

Universidad de las Ciencias Informáticas

Facultad 9



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

”Sistema de control de redes.”

**AUTORES: Manuel Alejandro Piña Tamayo.
Yusel Marrero Acanda.**

TUTOR: 1Tte Ing. Sergio Michel Rivera

Ciudad de La Habana, 2010.

“Año 52 de la Revolución”



“...aquí está una de las tareas de la juventud: empujar, dirigir con el ejemplo la producción del hombre de mañana. Y en esta producción, en esta dirección, está comprendida la producción de sí mismos...”

Ernesto Che Guevara.

Agradecimiento

De Yusel.

Quiero agradecer:

A mis padres por el apoyo y la confianza que me han dado en el transcurso de estos 5 años, a mi hermano por darme ánimo en todo momento.

A mi novia Belinda por estar siempre a mi lado en los buenos y malos momentos que pasamos juntos.

A mi familiares, mis abuelos que siempre se han preocupado por mi en todo momento.

Al tutor por ayudarnos en todos los momentos que lo fuimos a ver.

A todos mis amigos que están presentes y los que no están de mi antiguo grupo por los consejos que me dieron.

A todos aquellos que de una forma u otra contribuyeron a la realización de esta tesis

A todos muchas gracias.

De Manuel Alejandro.

Agradezco A:

Mis padres por estar presentes en todo momento y por el apoyo y la confianza me han dado en todas las decisiones que he tomado siempre.

Mis abuelos que siempre se han preocupado por mi y los llevo presentes en todo momentos... gracias por sus consejos.

Mis tíos que me tienen como un hijo más y a mis primos que siempre me han brindado su apoyo.

Katia por estar presente cuando más la necesitaba...

Al tutor por ayudarnos en todos los momentos que lo fuimos a ver, con sus mejores consejos y conocimientos.

Agradecer en especial a Samada, por su ayuda incondicional.

A todos mis amigos aquí presentes y a los que no han podido estar por un motivo u otro,

...gracias por su apoyo, su preocupación y su amistad sincera e incondicional.

A todos aquellos que de una forma u otra contribuyeron a la realización de esta tesis...gracias de corazón.

A la Revolución.

A Fidel, A Raúl.

A todos muchas gracias.

Debido a la necesidad que presenta las Fuerzas Armadas Revolucionarias (FAR) de un sistema que les ayude agilizar el trabajo de manera más rápida y segura sobre el control de la red y los servidores, surge la idea de este tema. La propuesta es realizar un sistema que le permita al administrador llevar el control de la red y de los servidores de una forma más eficiente.

Para dar cumplimiento a la solución de la problemática se realizó un proceso investigativo acerca de las tecnologías actuales. Para realizar el diseño se utilizó UML (Unified Modeling Language) como lenguaje de modelado y Visual Paradigm como herramienta CASE. Entre las principales herramientas utilizadas en el desarrollo de la aplicación se encuentra Sauxe 1.5.4 como marco de trabajo, usando PHP como lenguaje de programación por parte del servidor y java script con las hojas de estilo CSS por la parte del cliente. Como Sistema Gestor de Base de Datos se usó PostgreSQL y Servidor web Apache. Además se realizó la modelación del negocio en el cual se identificaron los principales procesos, identificándose los requisitos funcionales y realizándose en el diseño los distintos diagramas. También se utilizó el patrón arquitectónico Modelo Vista Controlador para el diseño de la aplicación. Finalmente se diseña la base de datos a través del diagrama de Entidad-Relación.

PALABRAS CLAVES

Red informáticas, Control, Inventario, Sistemas.

Índice

AGRADECIMIENTO..... I

ÍNDICE IV

INTRODUCCIÓN 2

CAPÍTULO I: FUNDAMENTACIÓN TEÓRICA..... 5

INTRODUCCIÓN 5

1.1. CONCEPTOS RELACIONADOS CON EL PROBLEMA.5

1.2. ANÁLISIS DE SOLUCIONES EXISTENTES EN EL MUNDO ACTUAL.6

 1.2.1. *Sistemas de gestión y control*.....6

 1.2.1.1. Nagios.....6

 1.2.1.2. Pandora FMS7

 1.2.2. *Sistemas de control mediante inventario*.....8

 1.2.2.1. Sistema de Inventarios de la Red Educativa Metropolitana9

 1.2.2.2. Administración de equipos en red VEO.....9

1.3. METODOLOGÍAS DE DESARROLLO DE SOFTWARE11

 1.3.1. *Proceso de Desarrollo y Gestión de Proyectos de Software*11

1.4. HERRAMIENTAS CASE DE MODELADO CON UML.....12

 1.4.1. *Características de Visual Paradigm*.....14

 1.4.2. *Ventajas de Visual Paradigm:*14

1.5. LENGUAJES DE PROGRAMACIÓN.....14

 1.5.1. *Ventajas de PHP:*.....15

 1.5.2. *Desventajas de PHP:*16

 1.5.3. *Ventajas de HTML:*.....16

 1.5.4. *Desventajas de HTML:*16

 1.5.5. *Ventajas de JavaScript:*17

 1.5.6. *Desventajas de JavaScript:*.....17

1.6. SISTEMAS GESTORES DE BASES DE DATOS17

 1.6.1. *Ventajas de PostgreSQL:*.....18

 1.6.2. *Características de PostgreSQL:*18

1.7. ENTORNO DE DESARROLLO INTEGRADO (IDE).....18

 1.7.1. *Ventajas de Zend Estudio:*.....19

 1.7.2. *Desventajas de Zend Estudio:*19

1.8. FRAMEWORKS.....19

1.9. SERVIDOR WEB21

 1.9.1. *Características de Apache:*.....21

1.10. CONCLUSIONES PARCIALES22

CAPITULO 2: CARACTERISTICAS DEL SISTEMA..... 23

INTRODUCCIÓN 23

2.1. PROCESOS DE NEGOCIO.....23

 2.1.1. *Mapa de procesos del negocio*.....23

2.2.	MODELO CONCEPTUAL	25
2.2.1.	<i>Conceptos del modelo conceptual:</i>	25
2.3.	REQUERIMIENTOS FUNCIONALES	27
2.4.	REQUERIMIENTOS NO FUNCIONALES.....	28
2.5.	CONCLUSIONES PARCIALES	29
CAPITULO 3: DISEÑO DEL SISTEMA		30
INTRODUCCIÓN		30
3.1.	DIAGRAMAS DE INTERACCIÓN	30
3.2.	DISEÑO.....	33
3.2.1.	<i>Modelo de diseño</i>	33
3.2.2.	<i>Patrón Modelo-Vista-Controlador (MVC)</i>	38
3.2.3.	<i>Patrones de Diseño</i>	40
3.3.	DISEÑO DE LA BASE DE DATOS	41
3.3.1.	<i>Modelo físico de datos. Diagrama Entidad-Relación.</i>	41
3.4.	CONCLUSIONES PARCIALES	42
CAPITULO 4: IMPLEMENTACIÓN Y PRUEBA		43
INTRODUCCIÓN		43
4.1.	MODELO DE DESPLIEGUE	43
4.2.	DIAGRAMA DE COMPONENTE	44
4.3.	MATRIZ DE INTEGRACIÓN DE COMPONENTES.....	45
4.4.	ESTÁNDARES DE CODIFICACIÓN.....	45
4.4.1.	<i>Nomenclatura por tipo de clase</i>	46
4.4.2.	<i>Nomenclatura de comentario en clases</i>	46
4.5.	PRUEBA	48
4.6.	CONCLUSIONES PARCIALES	57
CONCLUSIONES GENERALES		58
RECOMEDACIONES		59
REFERENCIA BIBLIOGRÁFICA.....		60
BIBLIOGRAFÍA.....		62
ANEXOS		64
ANEXO1: MODELO DEL NEGOCIO		64
	<i>Descripción de los procesos de negocio</i>	64
ANEXO2: REQUISITOS FUNCIONALES		65
	<i>Especificación del requisito Gestionar Configuración del servidor</i>	65
ANEXO 4: EJEMPLO DE DESCRIPCIÓN DE LAS CLASES.....		67
GLOSARIO DE TÉRMINOS.....		71

INTRODUCCIÓN

En la actualidad es evidente el auge alcanzado en el uso de las Tecnologías de la Informática y las Comunicaciones como apoyo a los centros laborales, al auto aprendizaje y como medio de comunicación entre las personas. Las aplicaciones informáticas que gestionan información son cada vez más utilizadas por las instituciones, debido a que permiten sustituir los procedimientos tradicionales de manipulación y control de la información, que se realizan manualmente, por métodos automatizados de almacenamiento que proveen un ambiente de trabajo para la toma de decisiones y el manejo de resultados. La informática ha tomado un gran auge a nivel mundial, pero a pesar del gran desarrollo que tiene, existe un gran problema en las redes de computadoras. Estas son asediadas constantemente por personas que se dedican a buscar información para usarla ya sea para mal o para bien. El interés y conocimiento de la seguridad en la red crece a la par de las mismas. El control de la misma es un tema que abarca mucho porque se tiene que tener en cuenta un grupo de parámetros, para que se tenga bien controlada. El control de la información de la red es un recurso que se utiliza para mantener los datos ya sea de un ordenador o servidor y así esta se almacena en una base de datos para después poder actualizar o modificar algún dato que se haga. Hoy día las redes se caracterizan por un constante incremento del número, complejidad y heterogeneidad de los recursos.

El mundo se mueve rápidamente en un ambiente de tecnologías de información. Las computadoras y con ellas las redes de computadoras han salido de los ambientes universitarios y científicos para incorporarse a todos los sectores de la vida, es común ver una red en una escuela, una fabrica, un edificio de oficinas o un hotel; nada ni nadie puede mantenerse al margen. Con una simple mirada al desarrollo de las tecnologías de la información como convergencia entre la Informática, las Telecomunicaciones, la Electrónica y la Automatización se puede asegurar que la introducción de estas técnicas en todas las áreas de la vida económica y social del país es un imperativo del avance en las condiciones actuales y el único medio eficaz para sustentar nuestro desarrollo y potenciar los logros alcanzados, haciendo posible convertir en realidad los sueños de una cultura general integral masiva. Consciente de la importancia que tienen las tecnologías de la información, nuestro gobierno ha definido que el amplio uso de todas las posibilidades de las tecnologías y servicios de información y las telecomunicaciones, deben constituir prioridad del país a los fines de garantizar la mayor eficiencia en la gestión y los procesos productivos. El

país debe encaminarse resueltamente a la modernización informática mediante un programa integral que involucre a las organizaciones que deben proveer los recursos materiales, financieros e intelectuales y a las entidades económicas, políticas y sociales que deben traducirlos en más y mejores productos y servicios. Para apoyar estos objetivos se deberá desarrollar un amplio movimiento de calificación, desde la formación de estudiantes hasta la recalificación de cuadros de dirección empresariales y estatales y demás trabajadores en todas las instancias. (1)

Así como en el mundo entero el avance de la tecnología tiene gran importancia, en nuestro país, trabajadores de las Fuerzas Armadas Revolucionarias (FAR) son partícipes de las ventajas que ofrece la navegación en la red. Para las entidades de las FAR es necesario tener la información oculta, por lo que se debe tener un control riguroso de cada una de las máquinas que se utilizan para el trabajo.

A pesar de que en estos momentos en las FAR no se cuenta con una herramienta de software para llevar el control de la red. Ya comienza a cobrar fuerza por parte de un grupo ya que actualmente en la FAR se lleva cabo esta función de forma manual, por lo que se hace más difícil llevar todos los datos debido al gran volumen de información. El mismo se hace por paso teniendo en cuenta una serie de información como es la configuración de los servidores básico que cuenta con distintos atributos que se debe llenar, también se debe llevar el tipo de política con la que cuenta los servidores, otros de los datos que se lleva es el control de los discos duros provocando que el trabajo sea más lento, tedioso y poco eficiente. Para poder manejar toda esta información se necesita una base de datos con el fin de lograr almacenarla y tenerla más organizada en un tiempo breve y de alta confiabilidad.

Por lo expresado anteriormente se plantea como **problema a resolver**: ¿Cómo gestionar de forma eficiente y sencilla la información requerida sobre el control a redes y servidores que se disponen en el manual de seguridad informática de las FAR?

Siendo identificado como **objeto de estudio**: La gestión de la seguridad informática en centros de datos.

Dentro de dicho objeto de estudio se enmarca el siguiente **campo de acción**: El proceso de control y gestión de las redes y servidores en la FAR.

Teniendo en cuenta lo antes planteado en el problema científico se propone como **objetivo general** de este trabajo: Desarrollar un sistema que controle la información referente a las redes y los servidores.

Como **objetivos específicos** se plantean los siguientes:

- Realizar el marco teórico de la investigación.
- Modelar los procesos del sistema.

- Implementar el sistema de control de redes.
- Validar la solución desarrollada.

Idea a defender

Con el desarrollo del sistema de control de las redes en la FAR se obtendrá un mecanismo de control de la información más eficiente y seguro, y a su vez se hará más ágil la actualización de la información.

Para poder guiar el estudio de la investigación se hizo necesario el uso de diferentes Métodos Científicos. Como método Empírico se utiliza la Entrevista con el objetivo de adquirir nuevo conocimiento sobre el tema que estamos investigando, también a través de él pudimos sacar los requisitos que el sistema debe tener para la solución de dicha investigación y los elementos para la modelación del negocio. Dentro de los métodos Teóricos se tuvieron en cuenta el Histórico-Lógico para así analizar las tecnologías, herramientas que se pueden utilizar para el sistema y por último se utiliza la Modelación para realizar ya sea el modelo de análisis y diseño y poder modelar la base de datos del sistema.

El presente documento está estructurado de la siguiente forma:

Capítulo 1 “Fundamentación teórica”: En este capítulo se exponen los principales conceptos asociados al dominio del problema. Además de los distintos sistemas existentes en Cuba y el mundo, se explican las metodologías, los lenguajes de programación y las herramientas utilizadas para el desarrollo de la aplicación.

Capítulo 2 “Características del sistema”: Se hace un estudio de los procesos de negocio. Se definen los requisitos funcionales y no funcionales, además de una breve descripción de los requisitos funcionales y se realiza la modelación del sistema propuesto.

Capítulo 3 “Diseño del sistema”: Se especifican los principios de diseño adoptados. Incluye los diagramas de clases del diseño, se explican los patrones de diseños que se utilizan así como el modelo de datos para el diseño de la base de datos.

Capítulo 4 “Implementación y prueba”: Se realiza el diagrama de componente y de despliegue. Se define el tipo de prueba que se utilizará, además se describen los casos de pruebas por requisitos.

CAPÍTULO I: FUNDAMENTACIÓN TEÓRICA

Introducción

Actualmente en el mundo de la informática podemos encontrar sistemas relacionados con el control de la red ya que en casi todas las entidades es de gran importancia contar con la posibilidad de tener un software o sistema de este tipo. Hoy día la FAR no cuenta con un sistema de esta índole destinado a llevar un control estricto de la información. En este capítulo abordaremos una serie de conceptos relacionados con nuestra investigación, también analizaremos otras soluciones existentes que abarquen este tema, en el se hará un estudio de las principales herramientas que se usan en el modelo de los diferentes sistemas informáticos.

1.1. Conceptos relacionados con el problema.

Para lograr un mejor entendimiento de las tecnologías, herramientas y estudios planteados en el trabajo, es preciso contar con el dominio y conocimiento de términos abordados en el mismo; para esto se plantea una serie de conceptos partiendo desde los sistemas, los cuales no son más que: “un todo integrado, aunque compuesto de estructuras diversas, interactuantes y especializadas. Cualquier sistema tiene un número de objetivos, y los pesos asignados a cada uno de ellos pueden variar ampliamente de un sistema a otro; un sistema ejecuta una función imposible de realizar por cualquiera de las partes individuales. La complejidad de la combinación está implícita; es una colección organizada de hombres, máquinas y métodos necesaria para cumplir un objetivo específico”.

Los sistemas son clasificados de diversas formas y tipos, entre los más comunes encontramos a los **sistemas de control** los cuales son definidos como: “un tipo de sistema que se caracteriza por la presencia de una serie de elementos que permiten influir en el funcionamiento del sistema. La finalidad de un sistema de control es conseguir, mediante la manipulación de las variables de control, un dominio sobre las variables de salida, de modo que estas alcancen valores prefijados. (2)

En el campo de la informática los sistemas de control son utilizados generalmente para el control de las **redes informáticas**, las cuales no son más que: “un conjunto de computadoras conectadas entre sí compartiendo información, recursos como CD-ROM, impresoras, grabadoras de DVD y servicios como e-mail, Chat, conexiones a Internet, juegos. (3)

En muchas ocasiones estas funcionalidades de control están apoyadas por los sistemas de inventariado los cuales permiten mantener un control exacto y eficiente de los datos; ofreciendo diversas estadísticas que permiten una mejor toma de decisiones y la detección de cambios. Además estos sistemas ofrecen información exacta y precisa en cualquier momento: información en tiempo real; mejoras en el control ofreciendo herramientas funcionales y fáciles de utilizar, agilización en el proceso de control: evitando el traslado de grandes volúmenes de documentación para realizar dicho proceso, eficiencia, reduciendo los errores durante el proceso, ya sea por datos erróneos o datos perdidos; y comodidad, ya que toda la información estará almacenada, a la espera de oprimir un solo botón.

Los sistemas de control y gestión de redes son herramientas robustas y eficaces, las cuales permiten la monitorización de la red, estos cuentan con una serie de servicios y plugins los cuales le permiten realizar y cumplir sus objetivos con eficacia; cada servicio o plugins en un sistema de control y gestión de redes se basa en la obtención, manejo y control de datos de un servicio específico, con el cual cuentan los ordenadores y la red que monitorea dicho sistema.

1.2. Análisis de soluciones existentes en el mundo actual.

En la actualidad existen herramientas destinadas a llevar el control de la red en diferentes empresas. Las cuales cuentan con un sistema capaz de solucionar problemas complejos, pero no todas lo hacen a un 100% ya que existen algunas deficiencias en cuantos varios factores.

1.2.1. Sistemas de gestión y control.

1.2.1.1. Nagios

Nagios es una aplicación para monitoreo de sistemas y redes. Revisa equipos y servicios que uno mismo le especifica, cuando algo está mal y cuando está bien. Se trata de un software que proporciona una gran versatilidad para consultar prácticamente cualquier parámetro de interés de un sistema, y genera alertas, que pueden ser recibidas por los responsables correspondientes mediante (entre otros medios) correo electrónico y mensajes SMS, cuando estos parámetros exceden de los márgenes definidos por el administrador de red.

Características

Sistema de monitorización de las aplicaciones y servicios proporcionados por el SI, que informa automática e instantáneamente 24 horas al día, 365 días al año, en caso de que se produzca un fallo en ellos.

- Software Libre. S.O. Linux.
- Altamente configurable mediante plugins.
- Disponible interfaz web para consulta del estado de los recursos y servidores
- Sistema de notificaciones muy flexible.
- Permiten resolver el problema más rápidamente y en ocasiones de forma automática mediante “handlers”o pequeños programas que se ejecutan cuando ocurre un fallo en el sistema.
- Tiene en cuenta las dependencias entre servicios.
- Se producen en función de la hora y/o día.
- Se elige el método en función del estado del servicio. (4)

1.2.1.2. Pandora FMS

Pandora FMS es un software de Código Abierto que sirve para monitorizar y medir todo tipo de elementos. Monitoriza sistemas, aplicaciones o dispositivos. Permite saber el estado de cada elemento de un sistema a lo largo del tiempo.

Puede recoger información de cualquier sistema operativo, con agentes, específicos para cada plataforma, que recolectan datos y los envían al servidor. Hay agentes específicos para GNU/Linux, AIX, SUN Solaris, HP-UX, BSD/IPS0 y Windows 2000, XP y 2003. También puede monitorizar cualquier tipo de servicio TCP/IP, sin necesidad de instalar agentes, y monitorizar sistemas de red como balanceadores de carga, routers, switches, sistemas operativos, aplicaciones o impresoras si se necesita hacerlo de forma remota.

Algunos ejemplos de recursos comunes que pueden ser monitorizados con Pandora FMS son, la carga del procesador, el uso de disco y memoria, procesos que están corriendo en el sistema, eventos determinados en un log, factores ambientales como la temperatura, la luz o la humedad, valores de aplicaciones como determinados textos en una página web, y en general cualquier cosa que se pueda recolectar de forma automatizada. (5)

Aunque Pandora FMS es conocido como un sistema con una alta capacidad de control a los servidores mediante la instalación de agentes; no es la solución al problema debido a que aunque monitorea un sin número de parámetros dentro de los que no se encuentran los requeridos para dar solución al problema planteado.

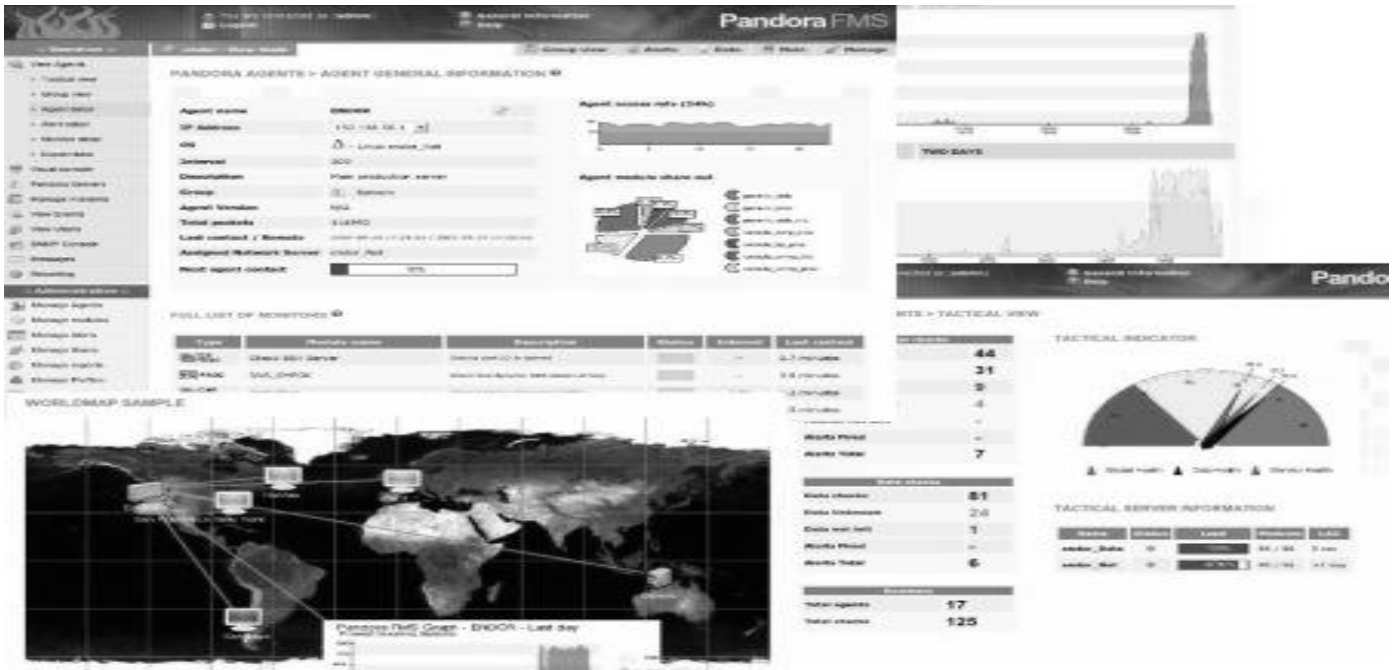


Fig.1 Mapa de Pandora FMS.

1.2.2. Sistemas de control mediante inventario.

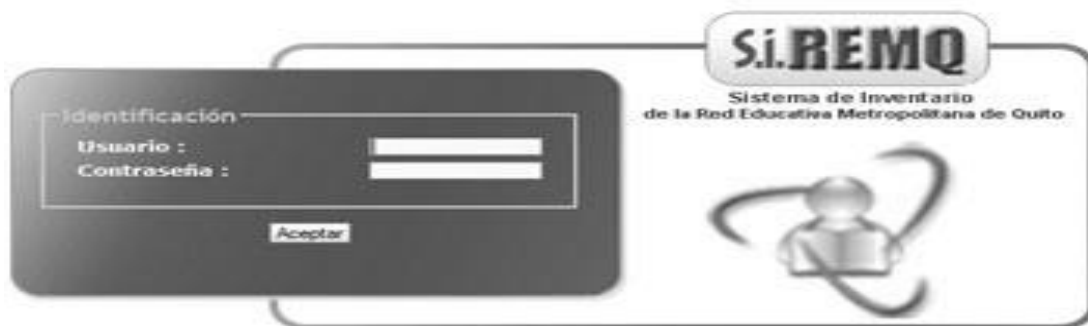
Los sistemas antes planteados, como son Nagios y Pandora FMS realizan un proceso de gestión y control con una gran exactitud y un manejo bastante abarcador de diversos parámetros; sin embargo aunque permiten llevar un control mediante estadísticas, las mismas no reportan un análisis completo de los datos de una red. Los sistemas que a continuación se muestran plantean una solución lo más semejante posible al sistema que se desea desarrollar ya que permiten el control de la redes mediante el inventariado de las mismas, ofreciendo un completo sistema de estadísticas que permiten el control de las redes.

1.2.2.1. Sistema de Inventarios de la Red Educativa Metropolitana

El Sistema de Inventarios de la Red Educativa Metropolitana, muestra consideraciones generales importantes como el de listar y consultar el inventario de hardware y software de las instituciones educativas beneficiarias de la REMQ. Tiene la posibilidad de visualizar el inventario de las instituciones beneficiarias, en el cual podemos efectuar búsquedas parametrizadas y ordenadas, sobre uno de los tipos de inventario, tales como:

- Computadores.
- Material de red.
- Impresoras.
- Monitores.
- Periféricos externos.
- Software.
- Reparaciones.

Los administradores del sistema pueden modificar o eliminar cada elemento, en cada ficha de información, hay pestañas disponibles para obtener una visión más legible de los diversos datos. (6)



1.2.2.2. Administración de equipos en red VEO

El software **VEO** es una herramienta con un rico conjunto de características y funcionalidades integradas en una Consola Central para la Administración que le permite el acceso remoto a sus equipos en redes

WAN, LAN. Utiliza un cliente remoto que se puede instalar de manera transparente a los equipos remotos, trabaja en un segundo plano y no podrá ser removido por el usuario.



Permite tomar el control remoto total de cualquier equipo en la red en donde se tenga instalado el agente remoto de VEO. Genera un inventario completo y detallado del hardware de los equipos remotos, cualquier información que no sea detectada podrá ser ingresada manualmente dentro de una serie de campos disponibles. La información obtenida puede ser filtrada a Excel.

Los datos son guardados en tablas dentro de la base de datos para su consulta histórica e incluye la siguiente información:

- **General:** Nombre de la estación, usuario del sistema, IP address, MAC address, grupo de trabajo, dominio, servidor (dns), Host IP, fecha en la que se realizó el último inventario.
- **Sistema Operativo:** Versión de Windows 95/98/Me/NT/2000/XP/2003, Professional, Standard, serial.
- **Memoria:** Física, swap y virtual, total disponible. Detalle de sockets con tamaño de la tarjeta instalada y RAM máximo soportado de la estación.
- **Procesadores:** Fabricante, tipo, nombre, frecuencia, frecuencia real, serial.
- **Dispositivos lógicos:** tipo de dispositivo: CD-ROM o disco duro; unidades utilizadas, espacio libre (GB) capacidad: total, usada y disponible, sistema de archivos: FAT, FAT32, NTFS; número de serie y descripción del dispositivo.
- **Hardware (Drivers existentes):** Fabricante, tipo y descripción. Todo el hardware definido es reconocido automáticamente por medio de los drivers instalados. Reconoce: monitor, teclado, mouse, tarjetas de video, discos externos e internos, MODEM, unidades de diskette, controladores

de disco, impresora instalada predefinida, dispositivos de audio (media), tarjetas de red, puertos, toda la información se puede filtrar y exportar a Excel. (7)

Tanto el sistema de Inventarios de la Red Educativa Metropolitana como la poderosa herramienta (VEO) destinada a la administración de equipos en red no ofrecen una solución adecuada a la situación problemática planteada debido a que manejan un conjunto de parámetros referentes a la red educativa metropolitana y redes empresariales respectivamente no acordes a los planteados en el anexo 7 del manual de seguridad informática de las FAR.

Teniendo en cuenta los sistemas analizados anteriormente, se puede contactar que satisfacen gran parte de los requerimientos correspondientes a nuestro problema, con el inconveniente que algunos son software de empresas privadas y otros no controlan los parámetros requeridos, motivo por el cual no son la solución a nuestro problema.

1.3. Metodologías de desarrollo de software

1.3.1. Proceso de Desarrollo y Gestión de Proyectos de Software

Entre sus principales características en el ciclo de vida expuesto se encuentra que: Las fases son secuenciales y su transferencia debe ser precedida por un proceso de revisión o liberación del Centro de Calidad y su aprobación en Consejo Técnico Formal. El nivel del personal es bajo al comienzo, alcanza su nivel máximo en la fase de construcción y decae rápidamente cuando el proyecto se aproxima a su conclusión. La participación de los interesados es alta en las etapas de Inicio y Modelación, baja en la etapa de Construcción y vuelve a subir en las etapas finales del proyecto.

Las etapas o fases permiten mejorar el planeamiento, ejecución y control del proyecto, se caracterizan por la conclusión y la aprobación de uno o más artefactos entregables siendo estos productos del trabajo, mensurables y verificables, correspondiendo algunos a los procesos de gestión de proyectos y otros al producto. La consecución exitosa de cada fase es indispensable para poder continuar con el proyecto. El ciclo de vida de un proyecto de software desarrollado con dicha metodología se descompone en el tiempo en cinco fases secuenciales que son: Inicio, Modelación, Construcción, Explotación Experimental, Despliegue.

Al final de cada fase los representantes de los grupos de roles presentes en el proyecto realizan una evaluación para determinar si los objetivos se cumplieron y así presentar a Consejo Técnico Formal para su evaluación y dar paso o no a la fase siguiente.



Fig. 2 Fases de la metodología

1.4. Herramientas CASE de Modelado con UML

CASE es una sigla, que corresponde a las iniciales de: Computer Aided Software Engineering; y en su traducción al Español significa Ingeniería de Software Asistida por Computación.

El concepto de CASE es muy amplio; y una buena definición genérica, que pueda abarcar esa amplitud de conceptos, sería la de considerar a la Ingeniería de Software Asistida por Computación (CASE), como la aplicación de métodos y técnicas a través de las cuales se hacen útiles a las personas comprender las capacidades de las computadoras, por medio de programas, de procedimientos y su respectiva documentación.

Concentrando nuestra atención en el uso de estas herramientas, para el desarrollo de proyectos informáticos que tengan como objetivo la automatización de procedimientos administrativos; podemos decir que:

Las herramientas CASE representan una forma que permite Modelar los Procesos de Negocios de las empresas y desarrollar los Sistemas de Información Gerenciales. (8)

Proporcionan un conjunto de herramientas semiautomatizadas y automatizadas que brindan ayuda y dan asistencia a los analistas, ingenieros de software y desarrolladores, durante todos los pasos del ciclo de vida de desarrollo de un software.

Lenguaje Unificado de Modelado (UML)

Es un lenguaje para el desarrollo de software orientado a objetos, su propósito es visualizar, especificar, construir y documentar proyectos de software. (9)

Hoy día, UML está consolidado como el lenguaje estándar en el análisis y diseño de sistemas de cómputo. Mediante UML es posible establecer la serie de requerimientos y estructuras necesarias para plasmar un sistema de software previo al proceso intensivo de escribir código.

En otros términos, así como en la construcción de un edificio se realizan planos previo a su construcción, en Software se deben realizar diseños en UML previa codificación de un sistema, ahora bien, aunque UML es un lenguaje, éste posee más características visuales que programáticas, mismas que facilitan a integrantes de un equipo multidisciplinario participar e intercomunicarse fácilmente, estos integrantes siendo los analistas, diseñadores, especialistas de área y desde luego los programadores.

Es importante destacar que UML no es una guía para realizar el análisis y diseño orientado a objetos, es decir, no es un proceso, sino un lenguaje que permite la modelación de sistemas con tecnología orientada a objetos. La finalidad de los diagramas es presentar diversas perspectivas de un sistema, a las cuales se les conoce como modelo. Además describe lo que supuestamente hará un sistema, pero no dice cómo implementar dicho sistema.

Debido a que soporta aplicaciones del tipo web como las que vamos a desarrollar genera código para java y exportación como HTML, genera base de datos y es de licencia gratuita es escogido **Visual Paradigm** como herramienta CASE que utiliza “UML” para modelar, visualizar y documentar los procesos pertinentes a nuestra aplicación. Como lenguaje de modelaje profesional, Visual Paradigm soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue.”

1.4.1. Características de Visual Paradigm

- Producto de calidad.
- Soporta aplicaciones web.
- Las imágenes y reportes generados, no son de muy buena calidad.
- Varios idiomas.
- Generación de código para Java y exportación como HTML.
- Fácil de instalar y actualizar.
- Compatibilidad entre ediciones.
- Licencia Gratuita.

Cabe destacar igualmente su robustez, usabilidad y portabilidad. En definitiva, visual paradigm es una herramienta muy a tener en cuenta a la hora de desarrollar un proyecto importante.

1.4.2. Ventajas de Visual Paradigm:

- Es multiplataforma y muy útil para la generación de código fuente en PHP.
- Tiene la capacidad de crear el esquema de clases a partir de una base de datos y crear la definición de base de datos a partir del esquema de clases.
- Permite invertir código fuente de programas, archivos ejecutables y binarios en modelos UML al momento, creando de forma simple toda la documentación.
- Incorpora el soporte para trabajo en equipo, que permite que varios desarrolladores trabajen a la vez en el mismo diagrama y vean en tiempo real los cambios hechos por sus compañeros de equipo.

Está compuesto por una serie de elementos gráficos que le permite al ingeniero lograr realizar un diseño del software, al igual que otras herramientas de modelado como el Rational, pero en este caso en la solución expuesta se ha tratado de orientar la misma a un entorno libre y de estas dos herramientas la que cumple con las condiciones es el Visual Paradigm.

1.5. Lenguajes de programación

Se propone utilizar tecnología Web para resolver el problema argumentado, debido a que esta tecnología posee muchas ventajas como:

- Se necesita solamente un navegador web y conexión al sistema.

- Se pueden ejecutar varios clientes simultáneamente y en diferentes estaciones de trabajo.
- Se necesitan pocos recursos para que la aplicación trabaje correctamente.
- Toda la información se encontrará de forma centralizada en el servidor, al igual que la administración.
- Por lo general no depende de ningún software.
- No hay generalmente discriminación del sistema operativo por parte del usuario, aunque exista por parte del servidor.

El servidor puede estar sujeto a cambios que deben influir poco o en nada en el cliente, también puede estar en diferente plataforma a la del servidor que no afectará el funcionamiento del sistema. Algo importante es que el servidor puede brindar servicios múltiples y simultáneos a clientes en cualquier lugar. Para llevar a cabo la implementación del sistema es necesario saber cual lenguaje de programación usar. (10)

Como lenguaje de programación se propone por parte del servidor **Hypertext Pre-Processor (PHP)** ya que es: "un lenguaje de script interpretado en el lado del servidor utilizado para la generación de páginas web dinámicas, incrustadas en páginas HTML y ejecutadas en el servidor. PHP no necesita ser compilado para ejecutarse. Para su funcionamiento necesita tener instalado un servidor Apache o IIS con las librerías de PHP. La mayor parte de su sintaxis ha sido tomada de C, Java y Perl con algunas características específicas. Los archivos cuentan con la extensión (php)."

1.5.1. Ventajas de PHP:

- Muy fácil de aprender.
- Se caracteriza por ser un lenguaje muy rápido.
- Soporta en cierta medida la orientación a objeto. Clases y herencia.
- Es un lenguaje multiplataforma: Linux, Windows, entre otros.
- Capacidad de conexión con la mayoría de los manejadores de base de datos: MySQL, PostgreSQL, Oracle, MS SQL Server, entre otras.
- Capacidad de expandir su potencial utilizando módulos.
- Posee documentación en su página oficial la cual incluye descripción y ejemplos de cada una de sus funciones.
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.

- Incluye gran cantidad de funciones.
- No requiere definición de tipos de variables.

1.5.2. Desventajas de PHP:

- Se necesita instalar un servidor web.
- La programación orientada a objetos es aún muy deficiente para aplicaciones grandes.
- Todo el trabajo lo realiza el servidor y no delega al cliente. Por tanto puede ser más ineficiente a medida que las solicitudes de acceso aumenten de número. (11)

Una vez identificado como lenguaje de programación PHP por parte del servidor se tiene en cuenta por parte del cliente **HTML** ya que: “Es un lenguaje estático para el desarrollo de sitios web (acrónimo en inglés de HyperText Markup Language, en español Lenguaje de Marcas Hipertextuales). Desarrollado por el World Wide Web Consortium (W3C). Los archivos pueden tener las extensiones (htm, html)”.

1.5.3. Ventajas de HTML:

- Sencillo que permite describir hipertexto.
- Texto presentado de forma estructurada y agradable.
- Archivos pequeños.
- Despliegue rápido.
- Lenguaje de fácil aprendizaje.
- Lo admiten todos los exploradores.

1.5.4. Desventajas de HTML:

- Lenguaje estático.
- La interpretación de cada navegador puede ser diferente.
- Guarda muchas etiquetas que pueden convertirse en “basura” y dificultan la corrección.
- El diseño es más lento.
- Las etiquetas son muy limitadas.

Por parte del cliente es utilizado además **JavaScript**: Este es un lenguaje interpretado, no requiere compilación. Es similar a Java, aunque no es un lenguaje orientado a objetos, el mismo no dispone de herencias. La mayoría de los navegadores en sus últimas versiones interpretan código Javascript. El

código Javascript puede ser integrado dentro de nuestras páginas web. Para evitar incompatibilidades el World Wide Web Consortium (W3C) diseñó un estándar denominado DOM (en inglés Document Object Model, en su traducción al español Modelo de Objetos del Documento).

1.5.5. Ventajas de JavaScript:

- Lenguaje de scripting seguro y fiable.
- Los scripts tienen capacidades limitadas, por razones de seguridad.
- El código Javascript se ejecuta en el cliente.

1.5.6. Desventajas de JavaScript:

- Código visible por cualquier usuario.
- El código debe descargarse completamente.
- Puede poner en riesgo la seguridad del sitio, con el actual problema llamado XSS (significa en inglés Cross Site Scripting renombrado a XSS por su similitud con las hojas de estilo CSS).

1.6. Sistemas Gestores de Bases de Datos

Un Sistema Gestor de base de datos (SGBD) es un conjunto de programas que permiten crear y mantener una Base de datos, asegurando su integridad, confidencialidad y seguridad. Por tanto debe permitir:

- Definir una base de datos: especificar tipos, estructuras y restricciones de datos.
- Construir la base de datos: guardar los datos en algún medio controlado por el mismo SGBD
- Manipular la base de datos: realizar consultas, actualizarla, generar informes.

Algunas de las características deseables en un Sistema Gestor de base de datos SGBD son:

- Control de la redundancia: La redundancia de datos tiene varios efectos negativos (duplicar el trabajo al actualizar, desperdicia espacio en disco, puede provocar inconsistencia de datos) aunque a veces es deseable por cuestiones de rendimiento.
- Restricción de los accesos no autorizados: cada usuario ha de tener unos permisos de acceso y autorización.
- Cumplimiento de las restricciones de integridad: el SGBD ha de ofrecer recursos para definir y garantizar el cumplimiento de las restricciones de integridad. (12)

Después de tener un pequeño conocimiento de que es un sistema gestor de base de datos se utiliza como herramienta para dicho sistema **PostgreSQL** que es: un sistema de gestión de base de datos relacional orientada a objetos de software libre, publicado bajo la licencia BSD (Berkeley Software Distribution). A continuación veremos algunas de sus características y ventajas:

1.6.1. Ventajas de PostgreSQL:

- Instalación ilimitada.
- Mejor soporte que los proveedores comerciales.
- Ahorros considerables en costos de operación.
- Estabilidad y confiabilidad legendarias.
- Extensible.
- Multiplataforma.
- Diseñado para ambientes de alto volumen.
- Herramientas gráficas de diseño y administración de bases de datos.

1.6.2. Características de PostgreSQL:

- Cumple completamente con ACID.
 - Cumple con ANSI SQL.
 - Integridad referencial.
 - Herencia.
 - Procedimientos almacenados.
 - Respaldo en caliente.
 - Funciones de compatibilidad para ayudar en la transición desde otros sistemas menos compatibles con SQL.
 - Integridad referencial.
 - Procedimientos almacenados.
 - Respaldo en caliente.
 - Herramientas para generar SQL portable para compartir con otros sistemas compatibles con SQL
- (13)

1.7. Entorno de desarrollo integrado (IDE)

Un IDE es un entorno de programación que consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica. Estas pueden ser aplicaciones por sí solas o pueden ser parte de aplicaciones existentes.

Dentro de los IDE a utilizar se tiene **Zend Studio** ya que no es más que: por sus siglas en inglés (Integrated Development Environment) destinado a desarrolladores profesionales. Es un IDE propietario, compatible con las plataformas Linux, MAC y Windows. Incluye todos los componentes necesarios durante el ciclo de vida de una aplicación en PHP. Incluye editor, análisis, depuración, optimizadores de código y herramientas de base de datos. Zend Studio nos permite agiliza el desarrollo web y permite simplificar proyectos complejos.

Dentro de sus características podemos mencionar que es: Excelente completamiento de código, coloreado en la sintaxis del código, administración avanzada de proyectos, múltiples lenguajes, incorpora el Framework de Zend, PHP Documentor, manual de PHP. Integración con subversión, los navegadores, integración avanzada con FTP. Soporte para Web Services, PHP4, PHP5 y SQL.

1.7.1. Ventajas de Zend Estudio:

- Agiliza nuestro trabajo, cuenta con un buen Depurador, infinitas opciones que permiten un desarrollo profesional de nuestras aplicaciones.

1.7.2. Desventajas de Zend Estudio:

- Requiere Licencia de pago, no incluye editor visual HTML, un poco complejo. (14)

1.8. Frameworks

Los Frameworks ayudan en el desarrollo de software, proporcionan una estructura definida la cual ayuda a crear aplicaciones con mayor rapidez. Ayuda a la hora de realizar el mantenimiento del sitio gracias a la organización durante el desarrollo de la aplicación. Los Frameworks son desarrollados con el objetivo de brindarles a los programadores y diseñadores una mejor organización y estructura a sus proyectos.

Los frameworks son diseñados con el intento de facilitar el desarrollo de software, permitiendo a los diseñadores y programadores pasar más tiempo identificando requerimientos de software que tratando con los tediosos detalles de bajo nivel de proveer un sistema funcional. Algunos tipos frameworks desarrollados en PHP:

Un framework permite separar en capas la aplicación:

- La lógica de presentación que administra las interacciones entre el usuario y el software.
- La lógica de dominio o de negocio, que manipula los modelos de datos de acuerdo a los comandos recibidos desde la presentación.
- La lógica de datos que permite el acceso a un agente de almacenamiento persistente u otros.

El framework seleccionado para desarrollar el sistema propuesto es denominado **Pacsoft** que es un ambiente de trabajo que está integrado por diferentes framework como son **ExtJs**, **Doctrine**, **Zend** y **Zend_ext** este último es un framework que se encarga de trabajar con las vistas, está integrado por **ExtJs** debido a que es: " una librería construida con JavaScript que proporciona una interfaz a las famosas librerías de Yahoo, jQuery, y Prototype + Scriptaculous, su potencia radica en la rica colección de componentes para el diseño de GUI's del lado del cliente haciendo uso extensivo de Ajax, su orientación a objetos es intensa, el diseño está completamente separado de la funcionalidad, usa funciones comunes como validación, combobox editables, ventanas arrastables (con minimizar y maximizar) y grillas editables las cuales son muy fáciles de implementar, además tiene una API fácil de usar y Licencias Open source. " (15)

También está integrado por **Doctrine** que este a su vez se encarga del acceso a la base de datos, debido a que "es un potente y completo sistema ORM (Object Relational Mapper) para PHP 5.2+ que incorpora una DBL (capa de abstracción a base de datos) y es usado por la habilidad de escribir opcionalmente las preguntas de la base de datos orientado a objeto, esto les proporciona una alternativa poderosa a diseñadores de SQL que mantiene un máximo de flexibilidad sin requerir la duplicación del código innecesario. También permite exportar una base de datos existente a sus clases correspondientes y también convierte clases (convenientemente creadas siguiendo las pautas del ORM) a tablas de una base de datos." (16)

La utilidad que aporta **zend** a Pacsoft es que además de trabajar con modelo vista controlador, cuenta con módulos para manejar archivos PDF y canales de RSS, trabaja con PHP5 y a su vez estandariza los procesos más frecuentes, dotándolos de gran robustez, facilita el mantenimiento de las aplicaciones y tiene el respaldo de la propia ZEND, creadora de PHP, lo que asegura su continuidad futura tanto como la del propio lenguaje PHP. (17)

El hecho de que se beneficie de estas importantes características con que cuentan cada uno de los diferentes framework por los cuales está compuesto Pacsoft lo hace un ambiente de trabajo óptimo y asequible para el desarrollo de aplicaciones web.

1.9. Servidor Web

Apache es el servidor web hecho por excelencia, su configurabilidad, robustez y estabilidad hacen que cada vez millones de servidores reiteren su confianza en este programa.

1.9.1. Características de Apache:

- Corre en una multitud de Sistemas Operativos, lo que lo hace prácticamente universal.
- Apache es una tecnología gratuita de código fuente abierto.
- Es altamente configurable de diseño modular.
- Trabaja con gran cantidad de Perl, PHP y otros lenguajes de script.
- Tiene una alta configurabilidad en la creación y gestión de logs. Apache permite la creación de ficheros de log a medida del administrador, de este modo puedes tener un mayor control sobre lo que sucede en tu servidor.
- Trabaja con Java y páginas jsp. Teniendo todo el soporte que se necesita para tener páginas dinámicas.

La licencia Apache es una descendiente de la licencias BSD, no es GPL. Esta licencia te permite hacer lo que quieras con el código fuente (productos propietarios) siempre que les reconozcas su trabajo. (18)

La selección de un ambiente de trabajo compuesto por la metodología del Proceso de Desarrollo y Gestión de Proyectos de Software, la herramienta CASE de modelado Visual Paradigm, y un conjunto de lenguajes como PHP, HTML, Javascript, así como la utilización del gestor de base de datos PostgreSQL y el servidor web Apache, es el resultado de un estudio hecho por los compañeros del centro UCID para lograr un mayor acoplamiento, intercambio de información, así como hacer más estructurado y lograr un estándar de trabajo en el centro.

1.10. Conclusiones Parciales

A través del análisis realizado mediante el desarrollo de este capítulo hemos abordado conceptos asociados al problema; se realizó un estudio meticuloso sobre los diferentes sistemas en Cuba y en el mundo que abarcan la gestión y control de las redes, determinando sus ventajas y desventajas en este proceso, para de esta forma descartar los que no aportaban una solución. Además se realizó un estudio de las herramientas, tecnologías y la metodología propuestas por la dirección del centro (UCID), permitiendo sentar las bases para el desarrollo de la aplicación.

Capítulo 2: Características del sistema

Introducción

A través de este capítulo se podrá realizar una descripción de los procesos del negocio del producto en relación al campo de acción determinado, en los cuales estarán presentes todo lo que esto involucra, diagramas, reglas y descripciones. También se abordará acerca del sistema en cuestión, definiendo aquí los requerimientos funcionales y no funcionales que debe cumplir la aplicación una vez quede implementada.

2.1. Procesos de negocio

Un proceso de desarrollo de software es una definición del conjunto completo de actividades necesarias para convertir los requisitos de usuario en un conjunto consistente de artefactos que conforman un producto software, y para convertir los cambios sobre esos requisitos en un nuevo conjunto consistente de artefactos.

2.1.1. Mapa de procesos del negocio

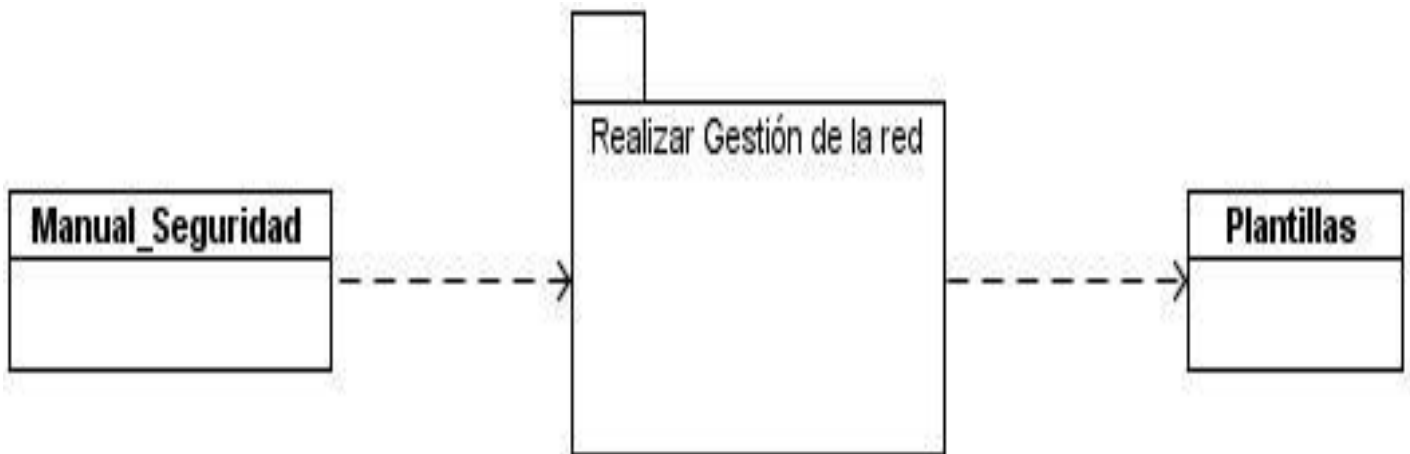


Fig. 3 Visión general de todos los procesos

Proceso Realizar gestión de la red

El objetivo de este proceso es realizar la gestión de la red correspondiente al manual de seguridad informática de las FAR. La figura 4 muestra las tareas de este proceso.

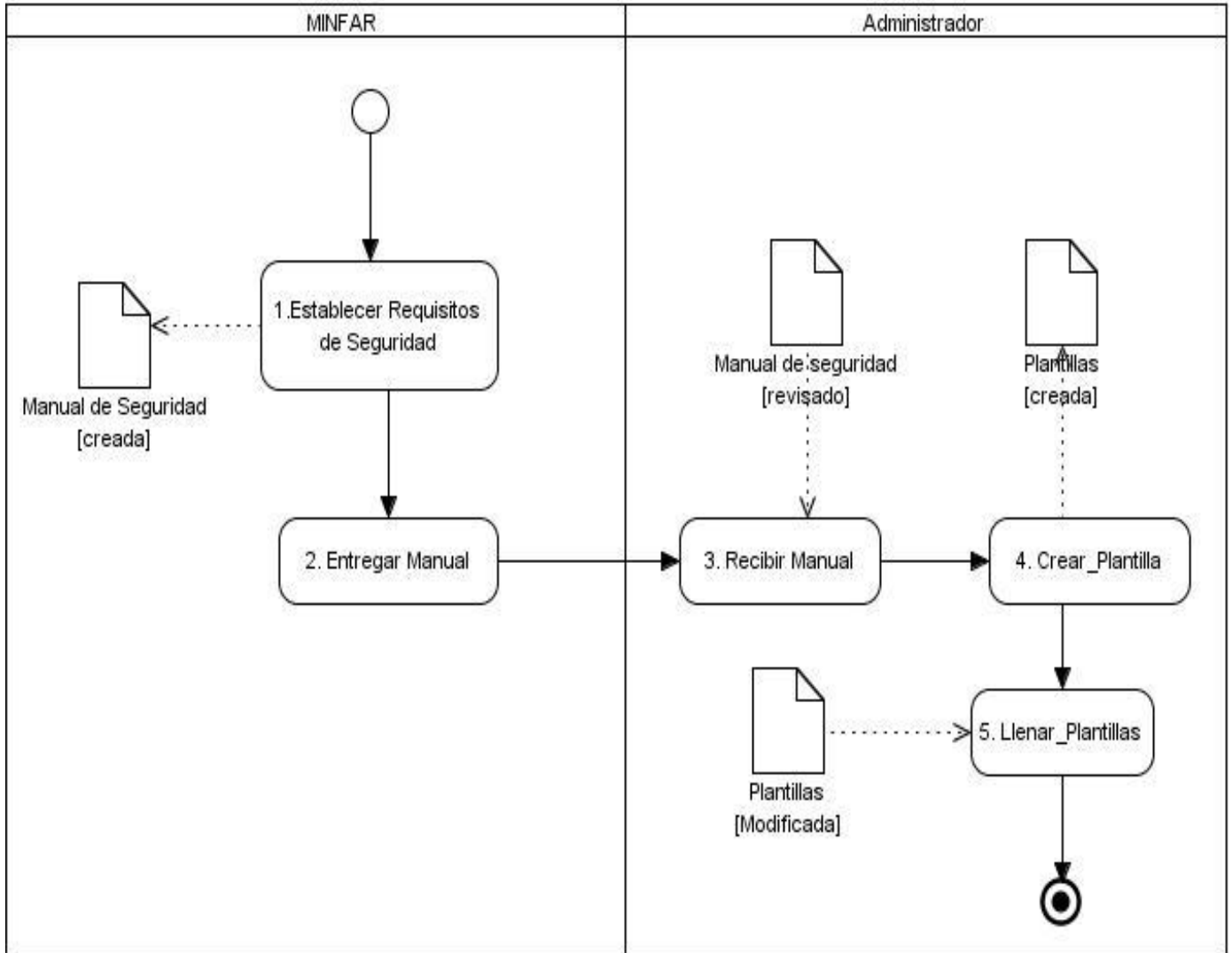


Fig. 4 Proceso Realizar gestión de la red.

2.2. Modelo Conceptual

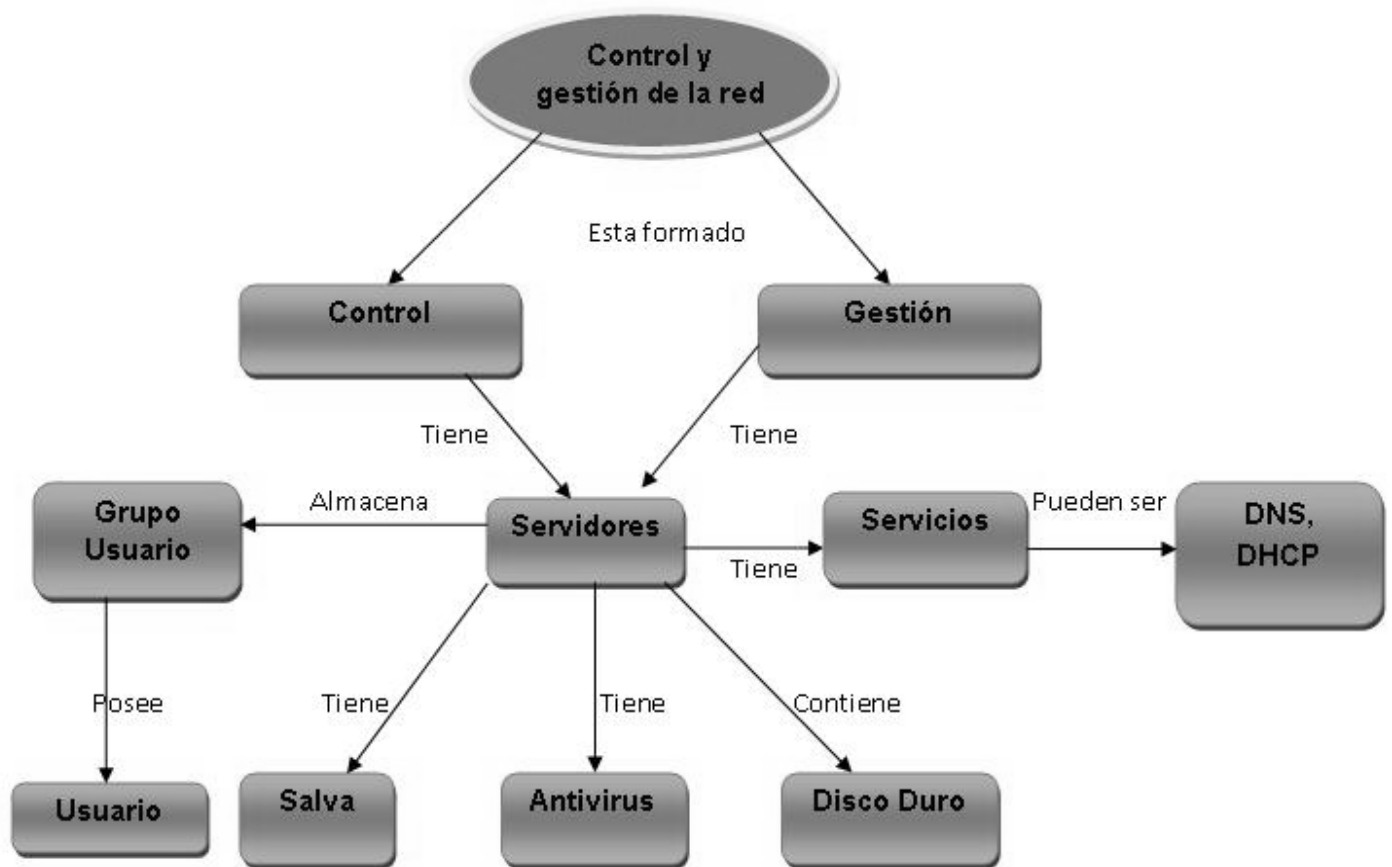


Fig.5 Mapa Conceptual.

2.2.1. Conceptos del modelo conceptual:

Control: Es el proceso para determinar lo que se está llevando a cabo, valorizándolo y si es necesario, aplicando medidas correctivas de manera que la ejecución se desarrolle de acuerdo con lo planeado.

Gestión: En términos generales, por el término de gestión se referirá a la acción y al efecto de administrar o gestionar un negocio. A través de una gestión se llevarán a cabo diversas diligencias, trámites, las cuales, conducirán al logro de un objetivo determinado, de un negocio o de un deseo que lleva largo tiempo en carpeta, como se dice popularmente.

Servidor: Es un tipo de software que realiza ciertas tareas en nombre de los usuarios. El término servidor ahora también se utiliza para referirse al ordenador físico en el cual funciona ese software, una máquina cuyo propósito es proveer datos de modo que otras máquinas puedan utilizar esos datos.

Servicio: Es el conjunto de prestaciones que un ordenador o sistema informático facilita a otros sistemas informáticos, o a otros ordenadores que están conectados a él.

Antivirus: es un programa informático específicamente diseñado para detectar y eliminar virus. Los reconoce, sabe cómo actúan y también sabe cómo eliminarlos. Si el antivirus no se actualiza no sería capaz de reconocer los nuevos virus que aparecen diariamente y perdería casi toda su eficacia, ya que cada día aparecen más de 20 virus nuevos, por lo que debes de asegurar que tu antivirus este siempre activo y muy especialmente cuando se esta conectado en la Internet.

Disco duro: también conocido como Hard Disk o HD, es el dispositivo magnético interno de la computadora en el que se almacena gran cantidad de información (programas, archivos, aplicaciones en general) y que es la parte principal de lo que se conoce como memoria secundaria.

DNS: (Domain Name System). Sistema de Nombres de Dominio. Es un servicio de búsqueda de datos de uso general, distribuido y multiplicado. Su utilidad principal es la búsqueda de direcciones IP de sistemas centrales (“hosts”) basándose en los nombres de estos. El estilo de los nombres de “hosts” utilizado actualmente en Internet es llamado “nombre de dominio”

DHCP: (Dynamíc Host Configuración Protocol). Protocolo de configuración dinámica de host. Protocolo que usan las computadoras para obtener información de configuración. El DHCP permite asignar una dirección IP a una computadora sin requerir que un administrador configure la información sobre la computadora en la base de datos de un servidor.

Usuario: Es utilizado generalmente en el campo de la tecnología informática para referirse a quien utiliza determinado hardware y/o software, mediante el cual obtiene un servicio. Quizá sea complejo entender que para una computadora un usuario puede no ser un ser humano: en este caso un programa u otra computadora podrían ser considerados usuarios, siendo que la PC realiza una abstracción a través de la cual brinda recursos de hardware, software mediante permisos de acceso a estos. (19)

2.3. Requerimientos Funcionales

Los requerimientos funcionales son capacidades o condiciones que el sistema debe cumplir. Estos se mantienen invariables sin importar con que propiedades o cualidades se relacionen. Por lo antes expresado se llega a la conclusión que el sistema debe cumplir los siguientes requisitos

El sistema debe:

RF1: Gestionar Configuración básica del servidor

Permitir que el administrador pueda insertar, modificar, buscar y eliminar los datos del servidor.

RF2: Gestionar configuración de la red

Permitir que el administrador pueda insertar, modificar, buscar y eliminar los datos relacionados con la configuración de la red.

RF3: Gestionar configuración de disco duro.

Permitir que el administrador pueda insertar, modificar, buscar y eliminar datos del disco duro al sistema

RF4: Gestionar usuarios de los servidores

Permitir que el administrador pueda insertar, modificar, buscar y eliminar los datos de los usuarios de los servidores.

RF5: Gestionar grupos de usuarios

Permitir que el administrador pueda insertar, modificar, buscar y eliminar los datos de los grupos de usuarios.

RF6: Gestionar salva y restauración

Permitir que el administrador pueda insertar, modificar, buscar y eliminar los datos relacionados con la salva y restauración.

RF7: Gestionar dispositivos de salva

Permitir que el administrador pueda insertar, modificar, buscar y eliminar los datos de dispositivos de salva.

RF8: Gestionar datos del antivirus

Permitir que el administrador pueda insertar, modificar, buscar y eliminar datos del antivirus.

RF9: Gestionar servicio de DNS

Permitir que el administrador pueda insertar, modificar, buscar y eliminar servicios de DNS al sistema.

RF10: Gestionar servicio de DHCP

Permitir que el administrador pueda insertar, modificar, buscar y eliminar servicios de DHCP al sistema.

2.4. Requerimientos no Funcionales

Captura de Requerimientos No Funcionales: Los requisitos no funcionales especifican propiedades del sistema, como restricciones del entorno o de la implementación, rendimiento, dependencias de la plataforma, facilidad de mantenimiento, extensibilidad y fiabilidad.

Estos requerimientos son propiedades o cualidades que el producto debe tener. Por esta razón debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable.

Tomando en cuenta esta descripción y basándonos en las categorías en las que se encuentran divididas, entonces quedarían definidos de esta manera:

RNF1 Apariencia o interfaz externa

- Debe mostrar una interfaz gráfica agradable al usuario, así como un diseño sencillo para que el usuario realice cualquier tipo de interacción con el sistema de manera fácil.
- Que el sistema sea capaz de mostrar mensajes con la información que al usuario le sean necesarias, los mensajes pueden ser de error, ayuda, alerta

RNF2 Seguridad

- La transmisión de los datos debe ser de una manera segura, por lo que se le dará acceso a la información al personal autorizado.
- Los usuarios antes de entrar al sistema se tienen que autenticar.
- Cada usuario tendrá acceso a la interfaz que tenga permiso para visitarla.

RNF3 Usabilidad

- El sistema debe ser de uso fácil para cualquier persona, tanto personas con conocimientos mínimos en el manejo de la computación, como personas con una vasta experiencia en este tipo de ambiente.

RNF4 Software

Software del Sistema

En la PC del cliente debe estar instalado:

- El Sistema Gestor de Base de Datos PostgreSQL_8.3.
- Recomendado Apache 2.0 como servidor web.
- Mozilla_Firefox como explorador web u otro que implemente el DOM 2.0 y que soporte JavaScript.

RNF5 Portabilidad

- El sistema debe ser multiplataforma, debe correr tanto en la plataforma Windows como en Linux.

2.5. Conclusiones Parciales

A través de este capítulo se pudieron emplear algunas técnicas con el fin de identificar requisitos los cuales permitieron determinar las funcionalidades principales del sistema a construir. Se identificaron además los procesos del negocio los cuales permiten definir las actividades que ayudan a construir el producto final de manera eficiente. Para ello, se realizó una descripción textual de cada uno de los procesos, con el fin de desglosar la información lo más posible y hacer más fácil el trabajo. También en este capítulo se pudieron tratar una serie de conceptos que ayudaron al desarrollo de las actividades.

Capítulo 3: Diseño del sistema

Introducción

En el presente capítulo estará contenido todo lo referente al análisis y diseño del sistema, el cual se realiza como una parte de la propuesta de solución. Para dar cumplimiento a esta propuesta se estará desarrollándose los diagramas de clases y de interacción como son los de secuencia y del diseño. Se presentará además el modelo de datos como parte del Modelamiento de la Base de Datos.

3.1. Diagramas de Interacción

Los diagramas de interacción son diagramas que modelan aspectos dinámicos del sistema.

El diagrama de secuencia de un sistema es una representación que muestra, en determinado escenario los eventos generados por actores externos, su orden y los eventos internos del sistema. (20)

A continuación se muestran los diagramas de secuencia correspondiente a cada requisito:

RF1: Gestionar Configuración básica del servidor.

Escenario Insertar servidor.

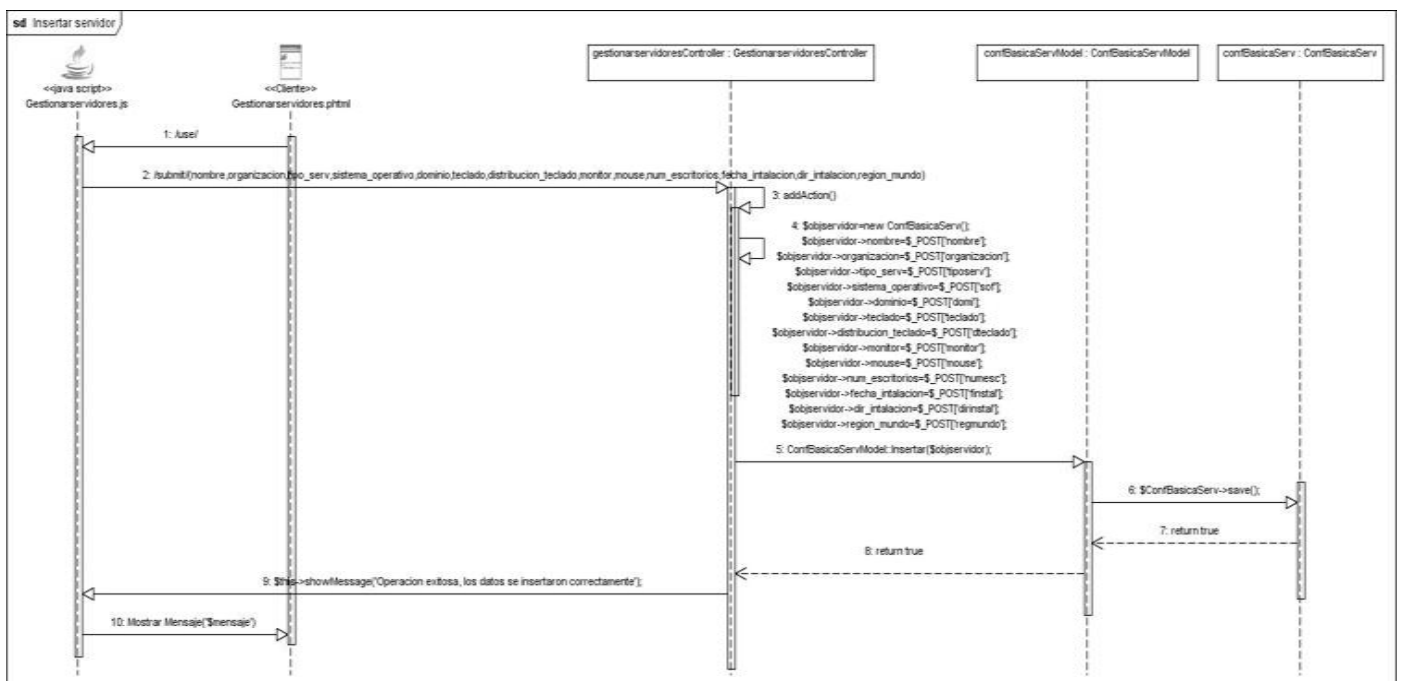


Fig.6 Diagrama de Secuencia Insertar Servidor.

Escenario Modificar servidor.

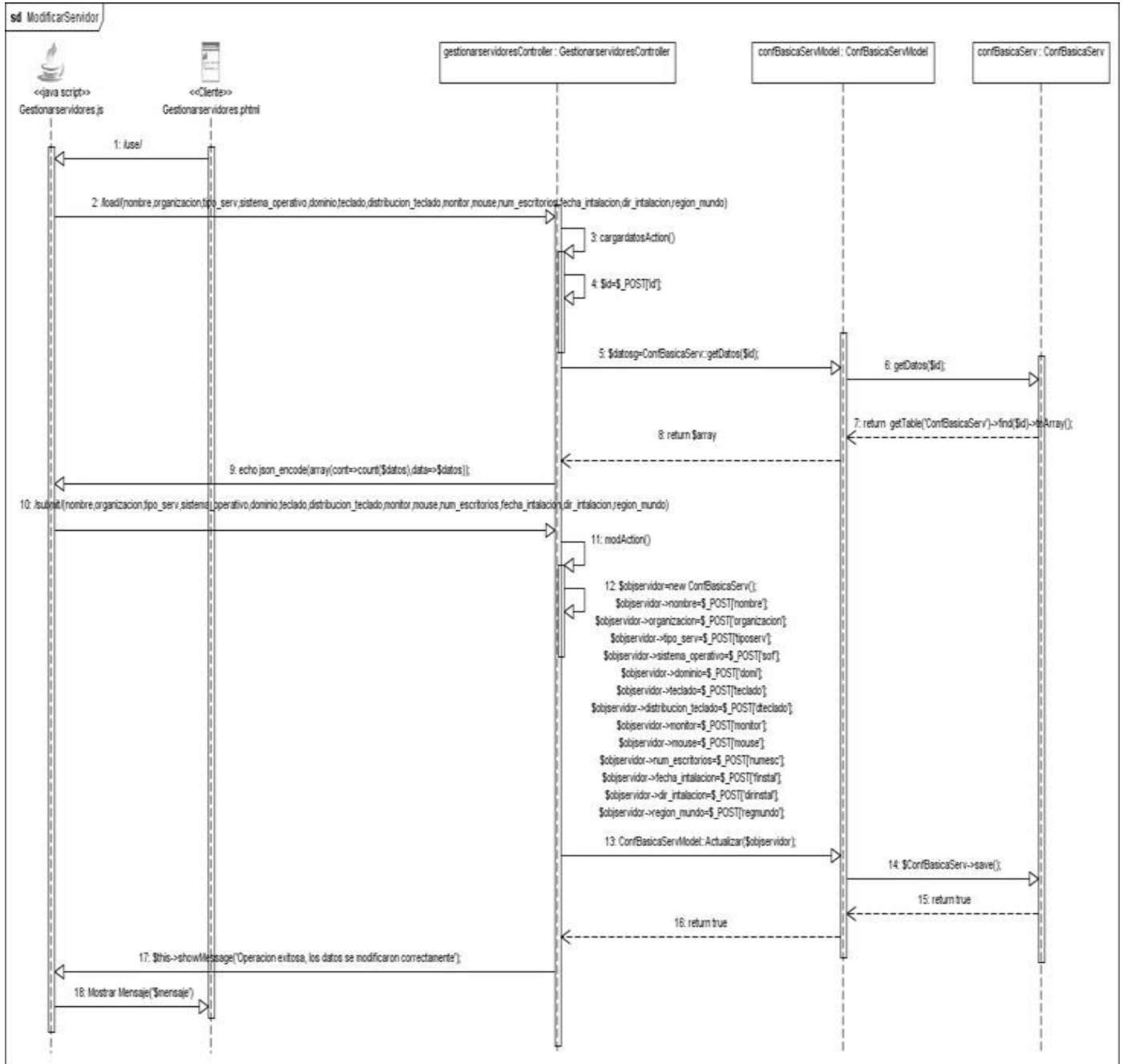


Fig. 7 Diagrama de Secuencia Modificar Servidor.

Escenario Eliminar servidor.

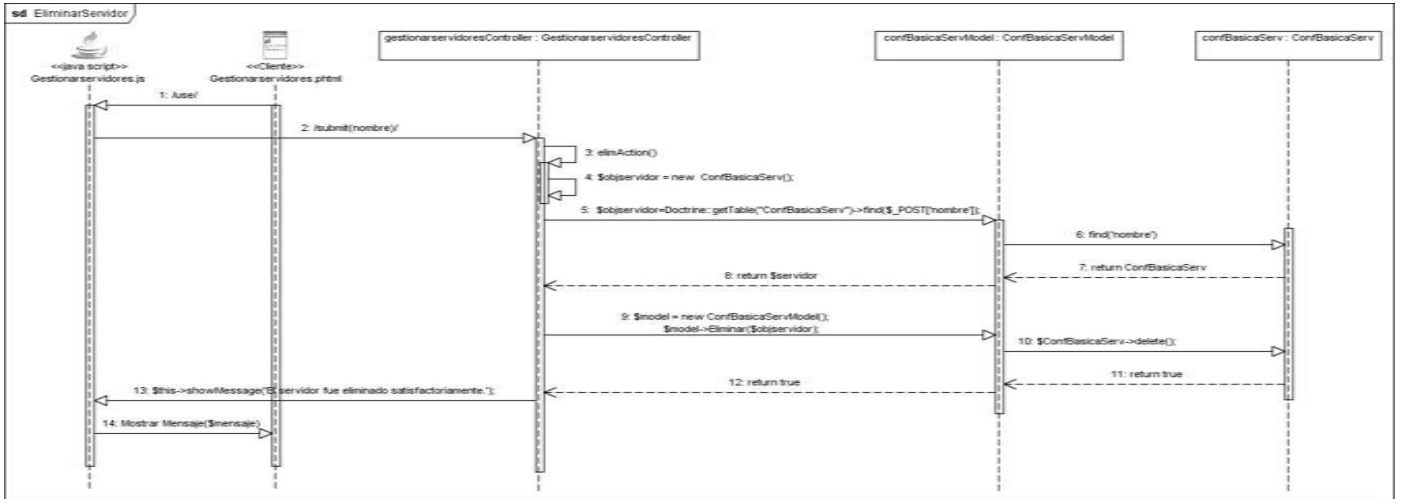


Fig.8 Diagrama de Secuencia Eliminar Servidor.

Escenario Buscar servidor.

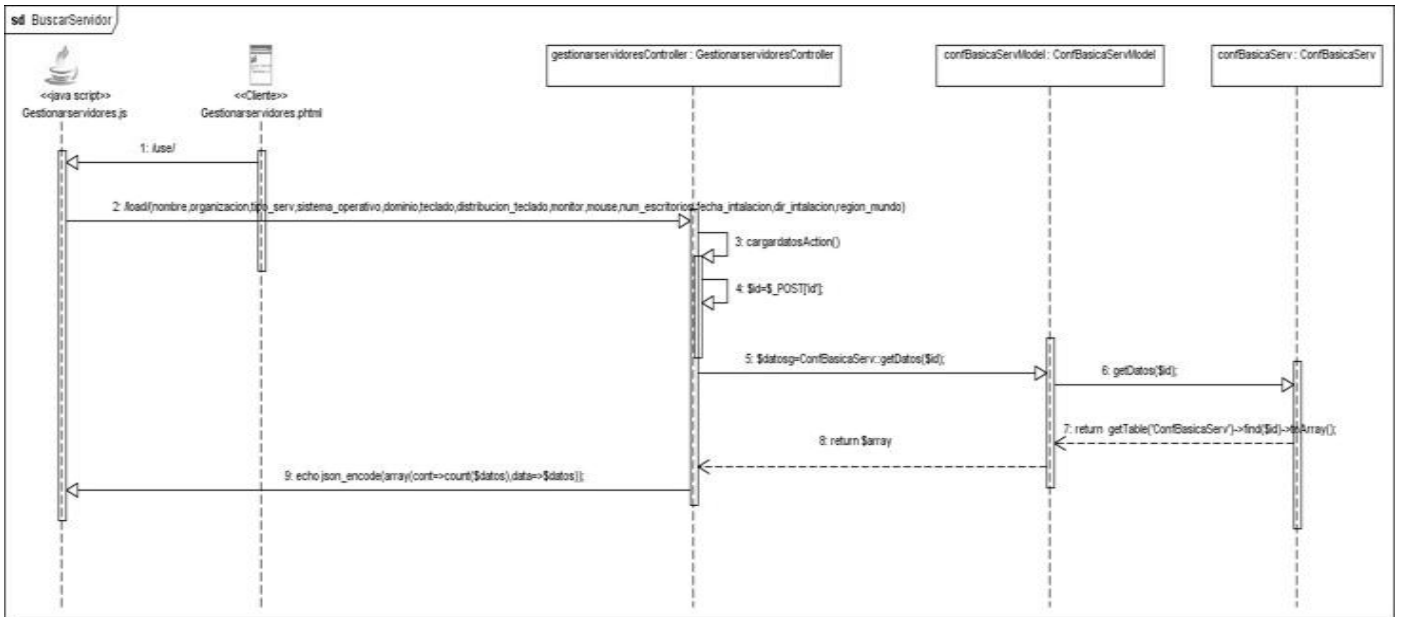


Fig.9 Diagrama de Secuencia Buscar Servidor.

3.2. Diseño

El diseño se utiliza para que el programador en cuestión pueda realizar una correcta implementación del sistema a desarrollar. Representa la parte estática del sistema así como las clases y sus relaciones. Para realizar el diseño propuesto para el posterior desarrollo de dicho sistema se siguió adecuadamente con la arquitectura propuesta.

3.2.1. Modelo de diseño

A continuación se mostrará los diagramas de clase del diseño de cada uno de los requisitos funcionales.

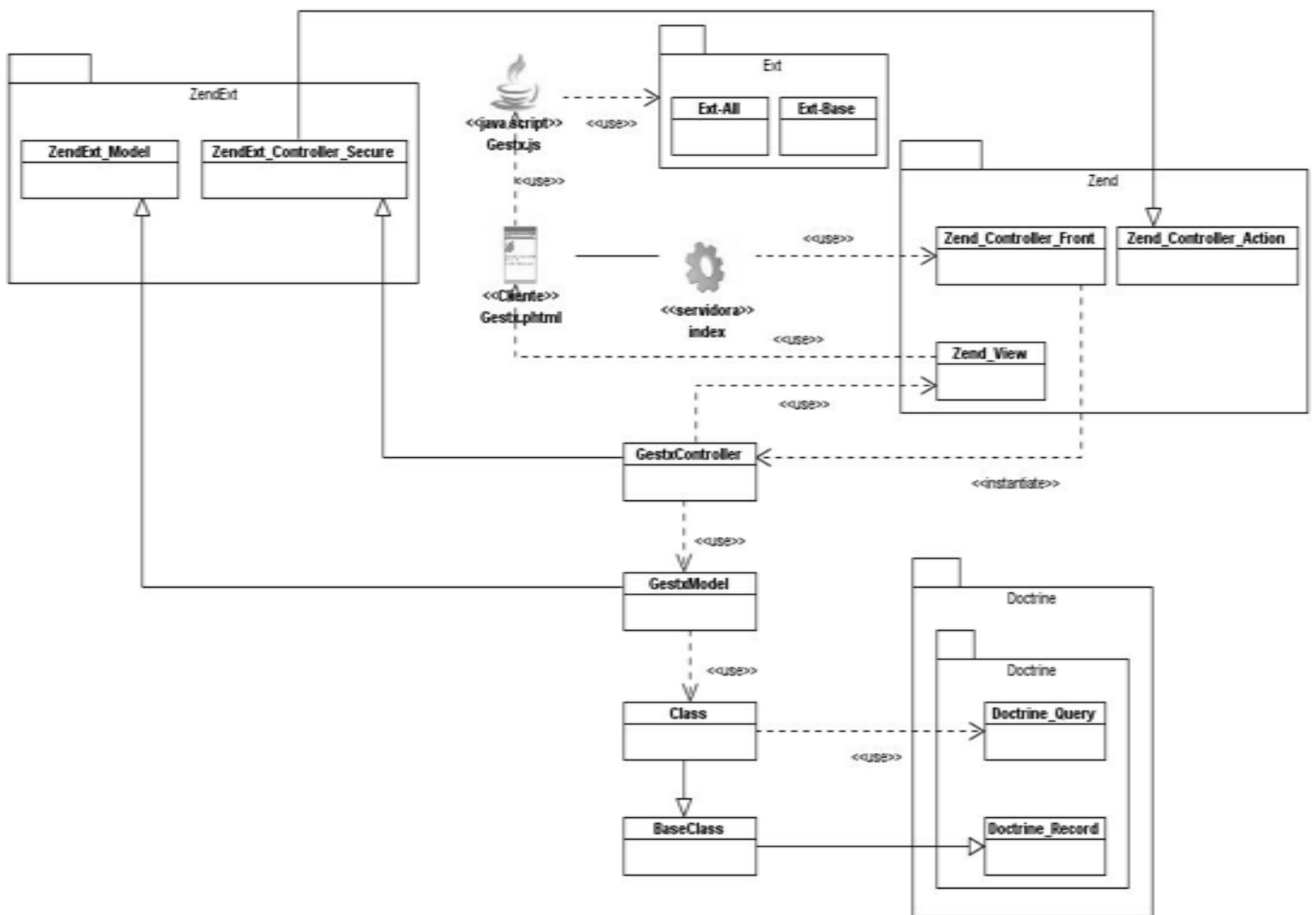


Fig.10 Diagrama de clase Genérico.

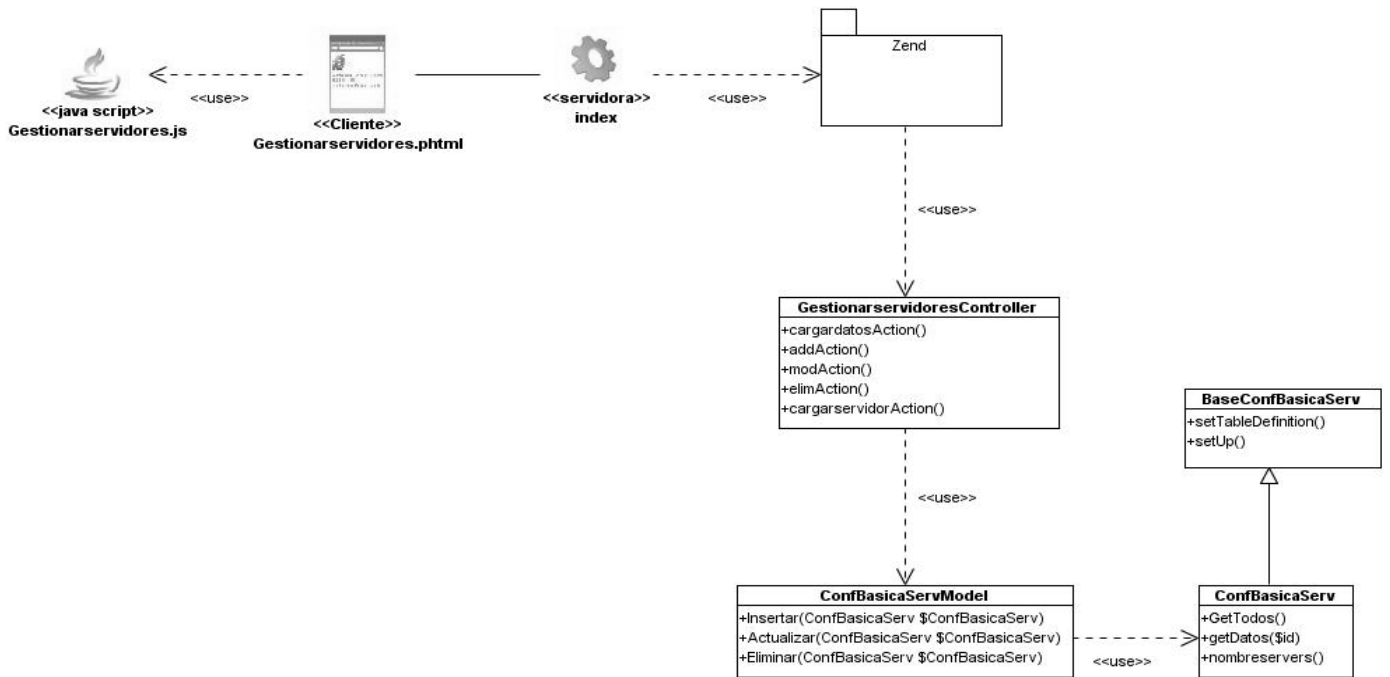


Fig. 11 Diagrama de clase del diseño “Gestionar configuración básica del servidor”.

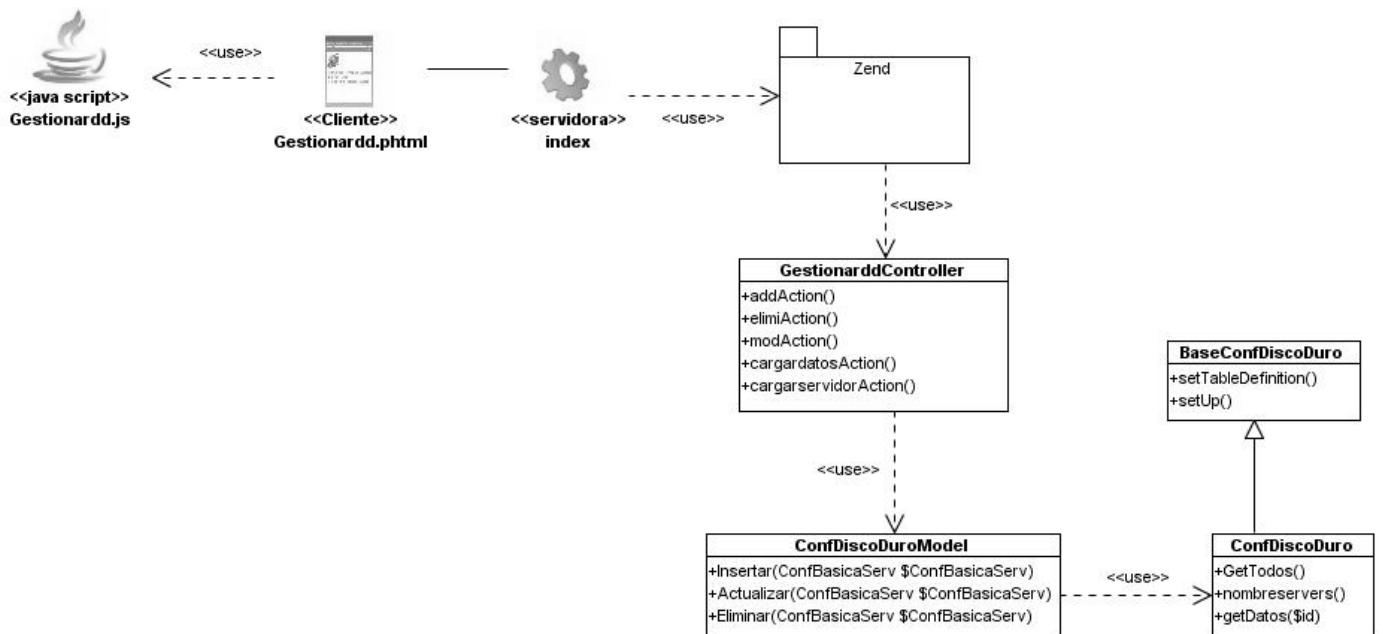


Fig. 12 Diagrama de clase del diseño “Gestionar configuración de disco duro”.

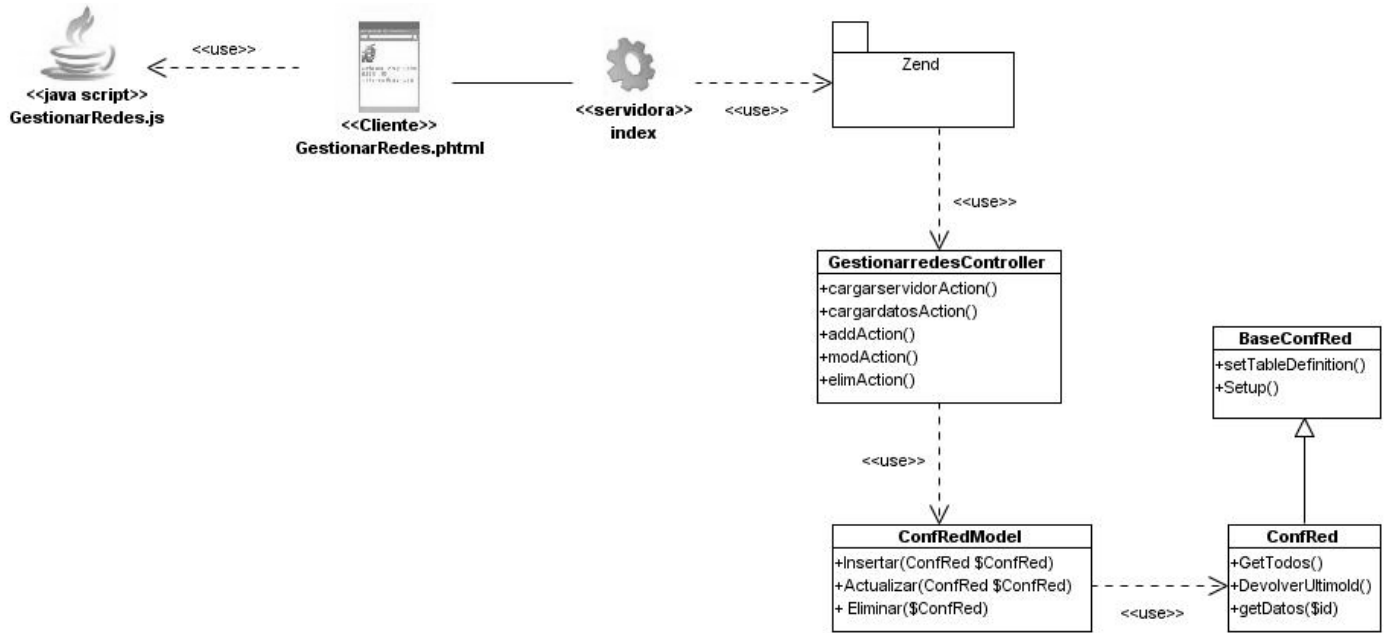


Fig. 13 Diagrama de clase del diseño “Gestionar configuración de la red”.

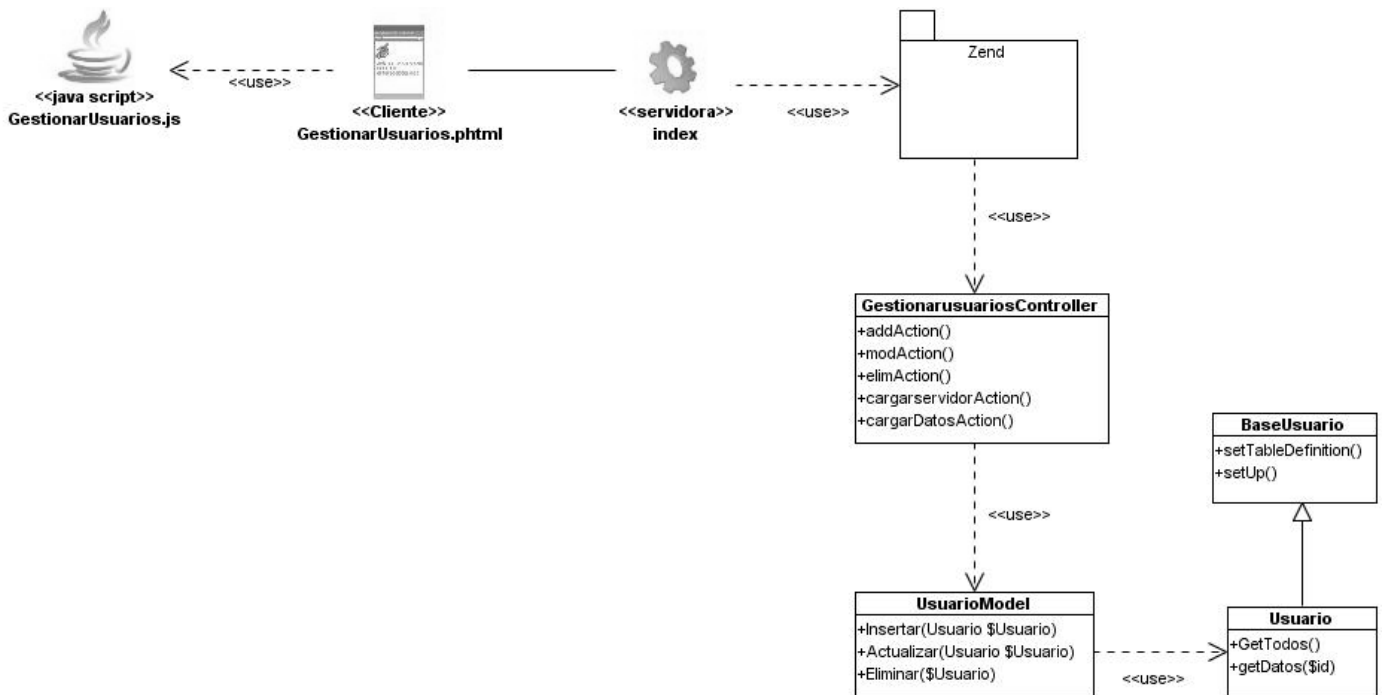


Fig. 14 Diagrama de clase del diseño “Gestionar usuarios de los servidores”.

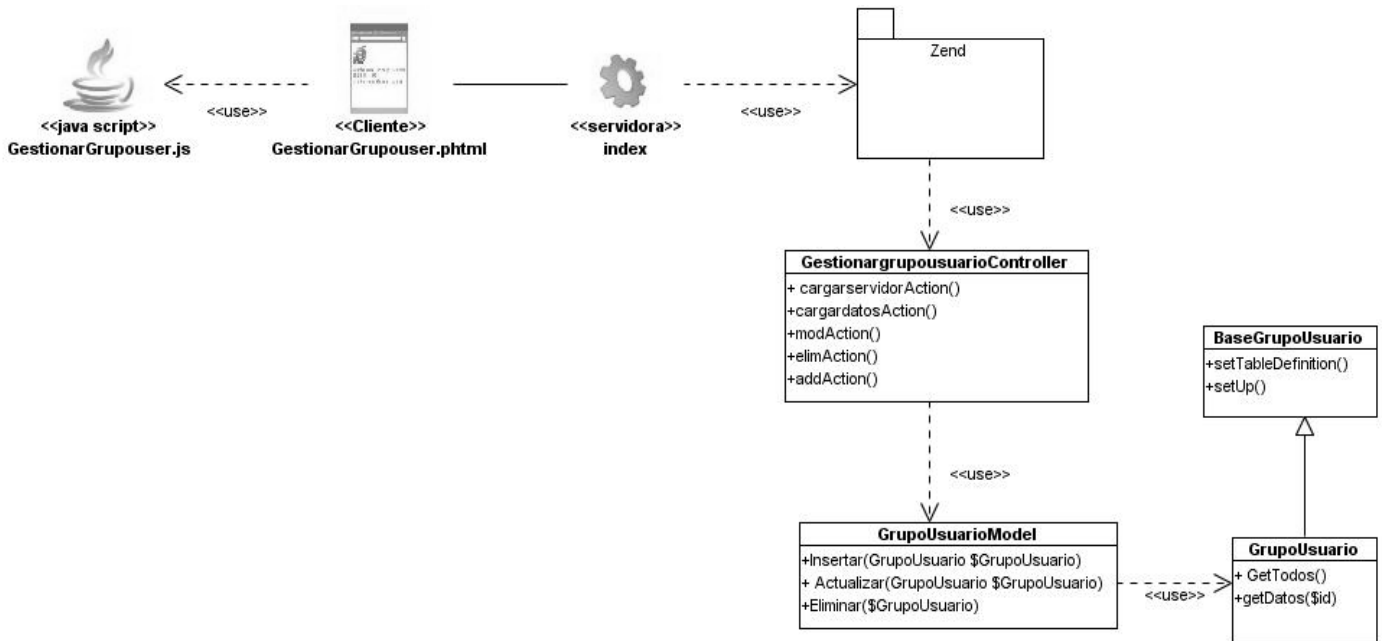


Fig. 15 Diagrama de clase del diseño “Gestionar grupos de usuarios”.

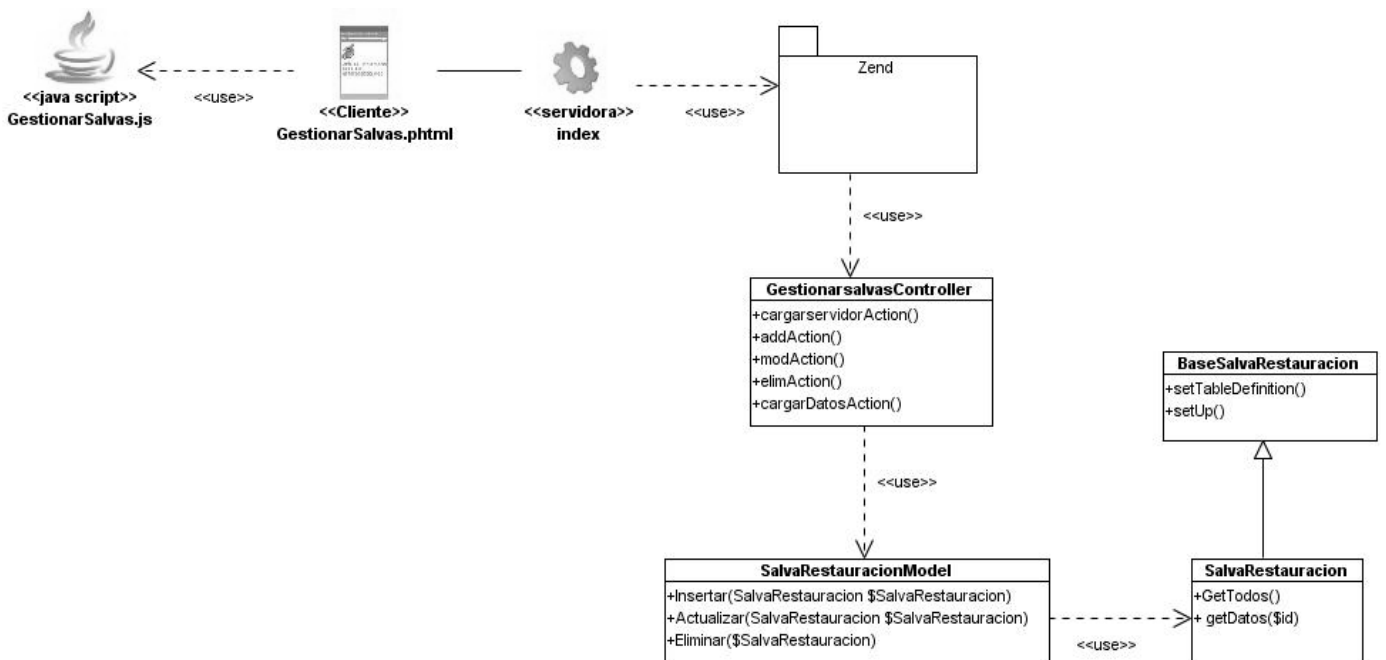


Fig. 16 Diagrama de clase del diseño “Gestionar salva y restauración”.

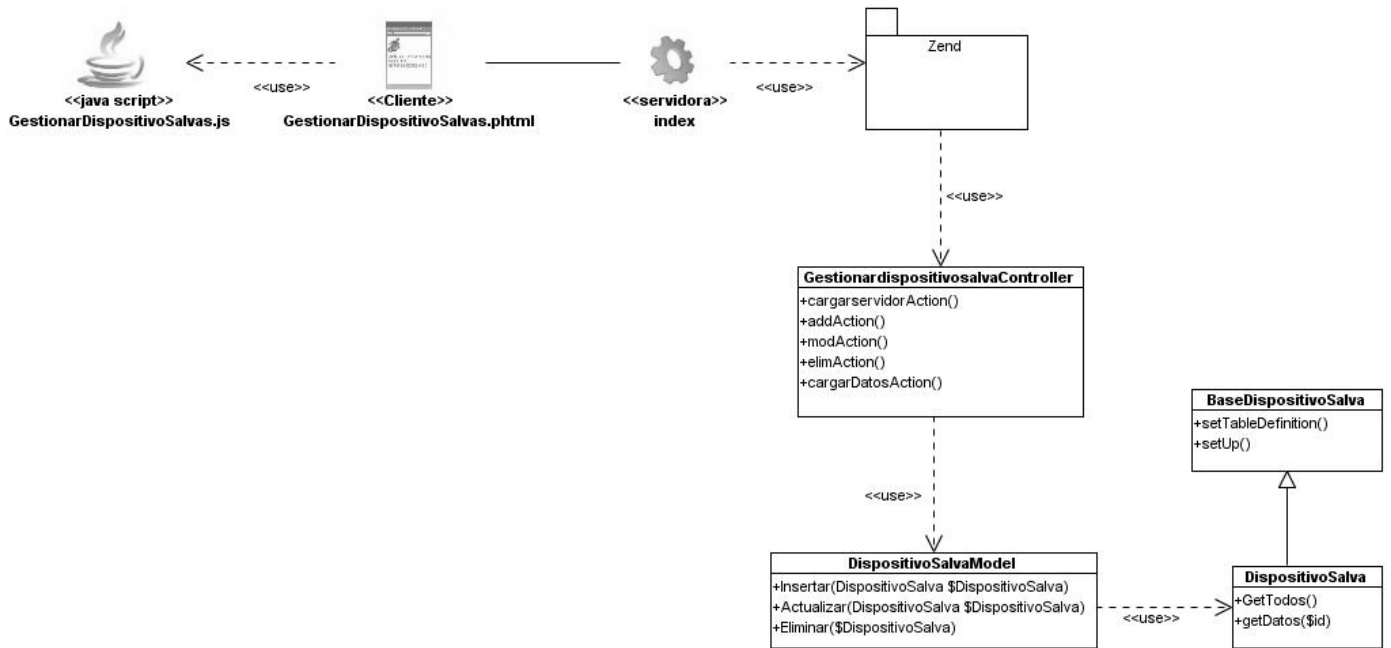


Fig. 17 Diagrama de clase del diseño “Gestionar dispositivos de salva”.

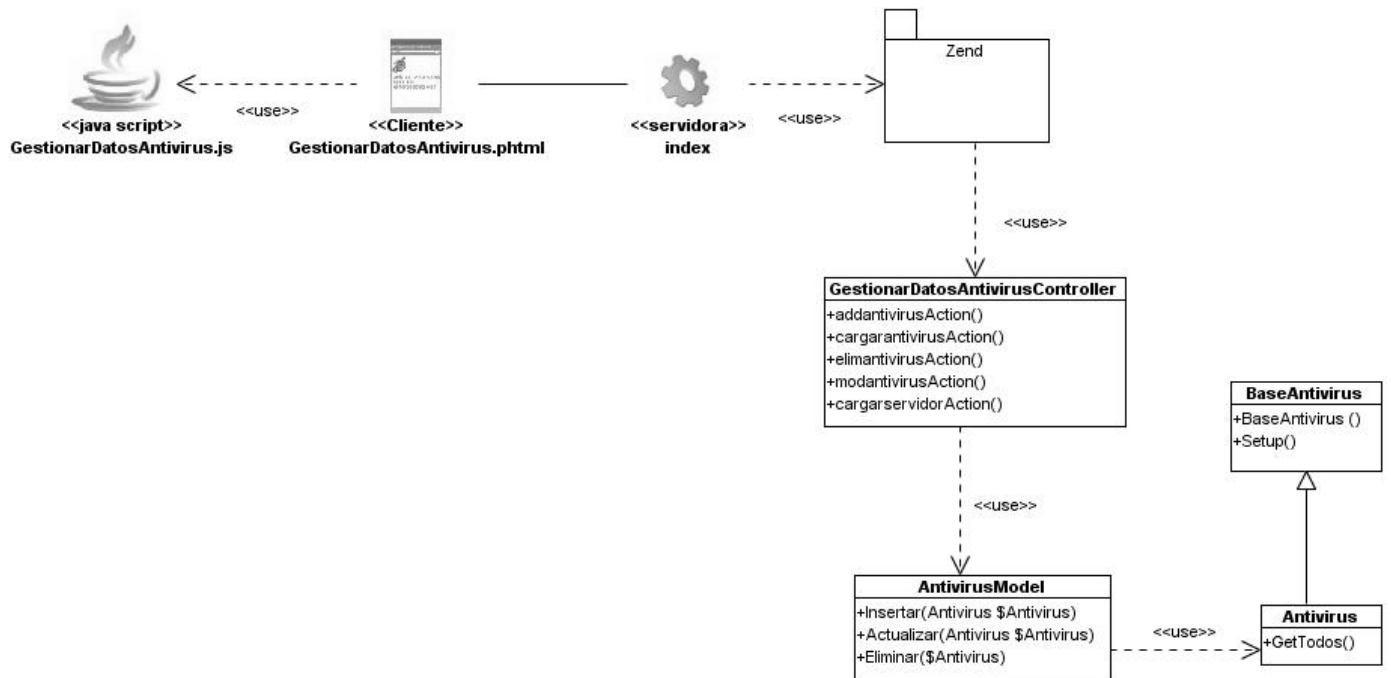


Fig. 18 Diagrama de clase del diseño “Gestionar datos del antivirus”.

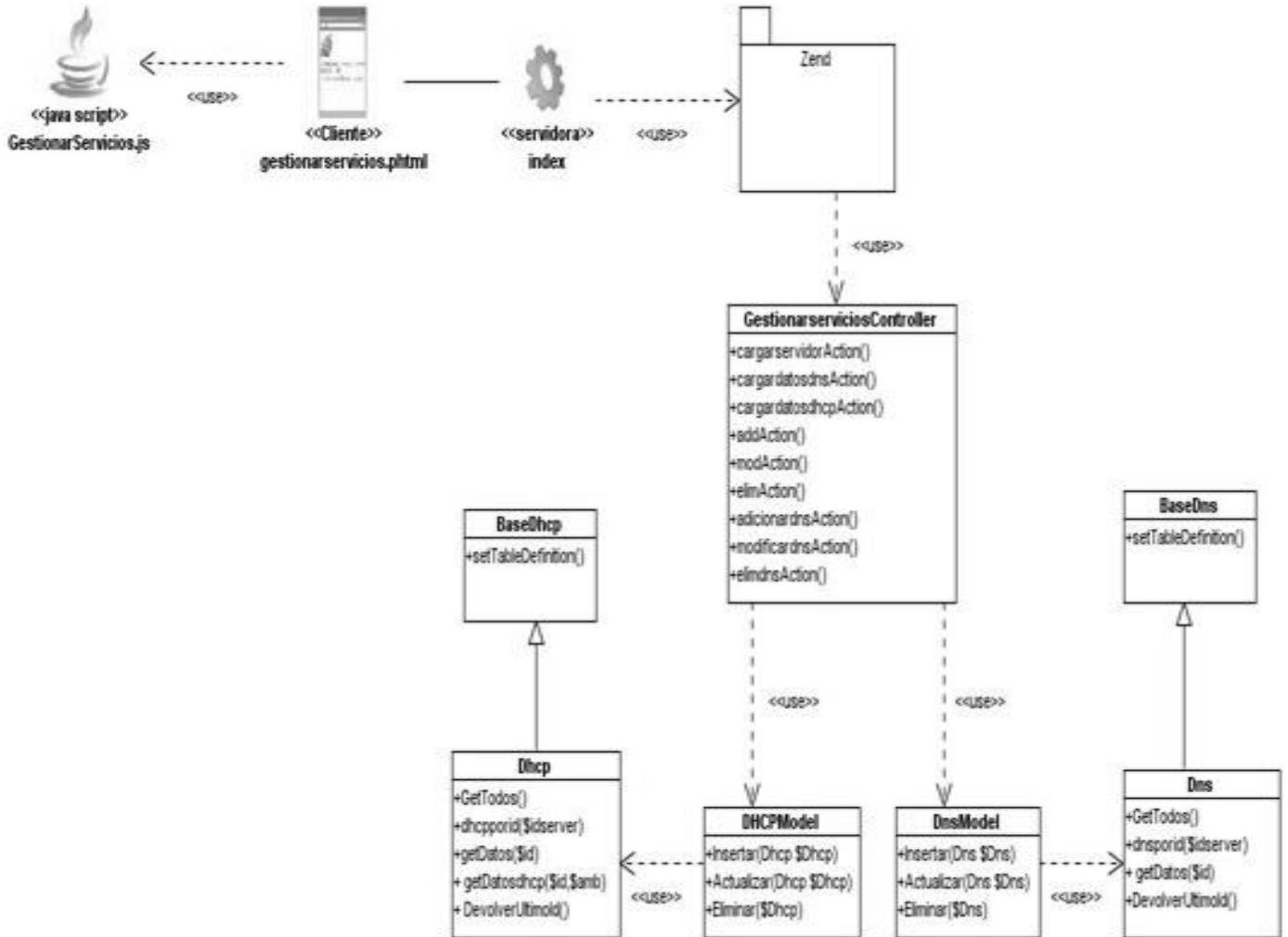


Fig. 19 Diagrama de clase del diseño “Gestionar servicios”.

3.2.2. Patrón Modelo-Vista-Controlador (MVC)

MVC es un patrón de diseño de arquitectura de software usado principalmente en aplicaciones que manejan gran cantidad de datos y transacciones complejas donde se requiere una mejor separación de los conceptos para que el desarrollo este estructurado de una mejor manera, facilitando la programación

en diferentes capas de manera paralela e independiente. MVC sugiere la separación del software en 3 estratos.

Modelo: Es la representación de la información que maneja la aplicación. El modelo en sí son los datos puros que puestos en un contexto del sistema proveen de información al usuario o a la aplicación misma.

Vista: Es la representación del modelo en forma grafica disponible para la interacción con el usuario. En el caso de una aplicación web la "Vista" es la página HTML con contenido dinámico sobre el cual el usuario puede realizar operaciones.

Controlador: Es la capa encargada de manejar y responder las solicitudes del usuario, procesando la información necesaria y modificando el Modelo en caso de ser necesario.

Las **Vistas** son responsables de:

- Recibir datos del modelo y lo muestra al usuario.
- Tienen un registro de su controlador asociado (normalmente porque además lo instancia).

El **Modelo** es el responsable de:

- Acceder a la capa de almacenamiento de datos. Lo ideal es que el modelo sea independiente del sistema de almacenamiento.
- Definir las reglas de negocio (la funcionalidad del sistema).
- Llevar un registro de las vistas y controladores del sistema.

El **Controlador** es responsable de:

- Recibir los eventos de entrada.
- Contiene reglas de gestión de eventos, del tipo "SI Evento Z, entonces Acción W". Estas acciones pueden suponer peticiones al modelo o a las vistas.

En la siguiente figura se muestra la estructura del patrón Modelo-Vista-Controlador.

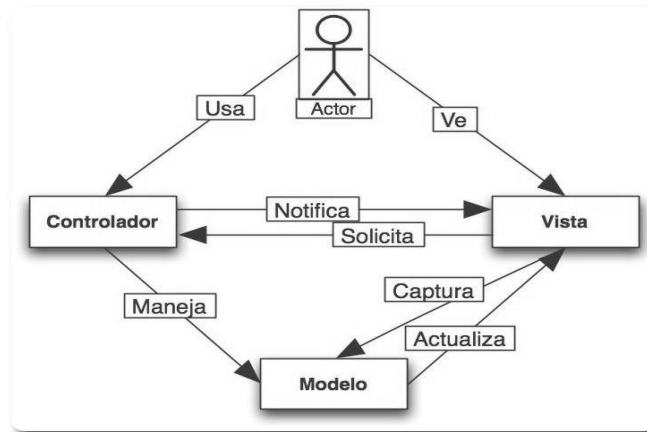


Fig. 20 Estructura del patrón Modelo-Vista-Controlador.

3.2.3. Patrones de Diseño

Los patrones de diseño constituyen soluciones exitosas a problemas comunes que se presentan durante el diseño de una aplicación. Evitan cometer errores, son soluciones concretas a problemas frecuentes que se presentan, están basados en la experiencia acumulada de expertos de desarrollo de software. A continuación una breve descripción de los utilizados en el desarrollo del sistema:

Vista: Implementa el patrón Decorator en la clase Zend_View, encargada de asignarle responsabilidades a objetos de manera dinámica y configurarlos con nuevos atributos.

Controlador: Zend Framework tiene implementado el patrón Front Controller que implica que todas las solicitudes son dirigidas a un único script PHP que se encarga de instanciar al controlador frontal y redirigir las llamadas.

Modelo: El doctrine tiene el patrón Active Record es un objeto que representa una fila en una tabla de una base de datos, encapsula su acceso a la base de datos y añade lógica de negocio. Es la aproximación más obvia, poniendo el acceso a la base datos en el propio objeto de negocio. De este modo es evidente como manipular la persistencia a través del mismo.

Creador: Este patrón como su nombre lo indica es el que crea, el guía la asignación de responsabilidades relacionadas con la creación de objetos.

Experto: Patrón que se usa más que cualquier otro al asignar responsabilidades; es un principio básico que suele utilizarse en el diseño orientado a objetos. Con él no se pretende designar una idea oscura ni extraña; expresa simplemente la "intuición" de que los objetos hacen cosas relacionadas con la información que poseen.

Bajo Acoplamiento: Este patrón es un principio que asigna la responsabilidad de controlar el flujo de eventos del sistema, a clases específicas. Esto facilita la centralización de actividades (validaciones, seguridad). El controlador no realiza estas actividades, las delega en otras clases con las que mantiene un modelo de alta cohesión. Un error muy común es asignarle demasiada responsabilidad y alto nivel de acoplamiento con el resto de los componentes del sistema.

Alta cohesión: La cohesión es una medida de la fuerza con la que se relacionan las clases y el grado de focalización de las responsabilidades de un elemento, cada elemento de nuestro diseño debe realizar una labor única dentro del sistema, no desempeñada por el resto de los elementos y auto-identificable, una clase con baja cohesión hace muchas cosas no relacionadas o hace demasiado trabajo.

Controlador: Es un evento generado por actores externos. Se asocian con operaciones del sistema, operaciones del sistema como respuestas a los eventos del sistema, tal como se relacionan los mensajes y los métodos.

3.3. Diseño de la Base de Datos

El diseño de la Base de Datos es de vital importancia, en ella se almacena toda la información de los datos que son necesarios para darle solución al problema existente. Permite obtener toda la información que se quiera recuperar del sistema.

3.3.1. Modelo físico de datos. Diagrama Entidad-Relación.

El Modelo de Datos es un conjunto de conceptos, reglas y convenciones que permiten describir y manipular los datos de un cierto mundo real que se desea almacenar en la Base de Datos.

A continuación se muestra el Diagrama Entidad-Relación

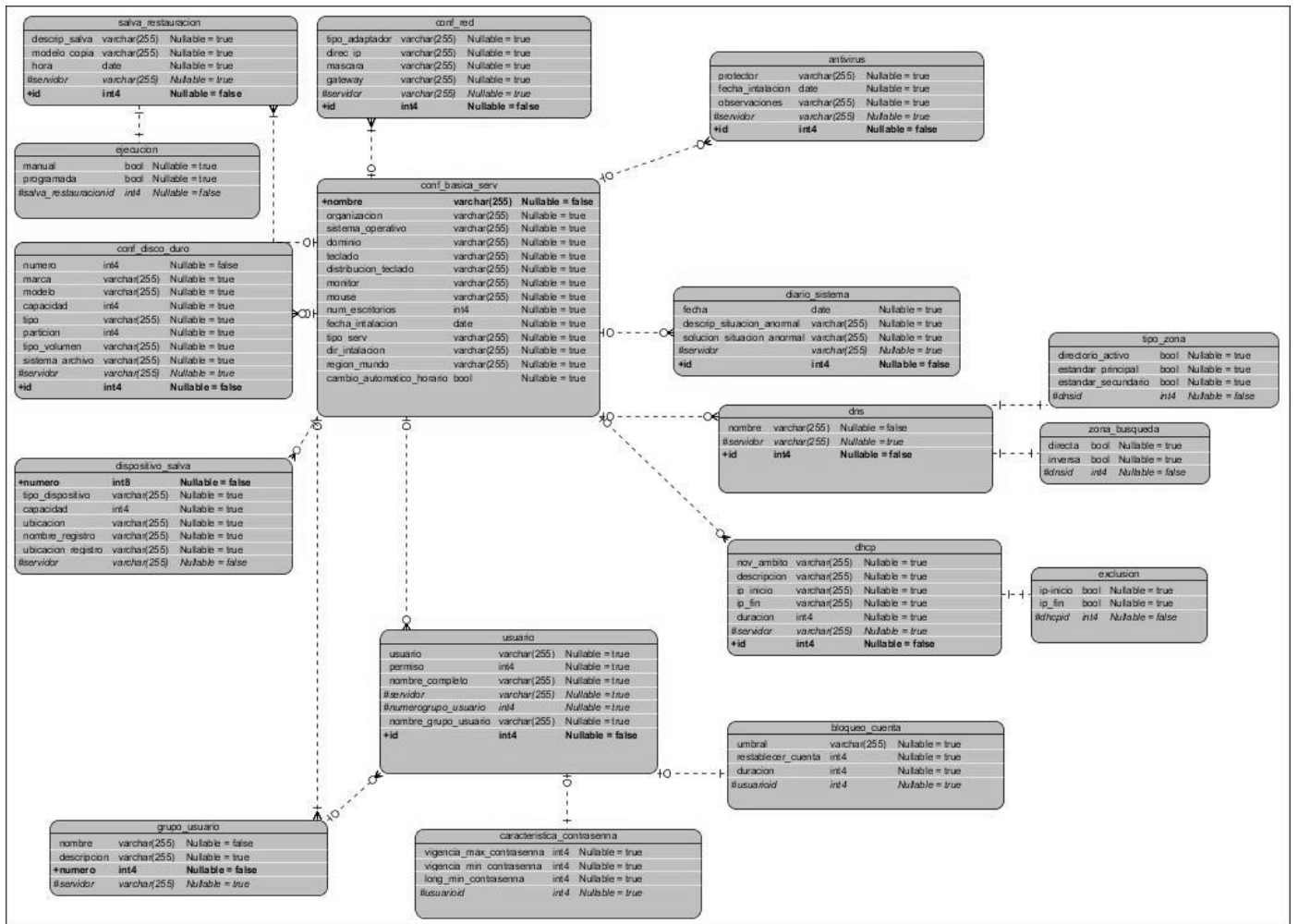


Fig. 21 Diagrama de Entidad-Relación.

3.4. Conclusiones Parciales

En este capítulo se realizó la descripción de las principales clases para facilitar la comprensión de la implementación, se observaron el empleo de algunos de los patrones de diseño y el patrón de arquitectura que se utiliza. Se desarrollaron los diagramas de de secuencia de cada uno de los requisitos funcionales, además de los diagramas de clases del diseño. También se desarrolló un diseño de la base de datos del sistema.

Capítulo 4: Implementación y prueba

Introducción

En el presente capítulo estará contenido lo referente a la implementación y prueba del sistema, en el mismo se hará el diagrama de componente, y el de despliegue, además del modelo de prueba donde se realizará la descripción de los casos de pruebas.

4.1. Modelo de Despliegue

Un diagrama de despliegue es la forma de mostrar la configuración de nodos de procesamiento en tiempo de ejecución y los componentes que en ellos residen. Estos nodos forman la topología de hardware sobre el que se ejecuta el sistema. Este diagrama se preocupa principalmente de la distribución, entrega e instalación de las partes que constituye el sistema físico. (21)

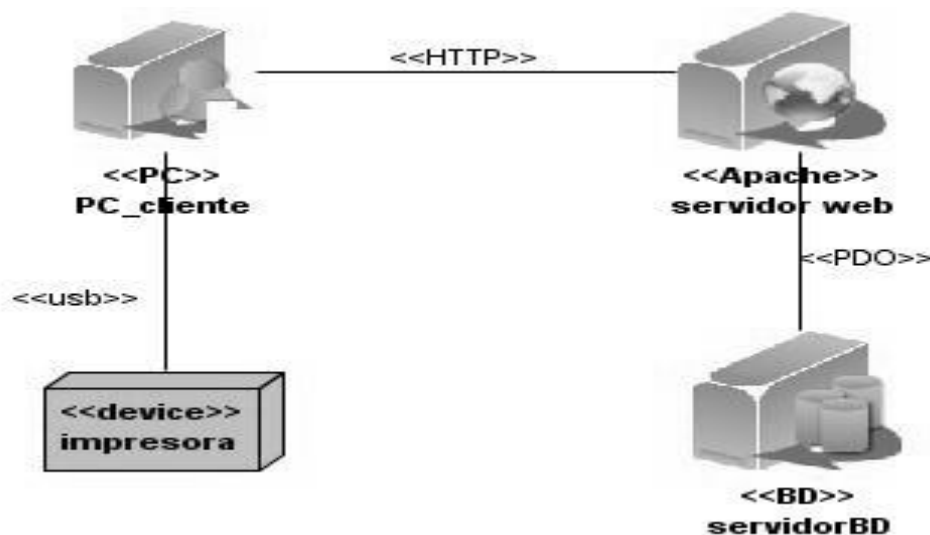


Fig. 22 Diagrama de Despliegue.

4.2. Diagrama de Componente

Un diagrama de componentes muestra las dependencias lógicas entre componentes software, sean éstos componentes fuentes, binarios o ejecutables, ilustran las piezas del software, controladores embebidos, etc. Los diagramas de Componentes prevalecen en el campo de la arquitectura de software pero pueden ser usados para modelar y documentar cualquier arquitectura de sistema, es decir para describir la vista de implementación estática de un sistema. Los diagramas de componentes se relacionan con los diagramas de clases, ya que un componente normalmente se corresponde con una o más clases, interfaces o colaboraciones pero un diagrama de Componentes tiene un nivel más alto de abstracción que un diagrama de clase, usualmente un componente se implementa por una o más clases (u objetos) en tiempo de ejecución. Estos son bloques de construcción, como eventualmente un componente puede comprender una gran porción de un sistema. (22)

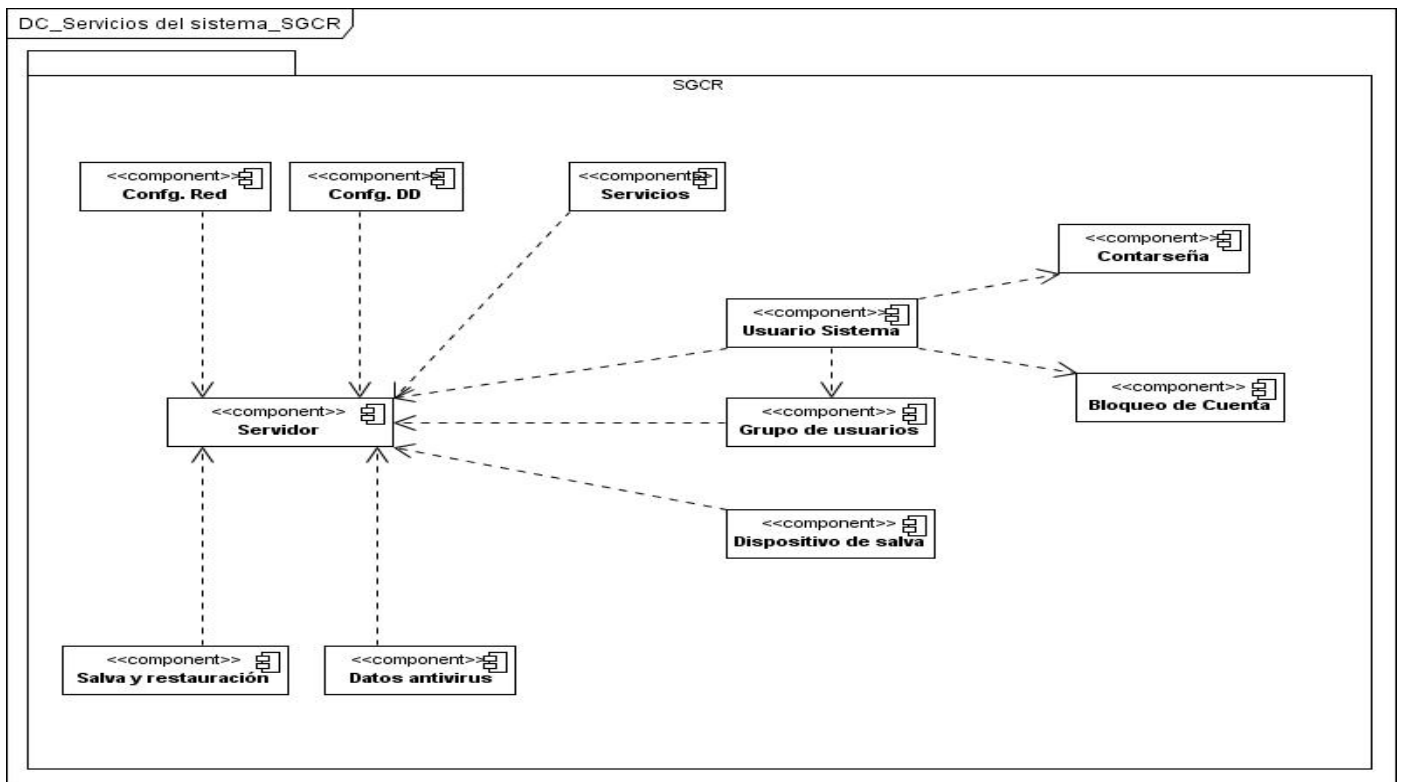


Fig. 23 Diagrama Componente del sistema

4.3. Matriz de Integración de componentes

La matriz de integración de componentes contiene todos los componentes definidos en el subsistema, de forma matricial, y en las intercepciones se especificarán los servicios que consume el componente en la vertical del horizontal. Existen dos matrices de integración de componentes, Interna y Externa.

Interna: se especifica la integración entre los componentes internos del sistema.

Componentes Internos	Componentes Internos			
	Servidores	Grupo usuarios	Contraseña	Bloqueo de cuenta
Conf. Red	Listar Servidores	-----	-----	-----
Servicios	Listar Servidores	-----	-----	-----
Usuario Sistema	Listar Servidores	Devolver Grupos	Devolver Contraseña	Devolver Bloqueo
Grupo Usuarios	Listar Servidores	-----	-----	-----
Dispositivos Salva	Listar Servidores	-----	-----	-----
Datos Antivirus	Listar Servidores	-----	-----	-----
Salva y Restauración	Listar Servidores	-----	-----	-----
Conf. DD	Listar Servidores	-----	-----	-----

4.4. Estándares de codificación

Un estándar de codificación completo comprende todos los aspectos de la generación de código. Si bien los programadores deben implementar un estándar de forma prudente, éste debe tender siempre a lo práctico. Un código fuente completo debe reflejar un estilo armonioso, como si un único programador hubiera escrito todo el código de una sola vez. (23)

4.4.1. Nomenclatura por tipo de clase

Clases controladoras: Las clases controladoras después del nombre se pone **Controller**.

Ejemplo: GestionarServidoresController

Clases modelos: Las clases modelos que están dentro de **Business** después del nombre se pone **Model**

Ejemplo: ConfBasicaServModel

Las clases que están dentro de **Domain** reciben el nombre de la tabla de la base de datos correspondiente.

Ejemplo: ConfBasicaServ

Las clases que se encuentran dentro de **Generated** el nombre comienza con el prefijo “Base” y seguido el nombre del dominio correspondiente.

Ejemplo: BaseConfBasicaServ

4.4.2. Nomenclatura de comentario en clases

Los comentarios en el sistema son claros y precisos los cuales explican una breve descripción de la clase a continuación se muestra un ejemplo.

Ejemplo: Comentario para clases

```
<?php
/**
 * Controlador para gestionar los Servidores
 *
 * @author Manuel Alejandro Piña Tamayo
 * @copyright UCID Cuba
 * @version 1.0
 */

class GestionarservidoresController extends ZendExt_Controller_Secure
{
    function init ()
    {
        parent::init ();
    }
}
```

Ejemplo: Comentario de funciones

```
//*****Adiciona un nuevo servidor*****
//*****Carga datos para el grid*****
//*****Cargar Servidores para el árbol*****
function cargarservidorAction()
{
    $nombres=ConfBasicaServ::nombreservers();
    $serv = array();
    foreach($nombres as $nombres){
        $servidores = array();
        $servidores['id']=$nombres['nombre'];
        $servidores['text']=$nombres['nombre'];
        $servidores['iconCls']='treeserv';
        $servidores['leaf']=true;
        array_push($serv,$servidores);
    }
    echo json_encode($serv);
}
```

Ejemplo: Declaración de Función

```
function cargardatosAction()
{
    $id=$_POST['id'];
    $datosg=ConfBasicaServ::getDatos($id);

    $datos=array();
    $datos[0]['nombre']=$datosg['nombre'];
    $datos[0]['organizacion']=$datosg['organizacion'];
    $datos[0]['tipo_serv']=$datosg['tipo_serv'];
    $datos[0]['sistema_operativo']=$datosg['sistema_operativo'];
    $datos[0]['dominio']=$datosg['dominio'];
    $datos[0]['teclado']=$datosg['teclado'];
    $datos[0]['distribucion_teclado']=$datosg['distribucion_teclado'];
    $datos[0]['monitor'] = $datosg['monitor'];
    $datos[0]['mouse'] = $datosg['mouse'];
    $datos[0]['num_escritorios'] = $datosg['num_escritorios'];
    $datos[0]['fecha_intalacion'] = $datosg['fecha_intalacion'];
    $datos[0]['dir_intalacion'] = $datosg['dir_intalacion'];
    $datos[0]['region_mundo'] = $datosg['region_mundo'];
    $datos[0]['cambio_automatiko_horario'] = $datosg['cambio_automatiko_horario'];

    echo json_encode(array(cont=>count($datos),data=>$datos));
}
```

4.5. Prueba

En la actualidad existen varios tipos de pruebas de software, destinadas para encontrar el mayor número de errores, para ofrecer un producto de óptima calidad. Dentro de los principales tipos de pruebas se pueden encontrar las pruebas de caja negra y las pruebas de caja blanca.

Las **pruebas de caja blanca** se basan en el minucioso examen de los detalles procedimentales. Se comprueban los caminos lógicos del software proponiendo casos de prueba que ejerciten conjuntos específicos de condiciones y/o bucles.

Las **pruebas de caja negra** se refieren a las pruebas que se llevan a cabo sobre la interfaz del software. Pretenden demostrar que las funciones del software son operativas, que la entrada se acepta de forma adecuada y que se produce una salida correcta, así como que la integridad de la información externa se mantiene. Una prueba de caja negra examina algunos aspectos del modelo fundamental del sistema sin tener mucho en cuenta la estructura lógica interna del software.

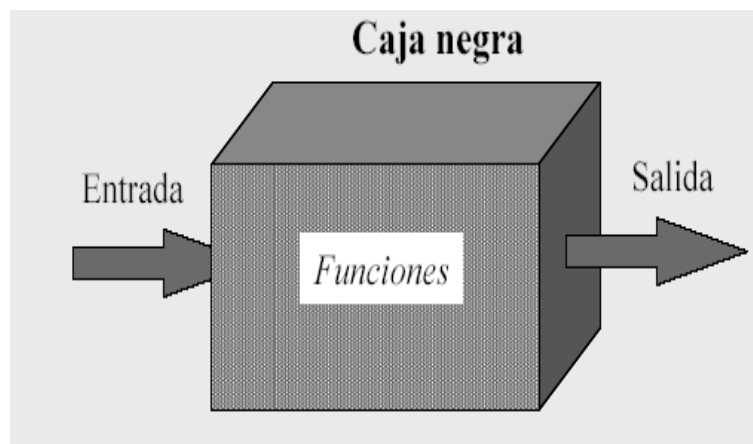


Fig. 24 Representación de la Prueba de Caja Negra.

Estas pruebas permiten obtener un conjunto de condiciones de entrada que ejerciten completamente todos los requisitos funcionales de un programa. En ellas se ignora la estructura de control, concentrándose en los requisitos funcionales del sistema y ejercitándolos.

A continuación se muestra el diseño de caso de prueba del Requisito **Gestionar configuración de la red**.

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
Gestionar Configuración de la red	Permitir que el administrador pueda insertar los datos relacionados con la configuración de la red.	EP 1.1: Adicionar configuración de la red correctamente.	<ul style="list-style-type: none">-Se da clic en el botón “Inicio”, se selecciona la opción de “Redes”, después se escoge la opción de “Gestión y control” y se da clic en la opción de “Gestionar redes”.-Se selecciona el servidor que se quiera adicionar la red.-Se selecciona en la interfaz la opción de “Adicionar”.-El sistema muestra los campos a llenar.-El usuario llena los datos correctamente y presiona el botón Insertar.-El sistema muestra un mensaje indicando que la operación se realizó de manera correcta.

		<p>EP 1.2: Adicionar configuración de la red con datos incorrectos.</p>	<p>-Se da clic en el botón "Inicio", se selecciona la opción de "Redes", después se escoge la opción de "Gestión y control" y se da clic en la opción de "Gestionar redes".</p> <p>-Se selecciona el servidor que se quiera adicionar la red.</p> <p>-Se selecciona en la interfaz la opción de "Adicionar".</p> <p>-El sistema muestra los campos a llenar.</p> <p>-El usuario llena los campos con datos incorrectos y/o deja campos obligatorios vacíos.</p> <p>-El sistema marca los datos incorrectos y/o vacíos.</p> <p>-El usuario presiona el botón Aceptar.</p> <p>-El sistema no permite insertar la configuración de la red.</p>
--	--	---	---

		<p>EP 1.3: Cancelar operaciones.</p>	<p>-Se da clic en el botón “Inicio”, se selecciona la opción de “Redes”, después se escoge la opción de “Gestión y control” y se da clic en la opción de “Gestionar redes”.</p> <p>-Se selecciona en la interfaz la opción de “Adicionar”.</p> <p>-El sistema muestra los campos a llenar.</p> <p>-El usuario llena o no los campos y presiona el botón “Cancelar”.</p>
--	--	--------------------------------------	---

Escenario Modificar configuración de la red

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
<p>Gestionar Configuración de la red</p>	<p>Permitir que el administrador pueda insertar, modificar, buscar y eliminar los datos relacionados con la configuración de la red.</p>	<p>EP 1.1: Modificar configuración de la red correctamente</p>	<p>-Se da clic en el botón “Inicio”, se selecciona la opción de “Redes”, después se escoge la opción de “Gestión y control” y se da clic en la opción de “Gestionar redes”.</p> <p>-Se selecciona el servidor que se quiera modificar la red.</p> <p>-Se selecciona en la interfaz la</p>

			<p>opción de “Modificar”.</p> <p>-El sistema muestra los campos a modificar.</p> <p>-El usuario modifica los datos correctamente y presiona el botón Modificar.</p> <p>-El sistema muestra un mensaje indicando que la operación se realizó de manera correcta.</p>
--	--	--	---

		<p>EP 1.2: Modificar configuración de la red con datos incorrectos.</p>	<p>Se da clic en el botón "Inicio", se selecciona la opción de "Redes", después se escoge la opción de "Gestión y control" y se da clic en la opción de "Gestionar redes".</p> <p>Se selecciona en la interfaz la opción de "Modificar".</p> <p>El sistema muestra los campos a modificar.</p> <p>El usuario modifica los campos con datos incorrectos y/o deja campos obligatorios vacíos.</p> <p>El sistema marca en rojo los datos incorrectos y/o vacíos.</p> <p>El usuario presiona el botón Modificar.</p> <p>El sistema no permite modificar la red.</p>
--	--	---	---

		<p>EP 1.3: Cancelar operación.</p>	<p>-Se da clic en el botón "Inicio", se selecciona la opción de "Redes", después se escoge la opción de "Gestión y control" y se da clic en la opción de "Gestionar redes".</p> <p>-Se selecciona en la interfaz la opción de "Modificar".</p> <p>-El sistema muestra los campos a modificar.</p> <p>-El usuario modifica los datos y presiona el botón Cancelar</p>
--	--	------------------------------------	--

Escenario Eliminar configuración de la red

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
----------------------	---------------------	-----------------------	---------------------

Gestionar Configuración de la red	Permitir que el administrador elimine los datos relacionados con la configuración de la red.	EP 1.1: Eliminar Configuración de la red.	<p>-Se da clic en el botón "Inicio", se selecciona la opción de "Redes", después se escoge la opción de "Gestión y control" y se da clic en la opción de "Gestionar redes".</p> <p>-Se selecciona el servidor donde se encuentre la configuración de la red siempre y cuando este insertado.</p> <p>-Se escoge la configuración que se desea eliminar.</p> <p>-Se presiona botón Eliminar.</p> <p>-El sistema muestra un mensaje de confirmación.</p> <p>-El usuario presiona el botón Aceptar del mensaje.</p> <p>El sistema muestra un mensaje de confirmación indicando que la operación se realizó de manera correcta.</p>
-----------------------------------	--	---	--

		EP 1.2: Cancelar operaciones.	-Se selecciona el servidor a eliminar. -El sistema muestra un mensaje de confirmación. -El usuario presiona el botón Cancelar del mensaje.
--	--	-------------------------------	---

Juegos de datos a probar

Id del escenario	Escenario	Tipo de adaptador	Máscara	Dirección IP	Gate way	Respuesta del sistema	Resultado de la prueba
EP 1.1	Eliminar Configuración de la red.	NA	NA	NA	NA	El sistema muestra el siguiente mensaje de confirmación: ¿Está seguro que desea eliminar la Red seleccionada?	
EP 1.2	Cancelar operaciones.	NA	NA	NA	NA	El sistema muestra el siguiente mensaje de confirmación: ¿Está seguro que desea eliminar la Red seleccionada? El sistema cierra la interfaz.	

Las celdas de la tabla contienen V, I, o NA. V indica válido, I indica inválido, y NA se debe proporcionar un valor del dato en este caso.

4.6. Conclusiones Parciales

En el presente capítulo se realizaron los diagramas de componentes y el diagrama de despliegue. En el diagrama de despliegue se determinaron las relaciones existentes entre el sistema y el hardware donde se implantará la aplicación. Por último se determinó el tipo de prueba a utilizar y se realizó los casos de pruebas a los requisitos seleccionados.

Conclusiones Generales

A continuación se presentarán las conclusiones a las cuales se ha arribado a través de la realización del presente trabajo con el cual se dio cumplimiento al objetivo planteado:

- Se realizó un estudio de los principales sistemas de gestión y control de la red existentes en Cuba y el Mundo
- Se siguieron una serie de pasos definidos por los estándares de diseño y los pasos propuestos por la metodología “Proceso de Desarrollo y Gestión de Proyectos de Software”, usando el Visual Paradigm como herramienta para generar los artefactos propuesto por esta metodología y UML como lenguaje de modelado.
- Se obtuvo el modelo del negocio del sistema lo cual permitió entender las necesidades del cliente.
- Se realizó el levantamiento de requisitos funcionales y no funcionales, a partir de los cuales se definieron y describieron cada una de las funcionalidades que posee el sistema.
- Para brindar una adecuada seguridad en el almacenamiento de la información se realizó el diseño la base de datos.
- Finalmente se realizaron pruebas al sistema con el objetivo de verificar la eficacia de las funcionalidades del mismo y dar solución a inconvenientes encontrados.

Recomendaciones

Para dar continuidad y seguimiento al presente trabajo de diploma se realizan las siguientes recomendaciones:

- Que se le de continuidad a este trabajo para lograr un versión superior con muchas más funcionalidades.
- El centro conserve el documento como consulta y guía para aquellos que darán mantenimiento al sistema.
- Que se implementen un conjunto de script que permitan la actualización automática del sistema.
- Que se realice un nuevo diseño de Interfaz de Usuario con un mejor acoplamiento, manejabilidad y facilidad de uso para los usuarios del sistema.

Referencia Bibliográfica

1. **Yannelys V. Jerez Naranjo, Sergio Tarrago Oliver.** *Sistema de Certificación de conocimientos. Una opción para los administradores de red.* 2008.
2. **UPCommon, Cataluniawww.upcommons.upc.edu.** upcommons. *UPCommon.* [En línea] 2006. www.upcommons.upc.edu.
3. **Informatica, Revista.** *larevistainformatica.* [En línea] 2006. <http://www.larevistainformatica.com/red-informatica.htm>.
4. **Alicante, Universidad.** *cervantes.cpd.* [En línea] abril de 2007. <http://cervantes.cpd.ua.es/es/servicios/si/documentacion/presentaciones/monitorizacion.pdf>.
5. **Pandora.** *PandoraFMS.* [En línea] 2008. <http://www.pandorafms.org/index.php?sec=project&sec2=home&lng=en.%20pandorafms.org>.
6. **REM-Q.** *www.remq.edu.ec.* [En línea] 2010. http://www.remq.edu.ec/index.php?option=com_content&view=article&id=114:sistema-de-inventarios-de-la-red-educativa-metropolitana&catid=54:remq.
7. **Veo.** *veo.com.* [En línea] 2010. <http://www.veo.com.mx/>.
8. **Cyta.** *cyta.com.* [En línea] 2005. <http://www.cyta.com.ar/biblioteca/bddoc/bdlibros/proyectoinformatico/libro/c5/c5.htm>.
9. **Luis Giraldo, Yuliana Zapata.** *hugolopez.* [En línea] 24 de septiembre de 2005. http://hugolopez.phi.com.co/docs/download/file=Giraldo-Zapata-Herramientas%20de%20ISW.pdf,_id=17.
10. **Lanzillotta, Analía.** *MasterMaganize.* [En línea] 2004. <http://www.mastermagazine.info/termino/5560.php>.
11. **Valdés, Damián Pérez.** *maestrosdelweb.* [En línea] 2 de noviembre de 2007. <http://www.maestrosdelweb.com/principiantes/los-diferentes-lenguajes-de-programacion-para-la-web/>.
12. **error500.** *error500.net.* [En línea] 1 de noviembre de 2004. http://www.error500.net/garbagecollector/archives/categorias/bases_de_datos/sistema_gestor_de_base_de_datos_sgbd.php. error500.
13. **PostgreSQL.** *soporte.tiendalinux.* [En línea] 31 de mayo de 2005. http://soporte.tiendalinux.com/portal/Portfolio/postgresql_ventajas_html. soporte.tiendalinux.
14. **Valdes, Damian Perez.** *maestrosdelweb.* [En línea] 31 de enero de 2007. <http://www.maestrosdelweb.com/editorial/editores-web-que-facilitan-tu-trabajo/>. maestrosdelweb.
15. **Wordpress.** *vargasti.* [En línea] 8 de agosto de 2006. <http://vargasti.wordpress.com/2007/08/06/libreria-extjs/>.
16. **Doctrine.** *doctrine-project.* [En línea] 2007. <http://www.doctrine-project.org/>. doctrine-projec.
17. **Alexa.** *bds.* [En línea] junio de 2009. <http://www.proyectosbds.com/software-y-programacion-a-medida/programacion-php-lamp-zf/mas-sobre-zend-framework/121/>.
18. **linux-ciberaula.** *ciberaula.* [En línea] 2006. http://linux.ciberaula.com/articulo/linux_apache_intro.
19. **mastermagazine.** *mastermagazine.com.* [En línea] 2004. <http://www.mastemagazine.info/termino/>.
20. **Guerrero, Luis A.** *face.ubiobio.* [En línea] 2007. www.face.ubiobio.cl/~cguatierr/aydoo.ppt.

21. **COSSIO TABOADA MIGUEL HUMBERTO, HUANCA APAZA FLOREN OVIDIO.** virtual.usalesiana. [En línea] 2009. virtual.usalesiana.edu.bo/web/practica/archiv/ddespli2.doc.
22. **Ramírez, Elisa Arizaca.** virtual.usalesiana.edu.bo. *web docente*. [En línea] 9 de junio de 2009. virtual.usalesiana.edu.bo/web/practica/archiv/compon.doc.
23. **Fernández, Giovanni.** www.elguille.info. *elguille.info*. [En línea] 21 de abril de 2005. http://www.elguille.info/colabora/NET2005/giovannyfernandez_EstandarCodificacionNET.htm.

Bibliografía

- Yannelys V. Jerez Naranjo, Sergio Tarrago Oliver.** *Sistema de Certificación de conocimientos. Una opción para los administradores de red.* 2008.
- UPCommon, Cataluniawww.upcommons.upc.edu.** upcommons. *UPCommon.* [En línea] 2006. www.upcommons.upc.edu.
- Informatica, Revista.** *larevistainformatica.* [En línea] 2006. <http://www.larevistainformatica.com/red-informatica.htm>.
- Alicante, Universidad.** *cervantes.cpd.* [En línea] abril de 2007. <http://cervantes.cpd.ua.es/es/servicios/si/documentacion/presentaciones/monitorizacion.pdf>.
- Pandora.** *PandoraFMS.* [En línea] 2008. <http://www.pandorafms.org/index.php?sec=project&sec2=home&lng=en.%20pandorafms.org>.
- REMQ.** *www.remq.edu.ec.* [En línea] 2010. http://www.remq.edu.ec/index.php?option=com_content&view=article&id=114:sistema-de-inventarios-de-la-red-educativa-metropolitana&catid=54:remq.
- Veo.** *veo.com.* [En línea] 2010. <http://www.veo.com.mx/>.
- Cyta.** *cyta.com.* [En línea] 2005. <http://www.cyta.com.ar/biblioteca/bddoc/bdlibros/proyectoinformatico/libro/c5/c5.htm>.
- Luis Giraldo, Yuliana Zapata.** *hugolopez.* [En línea] 24 de septiembre de 2005. http://hugolopez.phi.com.co/docs/download/file=Giraldo-Zapata-Herramientas%20de%20ISW.pdf,_id=17.
- Lanzillotta, Analía.** *MasterMaganize.* [En línea] 2004. <http://www.mastermagazine.info/termino/5560.php>.
- Valdés, Damián Pérez.** *maestrosdelweb.* [En línea] 2 de noviembre de 2007. <http://www.maestrosdelweb.com/principiantes/los-diferentes-lenguajes-de-programacion-para-la-web/>.
- error500.** *error500.net.* [En línea] 1 de noviembre de 2004. http://www.error500.net/garbagecollector/archives/categorias/bases_de_datos/sistema_gestor_de_base_de_datos_sgbd.php. error500.
- PostgreSQL.** *soporte.tiendalinux.* [En línea] 31 de mayo de 2005. http://soporte.tiendalinux.com/portal/Portfolio/postgresql_ventajas_html. soporte.tiendalinux.
- Valdes, Damian Perez.** *maestrosdelweb.* [En línea] 31 de enero de 2007. <http://www.maestrosdelweb.com/editorial/editores-web-que-facilitan-tu-trabajo/>. maestrosdelweb.
- WordPress.** *vargasti.* [En línea] 8 de agosto de 2006. <http://vargasti.wordpress.com/2007/08/06/libreria-extjs/>.
- Doctrine.** *doctrine-project.* [En línea] 2007. <http://www.doctrine-project.org/>. doctrine-projec.
- Alexa.** *bds.* [En línea] junio de 2009. <http://www.proyectosbds.com/software-y-programacion-a-medida/programacion-php-lamp-zf/mas-sobre-zend-framework/121/>.
- linux-ciberaula.** *ciberaula.* [En línea] 2006. http://linux.ciberaula.com/articulo/linux_apache_intro.
- mastermagazine.** *mastermagazine.com.* [En línea] 2004. <http://www.mastemagazine.info/termino/>.
- Guerrero, Luis A.** *face.ubiobio.* [En línea] 2007. www.face.ubiobio.cl/~cguierr/aydoo.ppt.
- COSSIO TABOADA MIGUEL HUMBERTO, HUANCA APAZA FLOREN OVIDIO.** *virtual.usalesiana.* [En línea] 2009. virtual.usalesiana.edu.bo/web/practica/archiv/ddespli2.doc.

Ramírez, Elisa Arizaca. virtual.usalesiana.edu.bo. *web docente*. [En línea] 9 de junio de 2009.
virtual.usalesiana.edu.bo/web/practica/archiv/compon.doc.

Fernández, Giovanni. www.elguille.info. *elguille.info*. [En línea] 21 de abril de 2005.
http://www.elguille.info/colabora/NET2005/giovannyfernandez_EstandarCodificacionNET.htm.

Anexos






Anexo1: Modelo del negocio**Descripción de los procesos de negocio**

P1: Realizar gestión de la red

Objetivos:	Establecer los requisitos expuesto en el manual de seguridad para el control de la red
Evento(s) que lo generan	La necesidad de llevar la gestión de la red más segura y organizada posible.
Precondiciones	Se cuenta con un documento llamado Manual de seguridad informática, donde se lleva la información se lleva de forma manual.
Post condiciones	Se crean las plantillas relacionadas con el documento y se llenan los datos de las mismas.
Clientes internos	Administrador
Clientes externos	MINFAR
Entradas	Documento de manual de seguridad.
Salidas	Plantillas.

Anexo2: Requisitos Funcionales

Especificación del requisito Gestionar Configuración del servidor

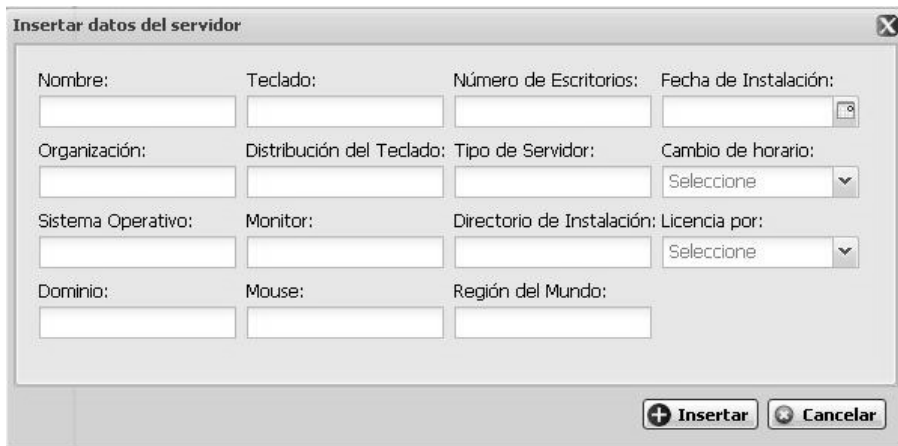
Conceptos tratados	Conceptos	Atributos
		Nombre, teclado, número de Escritorio, fecha de instalación, organización, distribución de teclado, tipo de servidor, cambio horario, sistema operativo, monitor, directorio de instalación, licencia por, dominio, mouse, región del mundo.
Precondiciones	Precondiciones	Pre-requisito
	El usuario debe estar autenticado como administrador.	No procede.
Descripción	<p>1. Se da clic en el botón “Inicio”, se selecciona la opción de “Redes”, después se escoge la opción de “Gestión y control” y se da clic en la opción de “Gestionar servidores”.</p>  <p>2. Se muestra la interfaz correspondiente con las siguientes opciones:</p>  <ul style="list-style-type: none">  Adicionar.  Modificar.  Eliminar. 	

✚ Buscar.

3. Se selecciona la opción que se desea realizar.

4. Si se selecciona la opción de “**Adicionar**”.

4.1. Se muestra una interfaz con los campos correspondientes que se deben llenar.



The screenshot shows a dialog box titled "Insertar datos del servidor" with a close button (X) in the top right corner. The dialog contains the following fields and controls:

Nombre:	Teclado:	Número de Escritorios:	Fecha de Instalación:
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Organización:	Distribución del Teclado:	Tipo de Servidor:	Cambio de horario:
<input type="text"/>	<input type="text"/>	<input type="text"/>	Seleccione
Sistema Operativo:	Monitor:	Directorio de Instalación:	Licencia por:
<input type="text"/>	<input type="text"/>	<input type="text"/>	Seleccione
Dominio:	Mouse:	Región del Mundo:	
<input type="text"/>	<input type="text"/>	<input type="text"/>	

At the bottom right of the dialog, there are two buttons: "+ Insertar" and "Cancelar".

4.2. Se llenan los campos correspondientes a la opción seleccionada.


4.3. Se verifican que los campos estén correctamente insertados.

4.4. Se presiona el botón Insertar y se muestra un mensaje: “Se insertaron los datos correctamente”.

5. Si se selecciona la opción de “**Modificar**”.

5.1. Se selecciona el servidor que se quiera modificar los datos.

5.2. Se muestra una interfaz con todos los campos correspondientes.

	 <p>5.3. Se presiona el botón modificar y se muestra un mensaje: "El elemento fue modificado".</p> <p>6. Si se selecciona la opción de " Eliminar "</p> <p>6.1. Se escoge el servidor siempre y cuando este insertado.</p> <p>6.2. Se presiona el botón eliminar y se muestra un mensaje: "El elemento fue eliminado satisfactoriamente".</p> <p>7. Si se selecciona la opción de " Buscar ".</p> <p>7.1. Se muestra una interfaz donde se pone el nombre del servidor que se desea buscar.</p>
Validaciones	Los campos deben estar correctamente llenos.
Post-condiciones	Queda Gestionado la configuración del servidor.
Post-requisito	No procede.

Anexo 4: Ejemplo de descripción de las clases.

Tabla 1: Descripción de la clase controladora GestionarServidoresController

Nombre: GestionarServidoresController
Tipo de clase: controladora

Atributo	Tipo
No procede	No procede
Para cada responsabilidad:	
Nombre:	cargarservidorAction
Descripción:	Devuelve una lista con el nombre de todos los servidores.
Nombre:	addAction
Descripción:	Se encarga de recoger los datos del formulario y acceder al negocio para insertar un servidor.
Nombre:	modAction
Descripción:	Recoge los datos que se desean modificar, accede al negocio enviándole dichos datos para que sean modificados.
Nombre:	elimAction
Descripción:	Accede al negocio enviándole el id del servidor seleccionado para que sea eliminado.
Nombre:	cargardatosAction
Descripción:	Hace una búsqueda del servidor seleccionado y recoge los datos de la misma.

Tabla 2: Descripción de la clase modelo ConfBasicaServModel

Nombre: ConfBasicaServModel	
Tipo de clase: modelo	
Atributo	Tipo
No procede	No procede
Para cada responsabilidad:	
Nombre:	Insertar

Descripción:	Inserta los datos del servidor que le envía la controladora a la base de datos
Nombre:	Actualizar
Descripción:	Actualiza los datos del servidor seleccionado en el base de datos
Nombre:	Eliminar
Descripción:	Elimina los datos del servidor seleccionado de la base de datos

Tabla 3: Descripción de la clase ConfBasicaServ

Nombre: ConfBasicaServ	
Tipo de clase: modelo	
Atributo	Tipo
No procede	No procede
Para cada responsabilidad:	
Nombre:	GetTodos
Descripción:	Devuelve una lista con todos los servidores que hay en la tabla de la base de datos
Nombre:	getDatos
Descripción:	Devuelve los datos de un servidor seleccionado
Nombre:	nombreservers
Descripción:	Devuelve una lista con todos los nombre del servidor que hay en la tabla

Tabla 4: Descripción de la clase Gestionarservidores.js

Nombre Gestionarservidores.js
Tipo de clase: Vista

Atributo	Tipo
No procede	No procede
Para cada responsabilidad:	
Nombre:	No procede
Descripción:	No procede

Glosario de Términos

UML: “Unified Modeling Language”. Lenguaje gráfico que brinda un vocabulario y reglas para especificar, construir, visualizar y documentar los artefactos de un sistema utilizando el enfoque orientado a objetos.

Visual Paradigm: Herramienta CASE (Computer-Aided Software Engineering) que permite realizar ingeniería tanto directa como inversa. Utiliza UML como lenguaje de modelado.

SGCR: Sistema de gestión y control de redes.

RF: Requisito Funcional.

RNF: Requisito No Funcional.

FMS: Sistema de monitorización flexible.

PHP: Es acrónimo de Hypertext Pre-processor. Es un lenguaje de programación del lado del servidor gratuito e independiente de plataforma, rápido, con una gran librería de funciones y mucha documentación (Personal Home Page).

HTML: Es un lenguaje de marcas diseñado para estructurar textos y presentarlos en forma de hipertexto, que es el estándar de las páginas Web. (Hyper Text Markup).

Apache: Proyecto nacido para crear un servidor Web estable, fiable y veloz para plataformas Unix.