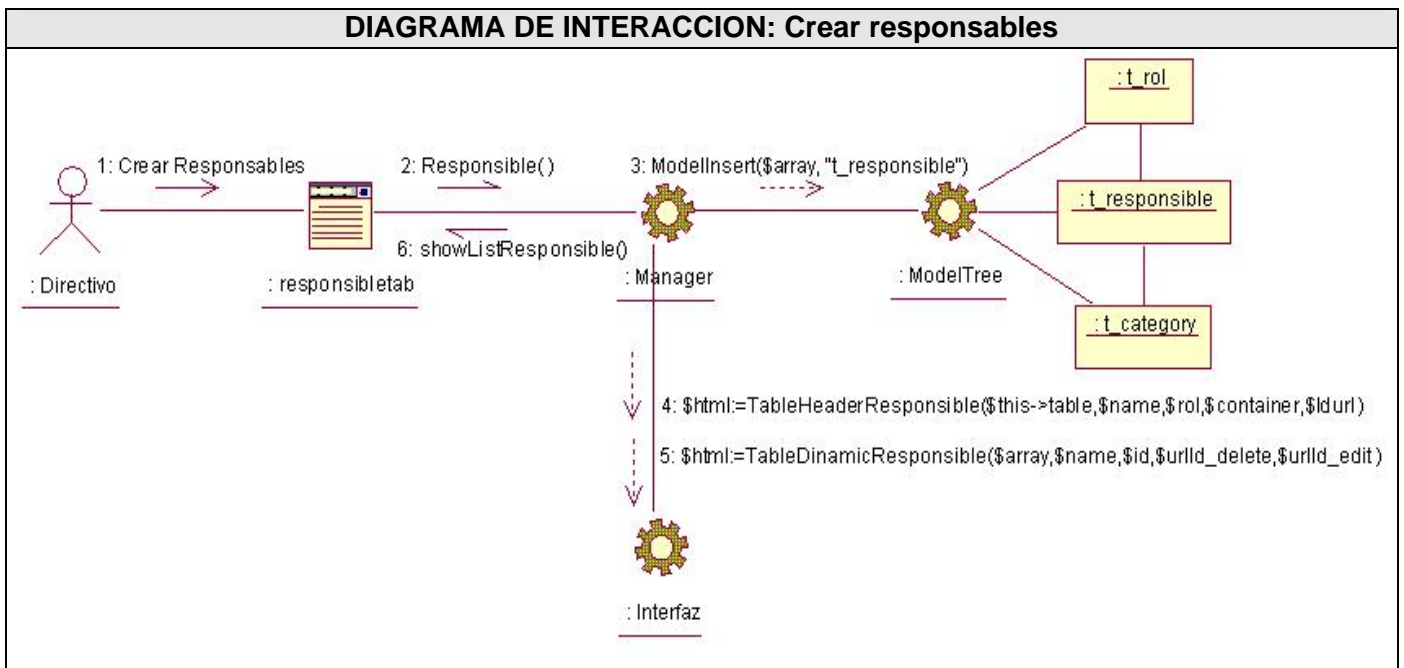


DIAGRAMA DE INTERACCION: Modificar responsables

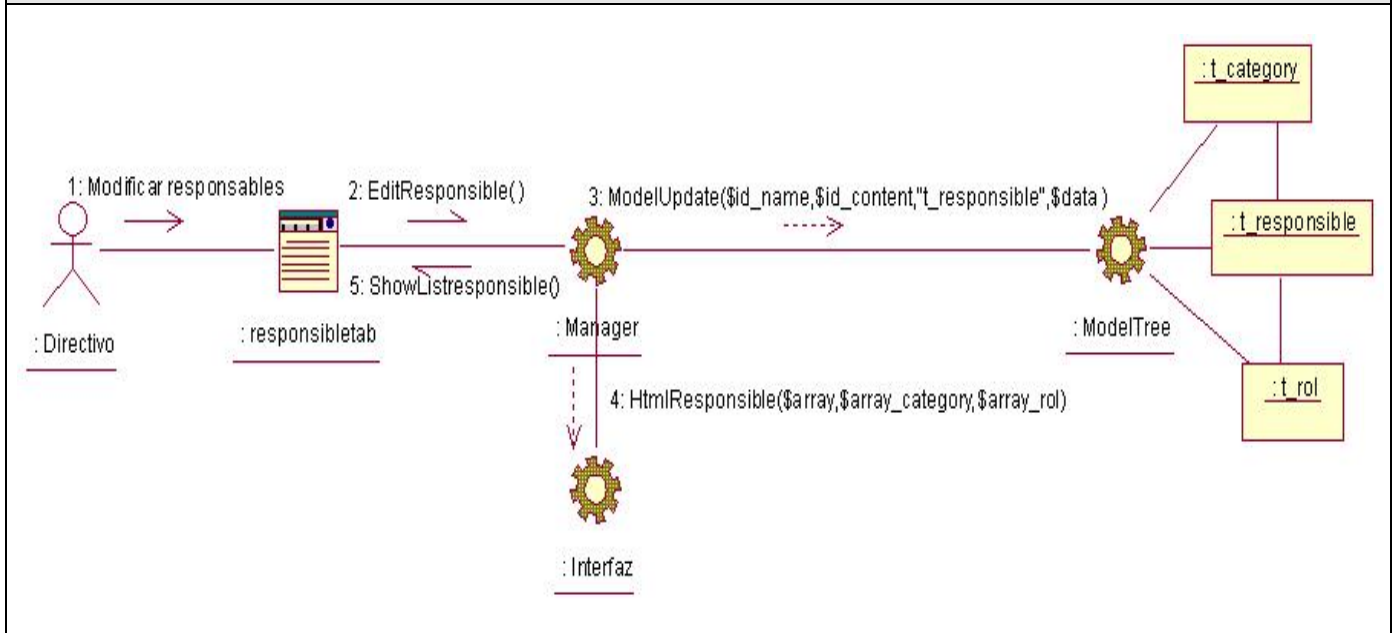
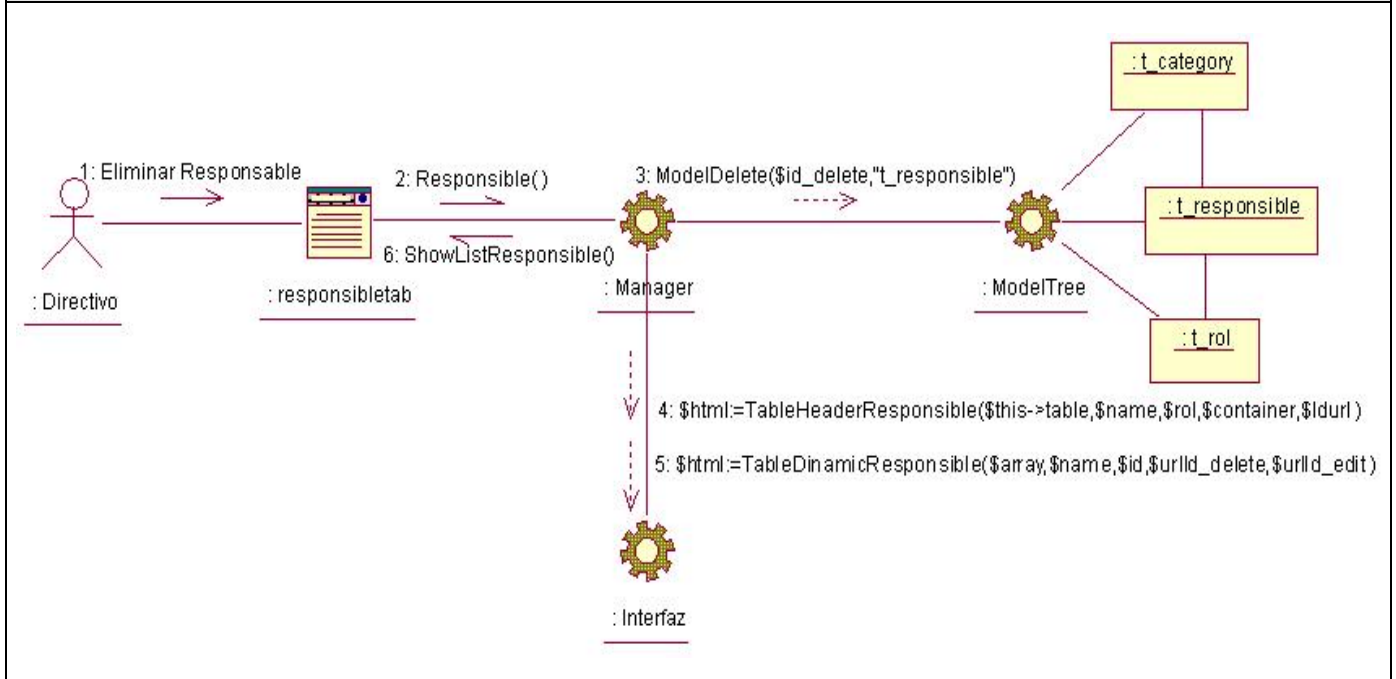
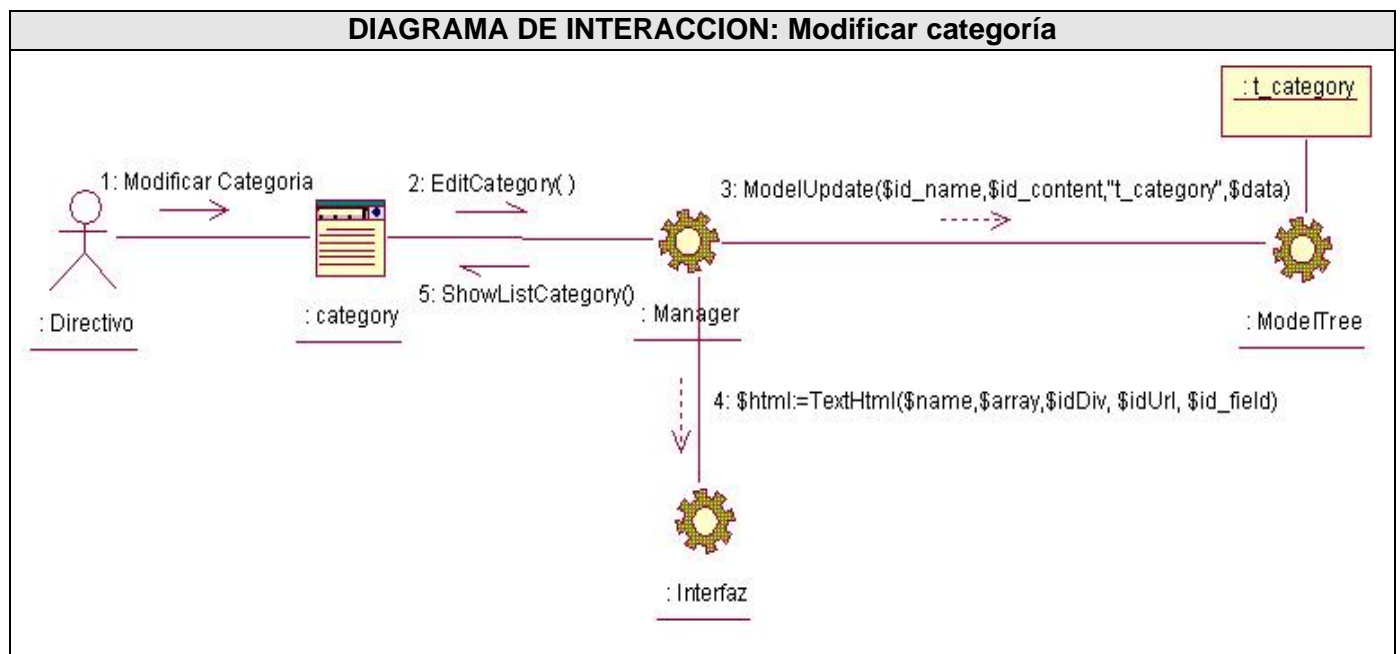
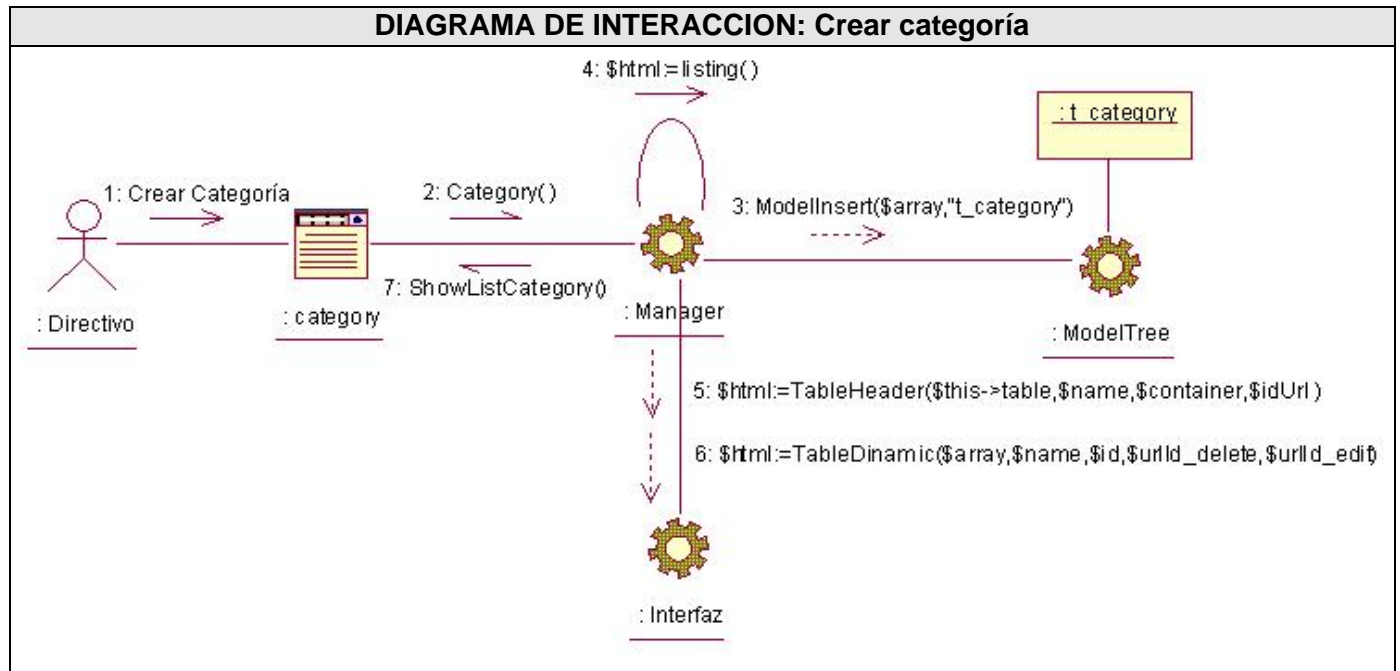
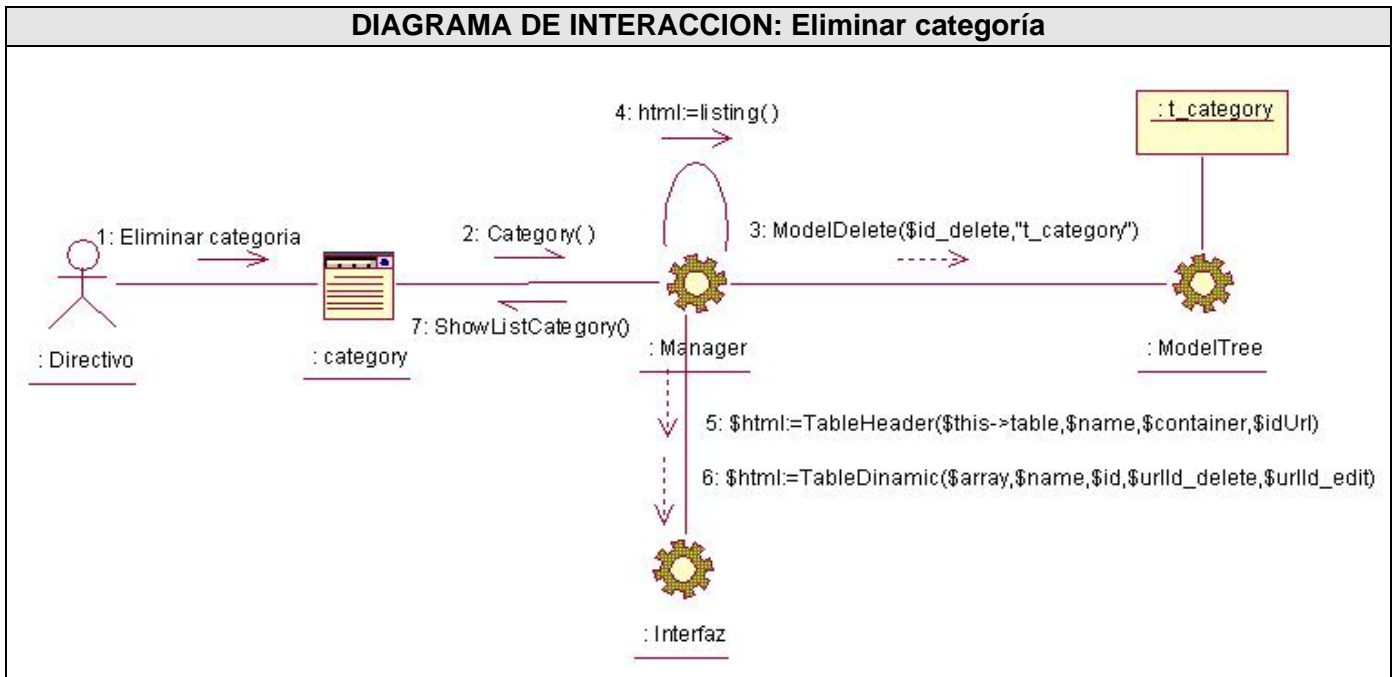


DIAGRAMA DE INTERACCION: Eliminar responsables



CUS Gestionar Categoría





CUS Gestionar Rol

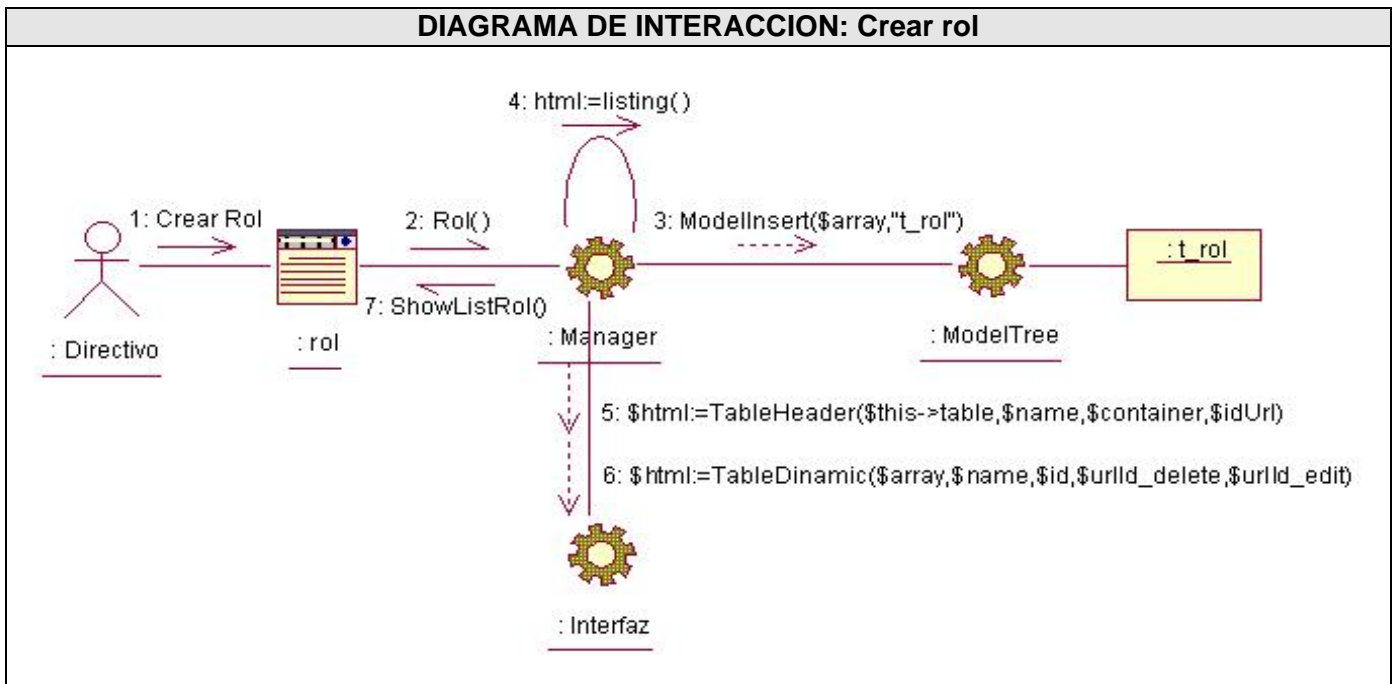


DIAGRAMA DE INTERACCION: Modificar rol

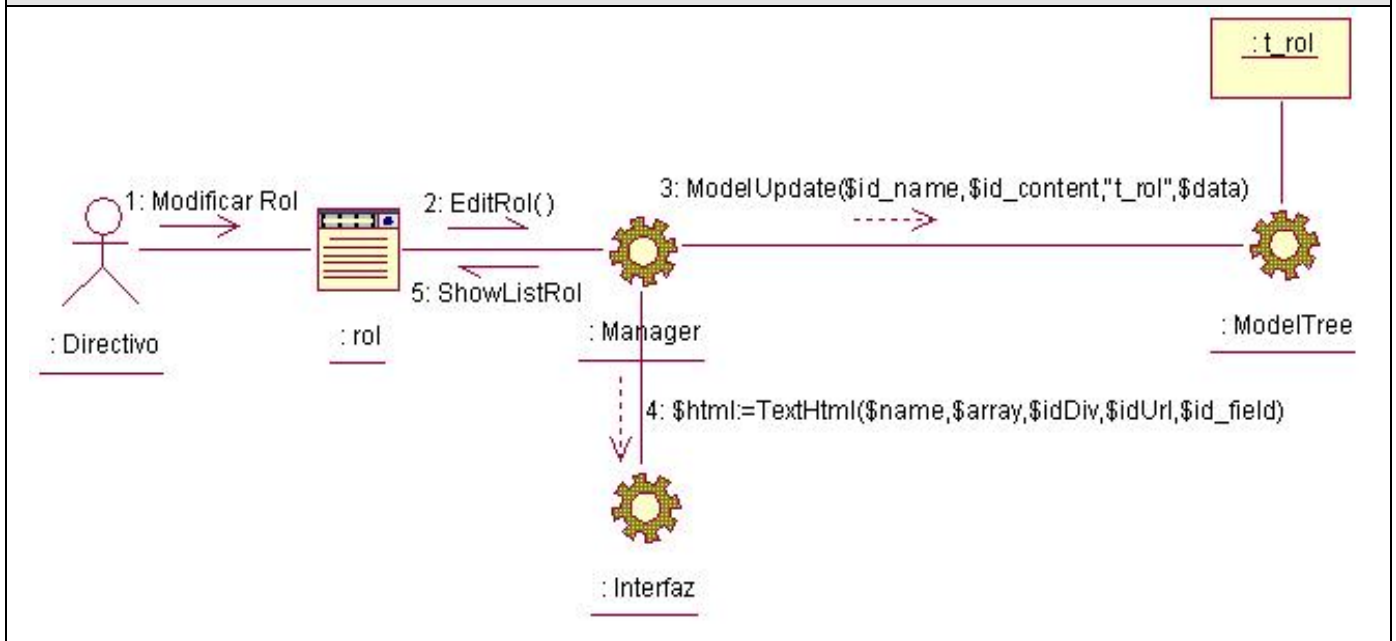
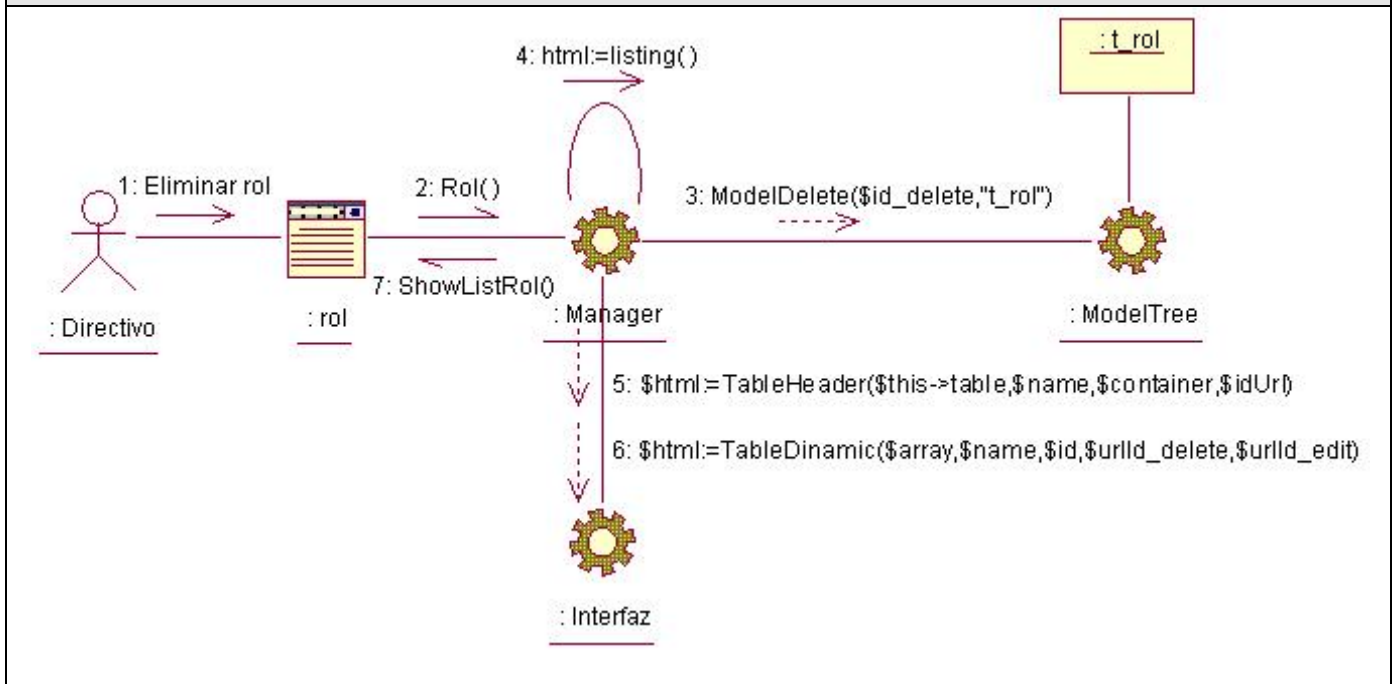


DIAGRAMA DE INTERACCION: Eliminar rol



Anexo 6. Descripción de las clases

Nombre:	Interfaz		
Tipo de clase:	Controladora		
Atributo:		Tipo:	
Para cada responsabilidad:			
Nombre:	TableDinamic		
Descripción:	Devuelve el HTML de una tabla creada con datos, dinámicamente quien va hacer la tabla cuerpo.		
Nombre:	TableHeader		
Descripción:	Devuelve el HTML de una tabla creada con datos, dinámicamente quien va hacer la tabla cabecera.		
Nombre:	TableDinamicResponsable		
Descripción:	Devuelve el HTML de una tabla creada con datos, dinámicamente quien va hacer la tabla cuerpo de los responsables.		
Nombre:	TableHeaderResponsable		
Descripción:	Devuelve el HTML de una tabla creada con datos, dinámicamente quien va hacer la tabla cabecera de los responsables.		
Nombre:	HtmlResponsable		
Descripción:	Genera el formulario para editar el responsable.		
Nombre:	FrmListObjectives		
Descripción:	Devuelve un formulario para editar objetivos.		
Nombre:	TextHtml		
Descripción:	Genera un formulario para editar un rol o una categoría.		

Nombre:	Manager		
Tipo de clase:	Controladora		
Atributo:	data	Tipo:	array
Para cada responsabilidad:			
Nombre:	Responsable		
Descripción:	Crea y elimina responsables		
Nombre:	Rol		
Descripción:	Crea y elimina roles		
Nombre:	Category		
Descripción:	Crea y elimina categorías		
Nombre:	EditResponsable		
Descripción:	Modifica responsables		
Nombre:	Listing		
Descripción:	Crea la lista de roles y categorías		
Nombre:	EditRol		
Descripción:	Modifica roles		
Nombre:	EditCategory		
Descripción:	Modifica categorías		
Nombre:	CreateObjectives		
Descripción:	Crea objetivos.		
Nombre:	EditListObjectives		
Descripción:	Modifica un objetivo.		
Nombre:	ListObjectives		
Descripción:	Muestra el listado de los objetivos.		

Nombre:	ModelTree		
Tipo de clase:	Controladora		
Atributo:	db	Tipo:	array
Para cada responsabilidad:			
Nombre:	ModelUpdate		
Descripción:	Se encarga de actualizar los cambios que se realicen.		
Nombre:	ModelDelete		
Descripción:	Elimina duplas en una tabla especificada según un identificador pasado como parámetro.		
Nombre:	ModelInsert		
Descripción:	Se encarga de insertar duplas a una tabla especificada.		

Anexo 7. Descripción de las tablas de la Base de datos

Nombre:	5	
Descripción:	Almacena los datos referentes a los datos de las tablas t_objectives y t_indicator.	
Atributo	Tipo	Descripción
id_indicator	Integer	Llave primaria foránea de la tabla t_indicator.
id_objectives	Integer	Llave primaria foránea de la tabla t_objectives.

Nombre:	t_category	
Descripción:	Almacena los datos referentes a las categorías, que pueden ser un área de trabajo, departamentos, entre otros.	
Atributo	Tipo	Descripción
id_category	Integer	Llave primaria de la tabla
category	Char(60)	Nombre de la categoría a la que pertenece un responsable.

Nombre:	t_indicator	
Descripción:	Almacena los datos referentes a los indicadores.	
Atributo	Tipo	Descripción
id_indicator	Integer	Llave primaria de la tabla
indicator	Char(60)	Fórmula del indicador.
description	Char(60)	Descripción del indicador.

Nombre:	t_rol	
Descripción:	Almacena los datos referentes a los roles que puede desempeñar un responsable.	
Atributo	Tipo	Descripción
id_rol	Integer	Llave primaria de la tabla
rol	Char(60)	Nombre del rol que desempeña un responsable.

Nombre:	t_objectives	
Descripción:	Almacena los datos referentes a los objetivos de la empresa.	
Atributo	Tipo	Descripción
id_objectives	Integer	Llave primaria de la tabla
name	Char(60)	Nombre del objetivo.
objetivo	Char(60)	Descripción del objetivo.
negocio	Integer	Nombre del tipo de negocio a que se dedica la empresa.
perspectiva	Integer	Nombre de la perspectiva.
estado	Integer	Estado del objetivo.
fecha_estimada	Datetime	Fecha estimada de la realización del objetivo.
strategy	Integer	Estrategia del objetivo.
plan	Integer	Plan de acción del objetivo.

Nombre:	t_strategy	
Descripción:	Almacena los datos referentes a las estrategias.	
Atributo	Tipo	Descripción
id_strategy	Integer	Llave primaria de la tabla
strategy	Char(60)	Descripción de la estrategia.
name	Char(60)	Nombre de la estrategia
id_objectives	Integer	Llave primaria foránea de la tabla t_objectives.

Nombre:	t_plan	
Descripción:	Almacena los datos referentes a los indicadores.	
Atributo	Tipo	Descripción
id_plan	Integer	Llave primaria de la tabla
plan de acción	Char(60)	Descripción del plan de acción.
id_objectives	Integer	Llave primaria foránea de la tabla t_objectives.

Nombre:	t_responsable	
Descripción:	Almacena los datos referentes a los responsables de las actividades.	
Atributo	Tipo	Descripción
id_responsable	Integer	Llave primaria de la tabla
responsable	Char(60)	Nombre del responsable.
correo	Char(60)	Dirección electrónica del correo.
rol	Integer	Nombre del rol del responsable
categoría	Integer	Nombre de la categoría a la que pertenece el responsable.
id_rol	Integer	Llave foránea de la tabla t_rol.
id_objectives	integer	Llave primaria foránea de la tabla t_objectives.
id_category	Integer	Llave foránea de la tabla t_category.

Nombre:	t_business	
Descripción:	Almacena los datos referentes a los negocios.	
Atributo	Tipo	Descripción
id_business	Integer	Llave primaria de la tabla
name	Char(60)	Nombre del negocio.
id_organization	Integer	Nombre de la organización.
id_objectives	Integer	Llave primaria foránea de la tabla t_objectives.

Nombre:	t_perspective	
Descripción:	Almacena los datos referentes a las perspectivas.	
Atributo	Tipo	Descripción
id_perspective	Integer	Llave primaria de la tabla
name	Char(60)	Nombre de la perspectiva.
id_business	Integer	Nombre del negocio.
id_objectives	Integer	Llave primaria foránea de la tabla t_objectives.

GLOSARIO DE TÉRMINOS

- **Acrónimo recurrente:** Un acrónimo en el que la primera letra del mismo hace referencia al propio acrónimo.
- **Applet:** Componente de una aplicación que corre en el contexto de otro programa. No puede correr de manera independiente, ofrece información gráfica y a veces interactúa con el usuario, típicamente carece de sesión y tiene privilegios de seguridad restringidos. Normalmente lleva a cabo una función muy específica que carece de uso independiente.
- **Banner:** Formato publicitario en Internet. Esta forma de publicidad online consiste en incluir una pieza publicitaria dentro de una página web. En la totalidad de los casos, su objetivo es atraer tráfico hacia el sitio web del anunciante que paga por su inclusión.
- **Base de datos:** Conjunto de datos que pertenecen al mismo contexto almacenados sistemáticamente para su posterior uso.
- **Cliente ligero:** Una computadora (cliente) en una arquitectura de red cliente-servidor que tiene muy poca o ninguna lógica del programa, por lo tanto depende principalmente del servidor central para las tareas de procesamiento. La palabra ligero se refiere a lo pequeña que es la imagen de arranque, quizá no más grande que la requerida para conectar a la red y arrancar un navegador web.
- **Código abierto:** Término con el que se conoce al software distribuido y desarrollado libremente.
- **Drupal:** Sistema de administración de contenido para sitios Web. Permite publicar artículos, imágenes, u otros archivos y servicios añadidos como foros, encuestas, votaciones, blogs y administración de usuarios y permisos.
- **Fichero:** Conjunto de información que se almacena en una computadora y puede ser identificado por su ruta completa.
- **Framework:** Estructura de soporte definida en la cual otro proyecto de software puede ser organizado y desarrollado. Puede incluir soporte de programas, bibliotecas y un lenguaje de scripting entre otros software para ayudar a desarrollar y unir los diferentes componentes de un proyecto. Representa una arquitectura de software que modela las relaciones generales de las entidades del dominio. Provee una estructura y una metodología de trabajo la cual extiende o utiliza las aplicaciones del dominio.
- **Hardware:** Conjunto de elementos materiales que componen una computadora.

- **Hilo:** Secuencia simple de instrucciones ejecutada en paralelo con otras secuencias. Permiten dividir un programa en dos o más tareas que corren simultáneamente, por medio de la multiprogramación. Permite incrementar el rendimiento de un procesador de manera considerable. Son utilizados para simplificar la estructura de un programa que lleva a cabo diferentes funciones. Todos los hilos de un proceso comparten los recursos del proceso. Residen en el mismo espacio de direcciones y tienen acceso a los mismos datos.
- **Holístico:** (De holismo) Idea de que todas las propiedades de un sistema no pueden ser determinadas o explicadas como la suma de sus componentes. El sistema completo se comporta de un modo distinto que la suma de sus partes. Se puede definir como un tratamiento de un tema que implica a todos sus componentes, con sus relaciones obvias e invisibles.
- **Host:** Máquina conectada a una red de ordenadores y que tiene un nombre de equipo. Es un nombre único que se le da a un dispositivo conectado a una red informática.
- **Iframe:** (*inline frame* o *marco incorporado*) es un elemento HTML que permite insertar o incrustar un documento HTML dentro de un documento HTML principal.
- **Lenguaje interpretado:** Aquellos lenguajes cuyos programas son habitualmente ejecutados en un intérprete en vez de compilados.
- **Licenciamiento dual:** Práctica de conceder dos o más licencias para el mismo producto intelectual.
- **Logo:** Dibujo que le sirve a una entidad o un grupo de personas para representarse. Suelen encerrar indicios y símbolos acerca de quienes representan.
- **Multihilo:** Ver *hilo*.
- **Multimedia:** Sistema que utiliza más un medio de comunicación al mismo tiempo en la presentación de la información, como el texto, la imagen, la animación, el vídeo y el sonido.
- **Navegador web:** Aplicación software que permite al usuario recuperar y visualizar documentos de hipertexto, comúnmente descritos en HTML, desde servidores web de todo el mundo a través de Internet. Esta red de documentos es denominada World Wide Web (WWW). Permite mostrar o ejecutar: gráficos, secuencias de vídeo, sonido, animaciones y programas diversos además del texto y los hipervínculos o enlaces. Permite la visualización de documentos de texto, posiblemente con recursos multimedia incrustados.

- **Parche:** Pequeña porción de código que se introduce en un programa, con la intención de subsanar un error conocido y acotado del mismo.
- **Plugin:** Aplicación informática que interactúa con otra aplicación para aportarle una función o utilidad específica. Se utilizan como una forma de expandir programas de forma modular, de manera que se puedan añadir nuevas funcionalidades sin afectar a las ya existentes ni complicar el desarrollo del programa principal.
- **Recursión:** Es la forma en la cual se especifica un proceso basado en su propia definición. Siendo un poco más precisos, y para evitar el aparente círculo sin fin en esta definición, las instancias complejas de un proceso se definen en términos de instancias más simples, estando las finales más simples definidas de forma explícita.
- **Script:** Guión o conjunto de instrucciones que permiten la automatización de tareas creando pequeñas utilidades.
- **Servidor:** Una aplicación informática o programa que realiza algunas tareas en beneficio de otras aplicaciones llamadas clientes. Algunos servicios habituales son los servicios de archivos, que permiten a los usuarios almacenar y acceder a los archivos de un ordenador y los servicios de aplicaciones, que realizan tareas en beneficio directo del usuario final.
- **Servidor web:** Programa que implementa el protocolo HTTP (hypertext transfer protocol). Este protocolo está diseñado para transferir lo que llamamos hipertextos, páginas web o páginas HTML (hypertext markup language): textos complejos con enlaces, figuras, formularios, botones y objetos incrustados como animaciones o reproductores de sonidos.
- **Sistema Gestor de Base de Datos:** Tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan. Se compone de un lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta.
- **Webmail:** Programa informático, concretamente un cliente de correo electrónico, que provee una interfaz web por la que acceder al correo electrónico.
- **Wiki:** Sitio web colaborativo que puede ser editado por varios usuarios. También se puede referir a una colección de páginas hipertexto, que pueden ser visitadas y editadas por cualquier persona.