



INFORMÁTICA

Universidad de Cienfuegos

“Carlos Rafael Rodríguez”

Facultad de Ingeniería Informática

**SISTEMA INFORMATICO PARA EL MONITOREO DE HARDWARE EN
COMPUTADORAS CON WINDOWS.**



Trabajo de Diploma para optar por el Título de Ingeniería en Informática.

Autores:

Yoanny Domínguez Carrero

Yasmany Rodríguez López

Tutores:

Ing. Mario Merino Escoto

Lic. Alexey Zamora Ferriol

Consultante:

Ing. Rewer Canosa

Cienfuegos 2008

“Año 50 de la Revolución”

La inteligencia consiste no sólo en el conocimiento, sino también en la destreza de aplicar los conocimientos en la práctica.

Aristóteles.

Declaración de autoría

Declaramos que somos los únicos autores del trabajo de diploma titulado Sistema Informático para el monitoreo de hardware en computadoras con Windows, y autorizamos al Departamento de Informática de la Facultad de Informática en la Universidad de Cienfuegos “Carlos Rafael Rodríguez”, para que hagan el uso que estimen pertinente con el trabajo de diploma.

Para que así conste firmamos la presente a los 12 días del mes de Junio del 2008.

Firma del Autor

Yoanny Domínguez Carrero

Firma del Autor

Yasmany Rodríguez López

Firma del Tutor

Ing. Mario Merino Escoto

Firma del Tutor

Lic. Alexey Zamora Ferriol

Firma ICT

Opinión del Usuario.

El Trabajo de Diploma, titulado Sistema Informático para el monitoreo de hardware en computadoras con Windows, fue realizado en la Facultad de Informática de la Universidad de Cienfuegos. Se considera que, en correspondencia con los objetivos trazados, el trabajo realizado nos satisface:

Totalmente

Parcialmente en un %

Los resultados del presente Trabajo de Diploma le reportan a nuestra entidad los beneficios siguientes:

Como resultado de la implantación de este trabajo se reporta un efecto económico que asciende a _____ MN y/o _____ CUC.

Y para que así conste, se firma la presente a los 12 días del mes de Junio del año 2008.

Firma: _____

Ing. Rewer Canosa Reyes

Cuño

Facultad de Informática de la Universidad
de Cienfuegos.

Opinión del Tutor

Trabajo de Diploma: Sistema Informático para el monitoreo de hardware en computadoras con Windows.

Autores: Yoanny Domínguez Carrero y Yasmany Rodríguez López.

Los tutores del presente Trabajo de Diploma consideramos que durante su ejecución los estudiantes mostraron las cualidades que a continuación se detallan.

El trabajo realizado por los estudiantes Yoanny Domínguez Carrero y Yasmany Rodríguez López consistió en desarrollar una herramienta para monitorear el hardware de las computadoras con Windows en la red, alertar de posibles cambios y generar los reportes necesarios. La tesis presentada está dividida en 4 capítulos: Fundamentación Teórica, Modelo del Dominio, Modelo del Sistema, Estudio de Factibilidad, además de Conclusiones, Recomendaciones, Bibliografía, Referencias Bibliográficas y Anexos. Es de destacar, todo el trabajo constante de investigación desarrollado por estos estudiantes sobre los diferentes softwares existentes en el mundo y la construcción del producto informático propuesto, trayendo consigo importantes beneficios para nuestra entidad.

Por tales motivos consideramos que los estudiantes Yoanny Domínguez Carrero y Yasmany Rodríguez López están aptos para ejercer como Ingenieros Informáticos; y propongo que se le otorgue al Trabajo de Diploma la calificación de 5 puntos.

Y para que así conste, se firma la presente a los 12 días del mes de Junio del año 2008.

Firma del Tutor

Ing. Mario Merino Escoto

Firma del Tutor

Lic. Alexey Zamora Ferriol

Agradecimientos

Quisiéramos agradecer a todas las personas que de una forma u otra han contribuido con su ayuda y sin la cual no hubiese sido posible la realización de este trabajo, en especial a:

- *Alexey Zamora Ferriol y Mario Merino Escoto tutores de esta investigación, por su paciencia, preocupación y tiempo que nos han dedicado.*
- *A los compañeros de grupo por estos cinco años de estudios compartidos.*
- *A los compañeros de cuarto.*
- *A todos los profesores que han contribuido en estos cinco años a nuestra formación como profesionales.*

A todos, MUCHAS GRACIAS.

Agradezco:

- ✓ *A mi mamá por su amor, su confianza en mí y su preocupación constante por mis estudios.*
- ✓ *A mi papá por su preocupación en mí y en mis estudios.*
- ✓ *A mi abuela tuta por su dedicación y cariño.*
- ✓ *A mi hermana por apoyarme.*
- ✓ *Al resto de mi familia por preocuparse.*

Yasmany

Agradezco:

- ✓ *A mi mamá y mi papá por su amor, comprensión y esfuerzos para que todos mis estudios salieran adelante.*
- ✓ *A Denaysis por estar a mi lado en todos los buenos y los malos momentos de la vida.*
- ✓ *A mi abuelo Billiqui que tanto se preocupa porque yo llegue lejos y lucha por eso.*
- ✓ *A mis suegros que tanto me alientan en seguir adelante.*
- ✓ *Al resto de mi familia que tanto batallan y se preocupan por mí.*
- ✓ *Todos mis amigos que me quieren y me respetan.*
- ✓ *A todos, todos los cuales de alguna manera u otra han hecho que yo haya llegado hasta aquí y logre cumplir mis sueños.*

Muchas Gracias...

Dedicatoria

*El presente trabajo está
dedicado a nuestra familia.*

Resumen

Producto de la ausencia de un software que lleve el control del hardware de las computadoras en una red. Se examinaron algunos de los software existentes para fundamentar la creación de uno con nuevas características.

Se llevaron a cabo las etapas del proceso de desarrollo del software propuesto, según lo que especifica la metodología de Proceso Unificado de Desarrollo, obteniéndose como producto final el software titulado “Sistema informático para el monitoreo de hardware en computadoras con Windows”, encargado de recolectar información del hardware de las computadoras en una red de área local y almacenarla en una base de datos, brindando la posibilidad de detectar y alertar de cambios ocurridos en el mismo.

Introducción.....	14
Capítulo I – Fundamentación Teórica.....	17
1.1– Introducción.....	17
1.2– Descripción del dominio del problema.....	17
1.2.1– ¿Qué es una Red LAN?	17
1.2.2– Topología.....	17
1.2.3– Topología de Bus.....	17
1.2.4– Topología de Anillo.....	18
1.2.5– Topología de Estrella.....	18
1.2.6– Topologías Híbridas de LANs.....	19
1.3 – Descripción del objeto de estudio.....	19
1.3.1 - Objetivos estratégicos de la organización.....	19
1.3.2 - Flujo actual de los procesos y análisis crítico de la ejecución de estos.....	20
1.4 – Descripción de los sistemas existentes.....	21
1.4.1–Everest Home Edition.....	21
1.4.2–Everest Corporate Edition.....	21
1.4.3–Total Network Inventory.....	21
1.4.4– LOGINventory.....	21
1.4.5– AjpdSoft Inventario PCs.....	22
1.4.6–OCS Inventory NG.....	22
1.4.7– Otras herramientas.....	22
1.4.8– El sistema propuesto.....	23
1.5 – Tendencias, metodologías y/o tecnologías actuales.....	24
1.5.1 La Metodología RUP y el lenguaje UML.....	24
1.5.4 Lenguajes de Programación. Herramientas visuales de desarrollo.....	26
1.5.6 Sistemas de Gestión de Bases de Datos.....	28
1.6 – Conclusiones.....	30
Capítulo II – Modelo del Dominio.....	31
2.1 Introducción.....	31
2.2 Descripción del modelo del dominio.....	31
2.2.1 Modelo de objetos del dominio.....	32
2.3 Reglas del negocio a considerar.....	32
2.4 Conclusiones.....	33
Capítulo III – Modelo del sistema.....	34
3.1 – Introducción.....	34
3.2 – Descripción del modelo de sistema.....	34
3.3 – Modelación del modelo de sistema.....	34
3.3.1 – Requerimientos funcionales.....	35
3.3.2 – Requerimientos no funcionales.....	39
3.3.3 – Actores del modelo de sistema.....	43
3.3.4 – Diagramas de casos de uso del sistema.....	43
3.3.6 – Descripción de los casos de usos del sistema.....	44
3.4– Construcción del sistema.....	53
3.4.1– Diagrama de clases del diseño.....	53
3.4.2– Diagrama del modelo lógico de datos.....	55
3.4.3– Diagrama del modelo físico de datos.....	56
3.4.4– Diagramas de implementación.....	57

3.5— Principios de diseño del sistema.....	59
3.5.1— Diseño de la interfaz.....	59
3.5.2— Concepción general de la ayuda.....	60
3.5.3— Tratamiento de errores.....	60
3.5.4— Estándares de codificación.....	60
3.6 – Conclusiones.....	61
Capítulo IV — Estudio de Factibilidad.....	62
4.1 — Planificación.....	62
4.2 — Costos.....	66
4.3 — Análisis de costos y beneficios.....	69
4.4 — Conclusiones.....	70
Conclusiones.....	71
Recomendaciones.....	72
Referencias Bibliográficas.....	73
Bibliografía.....	74
Glosario de Términos.....	75
Anexos.....	81

Figura 1. Flujos de trabajo de Rup (requisitos, análisis, diseño, implementación y prueba) tienen lugar sobre las cuatro fases. [1]	26
Figura 2. Diagrama de clases del modelo de objetos del dominio.	32
Figura 3: Diagrama de casos de uso del sistema.....	43
Figura 4: Diagrama de casos de uso del sistema.	44
Figura 5. Diagrama de clases del diseño perteneciente al monitor (servidor).....	53
Figura 6. Diagrama de clases del diseño perteneciente al agente (clientes).....	54
Figura 7. Diagrama modelo lógico de datos.....	55
Figura 8. Diagrama del modelo físico de datos.....	56
Figura 9. Diagrama de implementación del agente.	57
Figura 10. Diagrama de implementación del monitor.....	58
Figura 11. Diagrama de implementación de la consola.	59

Tabla 3.1: Descripción de los actores del sistema	43
Tabla 3.2: Descripción del caso de uso de sistema Pedir Información.	44
Tabla 3.3: Descripción del caso de uso de sistema Recibir Información.	45
Tabla 3.4: Descripción del caso de uso de sistema Almacenar Información.	45
Tabla 3.5: Descripción del caso de uso de sistema Activar Alarma.	46
Tabla 3.6: Descripción del caso de uso de sistema Obtener Información.	46
Tabla 3.7: Descripción del caso de uso de sistema Estructurar Información.	46
Tabla 3.8: Descripción del caso de uso de sistema Enviar Información.	47
Tabla 3.9: Descripción del caso de uso de sistema Cambiar Contraseña.	47
Tabla 3.10: Descripción del caso de uso de sistema Autenticar.	48
Tabla 3.11: Descripción del caso de uso de sistema Gestionar Usuario.	48
Tabla 3.12: Descripción del caso de uso de sistema Gestionar Locales.	49
Tabla 3.13: Descripción del caso de uso de sistema Configurar Monitor.	49
Tabla 3.14: Descripción del caso de uso de sistema Mostrar Todo.	49
Tabla 3.15: Descripción del caso de uso de sistema Mostrar PlacaBase.	50
Tabla 3.16: Descripción del caso de uso de sistema Mostrar Procesador.	50
Tabla 3.17: Descripción del caso de uso de sistema Mostrar Video.	51
Tabla 3.18: Descripción del caso de uso de sistema Mostrar Bios.	51
Tabla 3.19: Descripción del caso de uso de sistema Mostrar HDD.	51
Tabla 3.20: Descripción del caso de uso de sistema Mostrar Red.	52
Tabla 3.21: Descripción del caso de uso de sistema Mostrar Ram.	52
Tabla 3.22: Descripción del caso de uso de sistema Emitir Reportes.	52
Tabla 4.1 Entradas externas	63
Tabla 4.2 Salidas externas	64
Tabla 4.3 Peticiones	64
Tabla 4.4 Ficheros lógicos internos	65
Tabla 4.5 Puntos de función	65
Tabla 4.6 Miles de Instrucciones fuentes	66

Introducción

Las redes informáticas son un conjunto de técnicas, conexiones físicas y programas informáticos empleados para conectar dos o más computadoras con el objetivo de que los usuarios pueden compartir ficheros, impresoras y otros recursos, enviar mensajes electrónicos y ejecutar programas en otros ordenadores.

En las universidades cubanas las redes son el sustento de servicios vitales para el desarrollo de las labores investigativas y docentes. La universidad de Cienfuegos no es una excepción, contando con una gran cantidad de equipos conectados a su red. En especial, la facultad de informática cuenta con varios laboratorios a disposición de los usuarios para llevar a cabo sus actividades. Con el decursar de los años, teniendo en cuenta el incremento del número de usuarios que acceden a nuestros laboratorios y de la cantidad de equipos disponibles, se ha hecho más difícil mantener los niveles de seguridad físicas sobre las computadoras de la red.

Por este motivo el **problema a resolver** está dado por la ausencia de una herramienta que nos permita llevar el control de estos recursos y que a su vez facilite la gestión de los reportes que deben ser entregados en la dirección de la facultad.

Teniendo en cuenta esta situación, nos dimos a la tarea de investigar herramientas existentes que pudieran solucionar el problema o servir de punto de partida para adaptarlos a nuestras necesidades. Luego de realizar un estudio detallado de herramientas como Everest, Aida, Sandra, Network Assert Tracker, entre otros, encontramos inconvenientes como:

1. Muchos no poseen una Base de Datos centralizada en la cual almacenar la información de forma automática.
2. No detectan cambios en el hardware, ni emiten alarmas.
3. Unos pocos son gratis y la mayoría propietarios.

Nuestro **Objeto de estudio** se centra en la red de la facultad de informática y el **campo de acción** en el proceso de monitoreo del hardware de dicha red.

La **idea a defender** con esta investigación es que al desarrollar un sistema informático que realice inventarios del hardware de la red de la facultad de Informática, se podrá tener un mayor control del mismo y se podrán generar eficientemente los reportes necesarios.

Objetivo general consiste en: Desarrollar una herramienta que monitoree el hardware de las computadoras con Windows en la red, alerte de posibles cambios y genere los reportes necesarios.

Del mismo se desprenden los siguientes **objetivos específicos**:

- Investigar la existencia de sistemas vinculados al campo de acción.
- Crear una base de datos que garantice el almacenamiento, seguridad e integridad de la información.
- Construir el sistema propuesto.
- Probar el sistema por un periodo de dos semanas para la detección de errores y la evaluación por parte del usuario.

Para cumplir estos objetivos se realizaron las siguientes tareas:

1. Analizar algunos de los productos existentes en la actualidad que se utilizan para monitorear el hardware de las PC, con vistas a identificar las ventajas y desventajas que estos brindan.
2. Diseñar la base de datos del sistema.
3. Implementar la base de datos del sistema.
4. Hacer un estudio relacionado con los servicios de Windows y como se implementan.
5. Hacer un estudio sobre el uso de los sockets en Windows.
6. Construir el sistema propuesto.

7. Someter el sistema a un periodo de prueba determinado por el usuario.
8. Corregir errores encontrados en la fase de prueba.

El presente documento está estructurado en cuatro capítulos.

Capítulo 1: Este capítulo trata lo referente a la Fundamentación Teórica, abordando temas relacionados con el dominio del problema, objeto de estudio y tecnologías actuales.

Capítulo 2: Este capítulo trata lo referente al Modelo del Dominio, define las entidades y los conceptos fundamentales así como las reglas del negocio a considerar.

Capítulo 3: Este capítulo está relacionado con los Requisitos del sistema, en él se describe el sistema propuesto y se definen los requerimientos funcionales así como los no funcionales, mostrando el modelo de casos de uso con sus actores, diagrama de clases, el diseño de la base de datos así como el diagrama de implementación.

Capítulo 4: Este capítulo aborda lo relacionado con el Estudio de Factibilidad.

Capítulo I – Fundamentación Teórica.

1.1– Introducción.

Este Capítulo aborda lo relacionado con la descripción del dominio del problema donde se describen los principales conceptos para comprender el objeto de estudio, objetivos estratégicos de la organización, descripción de los sistemas existentes así como las tendencias, metodologías y/o tecnologías actuales.

1.2– Descripción del dominio del problema.

1.2.1– ¿Qué es una Red LAN?

Local Area Network, es la interconexión de varios ordenadores y periféricos. Su extensión esta limitada físicamente a un edificio o a un entorno de hasta 100 metros. Su aplicación más extendida es la interconexión de ordenadores personales y estaciones de trabajo en oficinas, fábricas, etc., para compartir recursos e intercambiar datos y aplicaciones. En definitiva, permite que dos o más máquinas se comuniquen.

1.2.2– Topología.

Es el patrón de interconexión de los participantes en la red, entre las topologías típicas están: estrella, anillo, bus (lineal) y malla. Un por ciento significativo del costo de una red la tiene el costo del cableado, de ahí que la Topología defina el tipo de red a instrumentar.

1.2.3– Topología de Bus.

La topología de bus tiene todos sus nodos conectados directamente a un enlace y no tiene ninguna otra conexión entre nodos. Físicamente cada host está conectado a un cable común, por lo que se pueden comunicar directamente, aunque la ruptura del cable hace que los hosts queden desconectados. La topología de bus permite que todos los dispositivos de la red puedan ver todas las señales de todos los demás dispositivos, lo que puede ser ventajoso si desea que todos los dispositivos obtengan esta información. Sin embargo, puede representar una desventaja, ya que es común

que se produzcan problemas de tráfico y colisiones, que se pueden paliar segmentando la red en varias partes. Es la topología más común en pequeñas LAN. Los extremos del cable se terminan con una *resistencia denominada terminador*, que además de indicar que no existen más ordenadores en el extremo, permiten cerrar el bus. También representa una desventaja ya que si el cable se rompe, ninguno de los ordenadores tendrá acceso a la red (ver **Anexo A.1**).

1.2.4— Topología de Anillo.

En este tipo de red la comunicación se da por el paso de un token o testigo, que se puede conceptualizar como un cartero que pasa recogiendo y entregando paquetes de información, de esta manera se evitan eventuales pérdidas de información debidas a colisiones. Cabe mencionar que si algún nodo de la red deja de funcionar, la comunicación en todo el anillo se pierde. En un anillo doble, dos anillos permiten que los datos se envíen en ambas direcciones. Esta configuración crea redundancia (tolerancia a fallos), lo que significa que si uno de los anillos falla, los datos pueden transmitirse por el otro (ver **Anexo A.2**).

1.2.5— Topología de Estrella.

Una **red en estrella** es una red en la cual las estaciones están conectadas directamente a un punto central y todas las comunicaciones se han de hacer necesariamente a través de este dado su transmisión. Una red en estrella activa tiene un nodo central *activo* que normalmente tiene los medios para prevenir problemas relacionados con el eco. Se utiliza sobre todo para redes locales. La mayoría de las redes de área local que tienen un enrutador (router), un conmutador (switch) o un concentrador (hub) siguen esta topología. El nodo central en estas sería el enrutador, el conmutador o el concentrador, por el que pasan todos los paquetes (ver **Anexo A.3**).

1.2.6— Topologías Híbridas de LANs.

La topología híbrida es el conjunto de todas las anteriores. Su implementación se debe a la complejidad de la solución de red, o bien al aumento en número de dispositivos, lo que hace necesario establecer una topología de este tipo. Las topologías híbridas tienen un alto costo debido a su administración y mantenimiento, ya que cuentan con segmentos de diferentes tipos, lo que obliga a invertir en equipo adicional para lograr la conectividad deseada (ver **Anexo A.4**).

1.3 – Descripción del objeto de estudio.

1.3.1 - Objetivos estratégicos de la organización.

Facultad de Informática

La Facultad de Informática de la UCF fue creada en Septiembre del 2000 y esta compuesta por dos Departamentos: Matemática e Informática.

Misión de la Facultad de Informática.

La Facultad de Informática, con un colectivo de trabajadores comprometidos con su patria, garantiza la formación integral de Ingenieros Informáticos y la superación continua de profesionales con capacidad para asumir las TIC. Participa en la formación integral de profesionales de otras especialidades, desarrolla investigaciones y servicios científico-técnicos, contribuyendo a la transformación y perfeccionamiento de la sociedad cubana en correspondencia con los programas de la Revolución.

Visión de la Facultad de Informática.

La Facultad de Informática es una organización que:

- Es líder en la implantación y fortalecimiento de las nuevas tecnologías de la información y las comunicaciones en el territorio.

- Forma profesionales caracterizados por su integralidad, incondicionalidad, compromiso revolucionario, creatividad, y competitividad.
- Da respuesta de manera ágil y actualizada a las necesidades demandadas por el Pregrado, Postgrado y la Ciencia y Técnica propiciando la transformación y perfeccionamiento de la sociedad cubana con un alto nivel de efectividad.
- Es distinguida por su claustro de alto nivel científico y pedagógico.
- Cuenta con un estado físico ambiental universitario, laboratorios funcionales y medios computacionales que facilitan la obtención de un clima laboral propicio al desarrollo de la organización.
- Mantiene un nivel estable de relaciones interinstitucionales con los organismos de la Administración Central del Estado, Centros de Educación Superior nacionales y extranjeros y otras entidades que permiten el desarrollo y financiamiento de la actividad científica y académica.

1.3.2 - Flujo actual de los procesos y análisis crítico de la ejecución de estos.

Actualmente no se poseen datos suficientes relacionados con el hardware de las PC en la red de la facultad de informática. Cuando ocurre una pérdida o cambio de algún componente, ya sea una memoria RAM, placa base o un disco duro, los administradores se dirigen a los técnicos de laboratorios para chequear el libro de incidencias. De esta forma se establece un intervalo de tiempo en el que posiblemente haya ocurrido el hecho y se chequea quiénes usaron la computadora en ese tiempo. En ocasiones esta operación resulta engorrosa, ya que puede darse el caso de que una computadora permanezca fuera de servicio por presentar alguna rotura y en ese momento ocurrir un cambio, de tal modo que los administradores no pueden descubrir ni la hora ni el día en que esto ocurrió. Incluso, en el peor de los casos, pueden transcurrir demasiado tiempo antes de ser detectada la irregularidad.

1.4 – Descripción de los sistemas existentes.

1.4.1—Everest Home Edition.

EVEREST Home Edition es una aplicación gratuita que muestra información sobre una parte de las características de las PC, detalles sobre el hardware, software, overclocking, benchmarks y enlaces a cada uno de los fabricantes. Permite generar reportes y dispone para su uso de 27 idiomas diferentes.

1.4.2—Everest Corporate Edition.

EVEREST Corporate Edition es un sistema de información y diagnóstico para entornos de red en pequeñas y grandes corporaciones. Utiliza las últimas tecnologías, incluyendo reportes XML y MHTML, y reportes completos para Base de datos SQL y Windows Server 2003. Ofrece información precisa y capacidad de diagnóstico, incluyendo mejoras de memoria, monitoreo de hardware y una información de hardware a bajo nivel.

1.4.3—Total Network Inventory.

Total Network Inventory es un programa de inventario de PC o sea un programa de inventario de red, para redes de oficina y de empresas grandes. Este interroga todos los ordenadores y portátiles en la red y genera la información completa en los equipos remotos sobre el sistema operativo, Service Pack, compilación, dispositivos, software, procesos ejecutados, entre otros. Esta información se añade a la base de datos centralizada y los administradores de la red pueden generar informes para cada PC o para todos los equipos en la red. El programa no necesita agente de software ni programa instalado en los equipos remotos.

1.4.4— LOGINventory.

LOGINventory permite realizar en pocos minutos un inventario de todo el software y hardware de una red Windows sin instalar software adicional ni agentes en los clientes. Esto posibilita tener un control total del software y hardware existente en la red corporativa de cualquier compañía, en cualquier momento y en forma rápida.

1.4.5— AjpdSoft Inventario PCs.

Aplicación Open Source que extrae los datos más importantes del PC donde se ejecuta (procesador, memoria RAM, dispositivos, software instalado, hardware, procesos en ejecución, datos de red, modem, monitor, impresoras, unidades de red, variables de entorno, discos duros, DNS, WinSock, IPs, datos del usuario, etc) y los almacena en una base de datos Access especificada para su posterior consulta y tratamiento.

1.4.6—OCS Inventory NG.

Open Computer and Software Inventory Next Generation es una aplicación open source y multiplataforma destinada para obtener información de hardware y software en una subred. El mismo requiere para su correcto funcionamiento de un agente tanto en el cliente como en el servidor.

1.4.7— Otras herramientas.

Existe una gran variedad de programas para inventariar el hardware de las computadoras, los mismos se pueden clasificar en dos grupos: atendidos y desatendidos.

Atendidos: son aquellos programas que necesitan de un cliente por máquina para poder ejercer su función.

Desatendidos: son aquellos que no necesitan instalar ningún cliente para poder obtener información.

A continuación mostramos algunos de estos programas:

Atendidos	Licencia	Desatendidos	Licencia
SIW (System Information for Windows)	freeware	Network Asset Tracker	shareware
Everest Home Edition	freeware	Audit Wizard	shareware
Aida32	freeware	Network Management Suite	shareware
Sandra	freeware	Total Network Inventory	shareware
MiTec System Information	freeware	Everest Corporate Edition	shareware
		LOGINventory	shareware

En el marco de la búsqueda que se llevó a cabo en esta investigación, se pudo llegar a la conclusión de que los softwares analizados no pueden ser adaptados a nuestras necesidades porque:

- Los open source, para adaptarlos, debemos de dominar el lenguaje de programación en que fueron creados para poder entender y modificar el código fuente. Esto traería consigo que el software no fuese entregado en tiempo.
- Se distribuyen bajo licencias comerciales, o sea son propietarios.
- No poseen una base de datos centralizada en la cual almacenar la información de forma automática.
- No son capaces de detectar cambios ni de emitir alarmas.

1.4.8— El sistema propuesto.

El sistema que proponemos desarrollar con esta investigación está compuesto por tres partes fundamentales: Agente, Monitor y Consola. Por sus características éste software pertenece al grupo de los atendidos.

Agente: Es un servicio que se ejecuta en las computadoras a las cuales se les está monitoreando el hardware. Dicho Agente es el encargado de recolectar la información del hardware de la computadora y enviarla al Monitor a través de un Socket, empleando en ello el puerto 4330. El uso de este puerto se justifica en la RFC mostrada en el **anexo C**.

Monitor: es un servicio que se ejecuta en una única computadora (típicamente un servidor) y es el encargado de recolectar los paquetes de información enviados por los Agentes, con dichos paquetes debe mantener actualizada una base de datos. También debe indicar periódicamente a los Agentes que deben enviar un paquete de información, intercambiar con los Agentes valores de configuración así como recibir órdenes y comandos desde la Consola.

Consola: es la interfaz del sistema para los usuarios del mismo. Es la encargada de mostrar toda la información al usuario referente al monitoreo del hardware, además debe procesar y generar las alarmas, realizar reportes, indicar valores de configuración.

Con este sistema se pretende suplir las necesidades de los responsables de la red de la facultad de informática, contando con una herramienta propia que a diferencia de las otras existentes, alerte a los administradores en caso de cambio o pérdida del hardware y realice los reportes necesarios que deben ser entregados con frecuencia en la dirección de la facultad.

1.5 – Tendencias, metodologías y/o tecnologías actuales.

1.5.1 La Metodología RUP y el lenguaje UML.

La tendencia actual en el software lleva a la construcción de sistemas más grandes y más complejos. Esto es debido en parte al hecho de que las computadoras son cada vez más potentes, y los usuarios, por tanto, esperan más de ellas [1].

A partir de lo expuesto anteriormente la compañía norteamericana *Rational Software Corporation* creó, a mediados de 1998, el Proceso Unificado de Rational (RUP).

RUP es un proceso de desarrollo de software, o sea, conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema de software. Es un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas de software, para diferentes áreas de aplicación, diferentes tipos de organizaciones, diferentes niveles de aptitud y diferentes tamaños de proyectos [1].

Es un proceso basado en componentes, que utiliza el Lenguaje Unificado de Modelado (UML) para preparar todos los esquemas de un sistema software. No obstante, los verdaderos aspectos definitorios del Proceso Unificado se resumen en que está dirigido por casos de uso, centrado en la arquitectura y es iterativo e incremental [1].

UML es un lenguaje que permite la modelación de sistemas con tecnología orientada a objetos. Aprobado como estándar por el OMG (Object Management Group) en noviembre de 1997.

Como sus principales objetivos UML persigue: **[2]**

- Poder ser usado por todos los modeladores.
- Incluir todos los conceptos que se consideran necesarios para utilizar un proceso moderno iterativo, basado en construir una sólida arquitectura para resolver requisitos dirigidos por casos de uso.
- Ser tan simple como sea posible pero manteniendo la capacidad de modelar toda la gama de sistemas que se necesita construir.
- Ser lo suficientemente expresivo para manejar todos los conceptos que se originan en un sistema moderno, tales como la concurrencia y distribución, así como también los mecanismos de la ingeniería de software, como son la encapsulación y componentes.
- Ser un lenguaje universal, como cualquier lenguaje de propósito general.
- Imponer un estándar mundial.

La figura 1 muestra el esquema iterativo y funcional estructurado en forma bidimensional, que propone el Proceso Unificado de Desarrollo (RUP). En el eje vertical están los distintos flujos de trabajo; y en el eje horizontal la evolución en el tiempo que se da en cuatro fases.

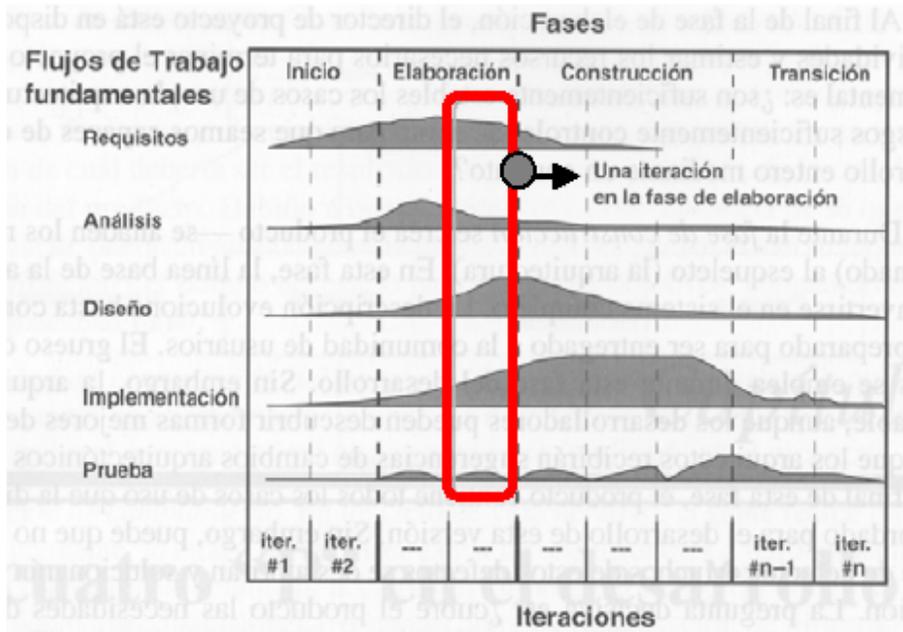


Figura 1. Flujos de trabajo de Rup (requisitos, análisis, diseño, implementación y prueba) tienen lugar sobre las cuatro fases. [1]

Después del análisis realizado se decidió, por parte de los autores, utilizar para la elaboración del presente documento y para llevar a cabo paso a paso todo el proceso de desarrollo del software propuesto la metodología RUP. Esto responde fundamentalmente a que esta metodología se ha convertido en un estándar internacional para guiar el proceso de desarrollo de software, al igual que en nuestro país y además porque contamos también con la herramienta Rational Rose del 2003, con la que se han elaborado todos los diagramas incluidos en este documento [3].

1.5.4 Lenguajes de Programación. Herramientas visuales de desarrollo.

Es necesario convencerse de que no existe ninguna herramienta ideal que cubra todos los intereses o sea la mejor en todos los aspectos; y es por esto que empresas como Microsoft o Inprise (Borland) tienen entre sus productos más de un entorno [4].

Dos de los entornos que gozan de un gran prestigio actualmente son:

- Borland Delphi
- Borland C++ Builder

Borland Delphi 6.0

El Object Oriented Pascal es el lenguaje que Delphi utiliza para crear las aplicaciones orientadas a objetos. Debido a que Delphi pertenece a la empresa Borland, la potencia de éste puede compararse con el compilador de C++. Borland Delphi es un ambiente de desarrollo rápido de aplicaciones (RAD) muy flexible y fácil de usar. Estos últimos años ha tenido una gran repercusión dentro del mundo de la programación visual. Presenta un ambiente visual de desarrollo para aplicaciones controlados por eventos de usuario sobre interfaces gráficas. Proporciona una jerarquía muy extensa de clases de objetos reusables. En cuanto a información sobre técnicas de programación en Delphi, existe una gran cantidad de opciones a elegir, tales como miles de páginas Web, muchos foros de debate, sitios FTP que contienen una enorme cantidad de librerías, y mucha más información que puede ser obtenida a través de Internet. Delphi es una herramienta de propósito general, se puede programar tanto a bajo nivel, como a alto nivel (simplemente usando controles y ajustando propiedades) y tiene buenas capacidades gráficas. Las aplicaciones creadas en Delphi sólo funcionan sobre la plataforma de trabajo Windows [4].

Borland C++ Builder 6.0

El lenguaje C++ es también un lenguaje orientado a objetos. Tiene un amplio soporte para red ya que es un sistema fundamentalmente para la creación de aplicaciones que están conectadas a red. Las aplicaciones creadas en C++ sólo funcionan sobre la plataforma de trabajo Windows. Como Delphi, el Borland C++ Builder, es un ambiente de desarrollo rápido de aplicaciones (RAD) muy flexible. Constituye una potente herramienta para el desarrollo de aplicaciones en C++. Ofrece un entorno visual de desarrollo. Permite la importación de código C++ existente. Posee una gran cantidad de clases y objetos reusables. Es una herramienta de propósito general. Existe mucha documentación referente a la programación en Borland C++ Builder [4].

¿Por qué Borland Delphi 6.0?

A partir de un análisis comparativo de las herramientas anteriores se decidió que se utilizaría Borland Delphi 6.0 por los siguientes motivos:

- La plataforma Borland Delphi 6.0 brinda todas las prestaciones necesarias y requeridas para el desarrollo del tipo de software propuesto.
- La mayoría de las máquinas de la Facultad se sustentan sobre la plataforma Windows.
- Disponibilidad de una componente de software para obtener información del hardware. La reutilización de esta nos permitirá disminuir el tiempo de desarrollo, haciendo factible este trabajo.

1.5.6 Sistemas de Gestión de Bases de Datos.

Los sistemas informáticos de gestión, en su inmensa mayoría, operan sobre el almacenamiento externo de la información a través de Sistemas Gestores de Bases de Datos (SGBD).

Un Sistema de Gestión de Base de Datos (SGBD) es el software que permite la utilización y/o la actualización de los datos almacenados en una (o varias) base(s) de datos por uno o varios usuarios desde diferentes puntos de vista y a la vez [4].

En la actualidad existen un sinnúmero de SGBD, muchos de ellos muy potentes, entre los que se encuentran Microsoft SQL Server, Oracle, Postgree SQL, MySQL.

MySQL

MySQL es un sistema de administración de Base de Datos, con una arquitectura cliente/servidor. Es un proyecto "Open Source" y permite una fácil conectividad, alta velocidad de respuesta a solicitudes, y gran seguridad. Por ello se utiliza para acceder a bases de datos desde Internet.

MySQL es muy rápido, confiable y fácil de usar, es multiplataforma, multiusuario y permite elaborar consultas con el robusto SQL, además no tiene valor monetario, es un software que se puede adquirir libremente.

SQL-Server

Microsoft SQL Server, propietario de Microsoft, pertenece a la familia de los sistemas de administración de base de datos, operando en una arquitectura cliente/servidor de gran rendimiento. Su desarrollo fue orientado para hacer posible manejar grandes volúmenes de información, y un elevado número de transacciones. SQL Server es una aplicación completa que realiza toda la gestión relacionada con los datos. El servidor sólo tiene que enviarle una cadena de caracteres (la sentencia SQL) y esperar a que le devuelvan los datos.

SQL Server permite la creación de procedimientos almacenados, los cuales consisten en instrucciones SQL que se almacenan dentro de una base de datos de SQL Server, realizados en lenguaje SQL, se trata de procedimientos que se guardan semicompilados en el servidor y que pueden ser invocados desde el cliente. Se ejecutan más rápido que instrucciones SQL independientes.

SQL Server puede manejar perfectamente bases de datos de TeraBytes con millones de registros y funciona sin problemas con miles de conexiones simultáneas a los datos, sólo depende de la potencia del hardware del equipo en el que esté instalado y solamente corre sobre Windows NT- 2000 Server.

¿Por qué MySQL?

Porque MySQL es un sistema que es multiplataforma, multiusuario, multihilo y libre.

1.6 – Conclusiones.

Como se pudo apreciar en este capítulo se fundamenta la necesidad de desarrollar una herramienta que permita realizar un monitoreo de las computadoras con el fin de detectar de forma rápida pérdidas o cambios en el hardware de las mismas. Asimismo, a partir de un estudio de las tendencias y tecnologías actuales, se orienta utilizar la metodología RUP y el lenguaje UML, por las ventajas que proporcionan y el alto nivel de aceptación que han tenido, además del uso del Rational Rose 2003 para la confección de los diagramas que se necesitan por cada modelo. Borland Delphi 6.0 fue escogido por las facilidades que brinda su entorno de desarrollo para la gestión de aplicaciones visuales y la riqueza del lenguaje de programación Object Oriented Pascal. MySQL se seleccionó como SGBD por ser una poderosa herramienta relacional, multihilo, multiusuario, multiplataforma y libre.

Capítulo II – Modelo del Dominio

2.1 Introducción

En el presente capítulo tomando como guía la Metodología RUP, se utiliza uno de los artefactos que brinda dicha metodología: el Modelo de Dominio, el cual ayuda a modelar y describir la solución propuesta. Además, se presenta una descripción detallada de las reglas de negocio que el objeto de automatización debe seguir para asegurar el cumplimiento de las restricciones que existen en el dominio.

2.2 Descripción del modelo del dominio

Un Modelo del Dominio captura los tipos más importantes de objetos en el contexto del sistema. Los objetos del dominio representan las "cosas" que existen o los eventos que suceden en el entorno en el que trabaja el sistema. Muchos de los objetos del dominio o clases pueden obtenerse de una especificación de requisitos [1]. La modelación del dominio tiene como objetivo fundamental la comprensión y descripción de las clases más importantes en el sistema.

El modelo de dominio empleado en este trabajo define las siguientes entidades principales: Administrador, Consola, Reportes y Alarma. La esencia del funcionamiento de este dominio, radica en la constante labor de los administradores de la facultad de Informática por conocer el estado del hardware de las PC en red, con el objetivo de tener un mayor control sobre las componentes de las mismas y agilizar los inventarios en la facultad. Con este objetivo se desarrolla un sistema el cual, teniendo un agente en cada PC cliente y un Monitor en el servidor, logra recopilar los datos necesarios y procesarlos a una buena velocidad. Una vez almacenados estos datos son utilizados por el administrador para tener los reportes necesarios. El sistema también es capaz de avisar rápidamente a los administradores en caso de que haya un cambio en el hardware de las PC clientes.

2.2.1 Modelo de objetos del dominio

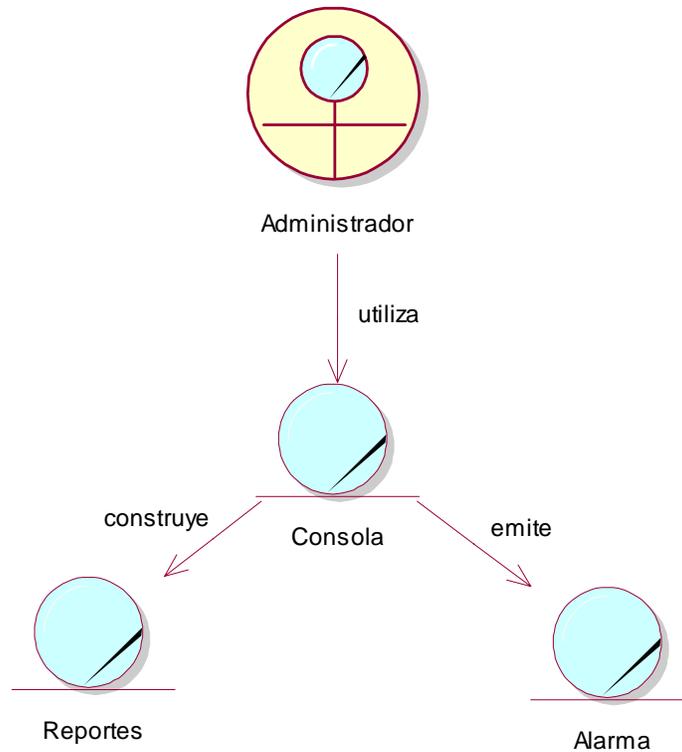


Figura 2. Diagrama de clases del modelo de objetos del dominio.

2.3 Reglas del negocio a considerar.

Las reglas de negocio regulan y describen las principales políticas que deben cumplirse para el adecuado funcionamiento del negocio. Para nuestro caso se definen las siguientes:

1. Los servicios del servidor y el cliente siempre deben estar ejecutándose.
2. La consola debe tener conexión en todo momento con la Base de Datos.
3. Solo los administradores pueden acceder a la información mediante la consola.
4. Solo los administradores pueden instalar y activar los servicios del servidor y la consola.

2.4 Conclusiones

En el presente capítulo queda definido el modelo de objetos del dominio correspondiente a la obtención de datos de las PC en red. Además fueron descritas las reglas que regulan y conducen al buen funcionamiento del negocio. Todo este análisis permitió desarrollar una visión nueva y más clara del problema a resolver.

Capítulo III – Modelo del sistema

3.1 – Introducción

En el presente capítulo se describe y analiza el modelo de sistema del objeto de automatización sobre la base de las especificaciones de la metodología RUP. Se identifican los requerimientos funcionales y no funcionales, se definen a los actores del sistema y a los servicios o funcionalidades que a su disposición se colocan (los casos de uso del sistema).

Además, se plantean y detallan una serie de diagramas que nos ayudan y guían en la implementación del modelo de sistema, como son: el diagrama de casos de uso del sistema, el diagrama de clases del diseño, el diagrama del modelo físico y lógico de datos y el diagrama de implementación.

3.2 – Descripción del modelo de sistema

El sistema propuesto tiene como objetivo el control del hardware de las PC (derivado de las palabras inglesas Personal Computer, se usa para denominar las computadoras personales en general) de la red en la facultad de Informática. El sistema presenta grandes opciones entre las que se encuentran la recopilación de toda la información en cuanto al hardware de las PC, luego esa información es almacenada en la base de datos para un análisis futuro. También tiene la opción en el caso de que cuanto se capture la información no sea igual a la guardada en la base de datos se active una alarma para que los administradores sepan que tienen un problema en la PC.

3.3 – Modelación del modelo de sistema

El modelado de Casos de Uso es la técnica más efectiva y a la vez la más simple que emplean los desarrolladores de software para modelar los requisitos del sistema desde la perspectiva del usuario. El modelo de casos de uso consiste en actores y casos de uso. Los actores representan usuarios y otros sistemas que interaccionan con el sistema y los casos de uso representan el comportamiento del sistema, los escenarios que el sistema atraviesa en respuesta a un estímulo desde un actor [5].

En esencia, el modelado de Casos de Uso nos describe lo que hace el sistema para cada tipo de usuario y nos ofrece un medio correcto para el análisis, el diseño y las pruebas.

3.3.1 – Requerimientos funcionales

Los requerimientos funcionales son declaraciones de los servicios o funciones que proveerá el sistema, de la manera en que éste reaccionará a entradas particulares. Estos dependen del tipo de software y del sistema que se desarrolle y de los posibles usuarios del software. Cuando se expresan como requerimientos del usuario, habitualmente se describen de forma general mientras que los requerimientos funcionales del sistema describen con detalle la función de éste, sus entradas y salidas, excepciones, etc.

En algunos casos, los requerimientos funcionales de los sistemas también declaran explícitamente lo que el sistema no debe hacer.

1. Pedir archivo de información.
2. Recibir comandos de configuración.
3. Obtener fecha del Bios.
4. Obtener versión del Bios.
5. Obtener modelo del Disco Duro.
6. Obtener número de serie del Disco Duro.
7. Obtener capacidad del Disco Duro.
8. Obtener IP de la máquina.
9. Obtener nombre de la PC.
10. Obtener marca de la Placa Base.
11. Obtener modelo de la placa base.
12. Obtener versión de la placa base.

13. Obtener número de serie de la placa base.
14. Obtener cantidad de procesadores.
15. Obtener marca del procesador.
16. Obtener modelo del procesador.
17. Obtener frecuencia del procesador.
18. Obtener RAM total.
19. Obtener RAM libre.
20. Obtener RAM ocupada.
21. Obtener memoria virtual total.
22. Obtener memoria virtual libre.
23. Obtener memoria virtual ocupada.
24. Obtener dominio.
25. Obtener MAC.
26. Obtener DNS.
27. Obtener puerta de enlace.
28. Obtener máscara.
29. Obtener tarjeta de red.
30. Obtener versión del Sistema Operativo.
31. Obtener plataforma del sistema operativo.
32. Obtener organización.
33. Obtener modelo de la tarjeta de video.

34. Obtener resolución.
35. Obtener número de colores.
36. Obtener memoria de video.
37. Crear archivo de información.
38. Enviar archivo de Información.
39. Recibir archivo de Información.
40. Leer archivo de Información.
41. Almacenar Información.
42. Emitir sonido de alarma.
43. Enviar correo al administrador.
44. Autenticación en el sistema, para los usuarios autorizados.
45. Mostrar todos los datos almacenados.
46. Configuración del Monitor en cuanto al tiempo de actualización.
47. Insertar local.
48. Actualizar local.
49. Eliminar local.
50. Insertar usuario.
51. Eliminar usuario.
52. Cambiar contraseña de usuario en el sistema.
53. Mostar IP de las PC.
54. Mostar nombre de la PC.

55. Mostrar cantidad de los procesadores de la PC.
56. Mostar modelo de los procesadores de las PC.
57. Mostar modelo de los procesadores de las PC.
58. Mostrar marca de los procesadores de las PC.
59. Mostrar frecuencia de los procesadores de las PC.
60. Mostar marca de la placabase de las PC.
61. Mostar modelo de la placabase de las PC.
62. Mostar numero de serie de la placabase de las PC.
63. Mostar versión de la placabase de las PC.
64. Mostrar versión del bios de las PC.
65. Mostrar fecha del bios de las PC.
66. Mostrar modelo de la tarjeta de video de las PC.
67. Mostrar resolución del monitor en ese momento de las PC.
68. Mostrar memoria de la tarjeta de video de las PC.
69. Mostrar no colores de la tarjeta de video de las PC.
70. Mostrar modelo de los discos duros de las PC.
71. Mostrar número de serie de los discos duros de las PC.
72. Mostrar modelo de los discos duros de las PC.
73. Mostrar capacidad de los discos duros de las PC.
74. Mostrar modelo de las tarjetas de red de las PC.
75. Mostrar dominio de las PC.

- 76. Mostrar dns de las PC.
- 77. Mostrar mac de las PC.
- 78. Mostrar mascara de las PC.
- 79. Mostrar puerta de enlace de las PC.
- 80. Mostrar ram total de las PC.
- 81. Mostrar ram libre de las PC.
- 82. Mostrar ram ocupada de las PC.
- 83. Mostrar virtual total de las PC.
- 84. Mostrar virtual libre de las PC.
- 85. Mostrar virtual ocupada de las PC.
- 86. Mostrar sistema de ayuda.
- 87. Emitir reporte de procesador.
- 88. Emitir reporte de placabase.
- 89. Emitir reporte de disco duro.

3.3.2 – Requerimientos no funcionales

Los requerimientos no funcionales describen las restricciones del sistema o del proceso de desarrollo; no se refieren directamente a las funciones específicas que entrega el sistema, sino a las propiedades emergentes de éste como la fiabilidad, la respuesta en el tiempo y la capacidad de almacenamiento. De forma alternativa, definen las restricciones del sistema como la capacidad de los dispositivos de entrada/salida, en cuanto a prestaciones, atributos de calidad y la representación de datos que se utiliza en la interfaz del sistema [5].

Apariencia e interfaz externa

- La interfaz se ajusta al estándar de ventanas que el sistema operativo Windows ha establecido e internacionalizado.
- La interfaz esta diseñada de modo tal que el usuario pueda tener en todo momento el control de la aplicación, lo que le permite ir de un punto a otro dentro de ella con gran facilidad. Se cuida porque la aplicación sea lo más interactiva posible. La interfaz es uniforme de manera que a pesar de tener varios módulos, el usuario los identifique como un mismo sistema

Requisitos de Usabilidad

- La explotación del sistema agiliza la actividad científica del centro donde se ponga en práctica gracias a las comodidades brindadas por esta aplicación y la rapidez.
- La información original del software sólo podrá ser modificada por especialistas en la materia.
- Tiene una ayuda básica que comprende los aspectos generales a tener en cuenta para trabajar con la aplicación.
- El sistema puede ser usado por aquellas personas que no tengan experiencia en el uso de la computadora, sólo necesitarían un ligero entrenamiento sobre el funcionamiento de los principales elementos de una interfaz estándar en el ambiente del sistema operativo Windows (uso del Mouse, manejo de menús, botones, cuadros de texto, etc.)

Requisitos de Rendimiento

- Se requiere de una capacidad de procesamiento baja para ejecutar la aplicación.
- Los tiempos de respuesta son cortos.

Requisitos de soporte

- Las pruebas del sistema se realizan en la Universidad de Cienfuegos. Dichas pruebas permiten evaluar en la práctica la funcionalidad y las ventajas de este nuevo sistema.
- El sistema da las posibilidades a futuras mejoras y nuevas opciones que se le quieran incorporar.

Requisitos de portabilidad

- La plataforma seleccionada para desarrollar la aplicación fue Windows.

Requisitos de político-culturales

- El nivel social, cultural o étnico; no determinarán una prioridad o limitante a la hora de brindar los servicios que ofrece el producto.

Requisitos legales

- Este software es propiedad intelectual de la Universidad de Cienfuegos y solo es permitida su comercialización con el consentimiento de los autores y la entidad involucrada.

Requisitos de confiabilidad

- El sistema presenta un mecanismo de respuesta rápida ante fallos y que en caso de ocurrencia se minimizan las pérdidas de información.

Requerimientos de ayuda y documentación en línea

- Se dispone de una ayuda bien detallada sobre las principales opciones del sistema.

Requerimientos de software

- Se debe disponer de Windows 95 o superior, para la instalación de la aplicación.
- Se debe garantizar antes de llevar a cabo la instalación tanto del Monitor con del Agente que todas las aplicaciones estén cerradas incluyendo Delphi.
- Antes de instalar el monitor y ejecutar la Consola debe de instalarse el driver ODBC para MySQL y configurarse el mismo desde los orígenes de datos en DSN del sistema.
- Para el buen funcionamiento de la aplicación no debe mandarse a hibernar ninguna estación de trabajo.

Requerimientos de seguridad

- ✓ Solo los administradores deben poseer la cuenta de administrador de las maquinas para evitar que se detengan los servicios.
- ✓ Debe existir al menos un usuario en la base de datos para poder usar el sistema.

Requerimientos de hardware (mínimos)

- Familia de procesadores 486.
- 128 o más de Ram.
- 3 GB de Hdd.
- Mouse.
- Teclado.
- Todas las estaciones de trabajo deben estar en red.

3.3.3 – Actores del modelo de sistema

Actor	Descripción
Administrador	Cualquier administrador que interactúe con el sistema. Este usuario tendrá acceso a todos los requerimientos funcionales del sistema.
Monitor	Es un servicio que se ejecuta en una única computadora (típicamente un servidor) y es el encargado de recolectar los paquetes enviados por los agentes.
Agente	Es un servicio que se encuentra ejecutándose en las computadoras a las cuales se le está monitoreando el hardware y a su vez es el encargado de recolectar y enviar la información al monitor.

Tabla 3.1: Descripción de los actores del sistema

3.3.4 – Diagramas de casos de uso del sistema

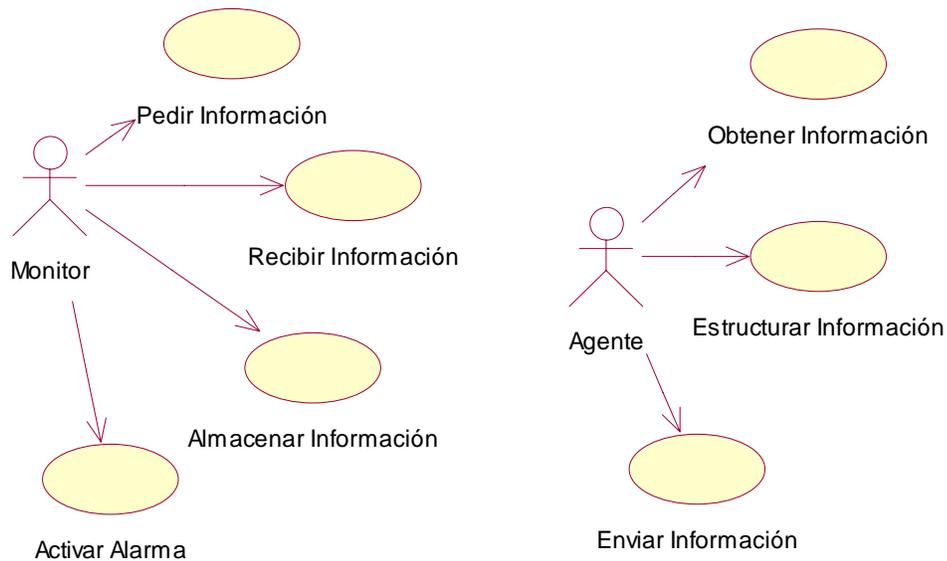


Figura 3: Diagrama de casos de uso del sistema.

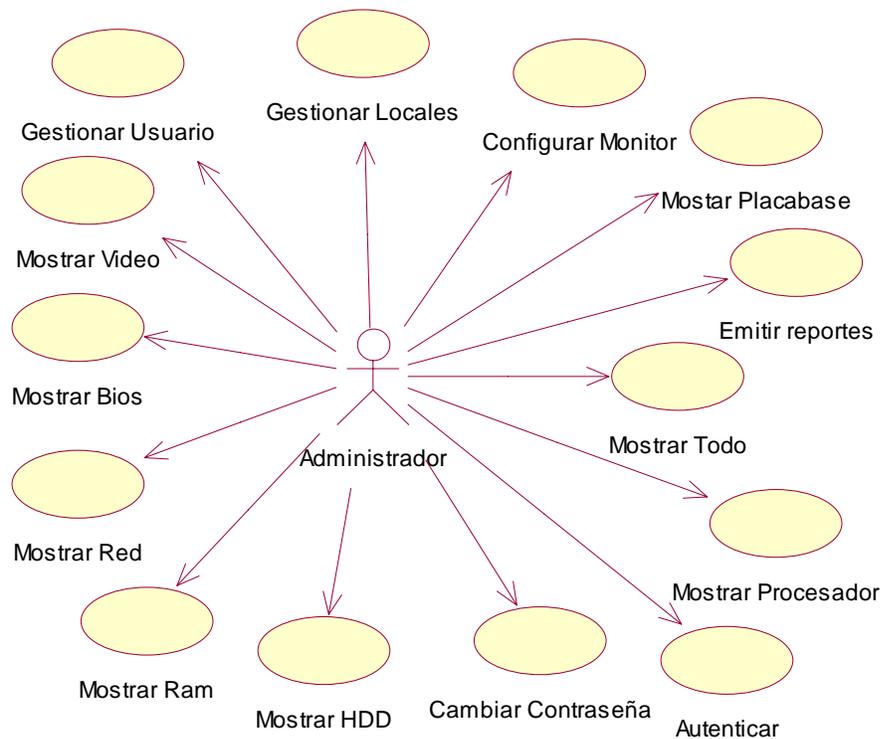


Figura 4: Diagrama de casos de uso del sistema.

3.3.6 – Descripción de los casos de usos del sistema

Caso de uso	Pedir Información
Actores	Monitor(inicia)
Propósito	Pedir a los Agentes la Información.
Resumen	El Monitor según el parámetro de configuración de actualización recibido por la Consola revisa todas las conexiones activas que posee realizando un ping a todos los agentes conectados en el momento, de esta forma y con intervalos de 2 minutos por máquina comienza a pedirle a cada agente que envíen el archivo con la información, el caso de uso culmina una vez que sea recibido el archivo.
Referencias	R1, R2.
Precondiciones	Debe existir al menos un agente conectado.
Post-condiciones	El monitor debe estar en ejecución.
Requisitos Especiales	No desconectar el cable de red.
Prototipo	

Tabla 3.2: Descripción del caso de uso de sistema Pedir Información.

Caso de uso	Recibir Información.
Actores	Monitor(inicia)
Propósito	Recibir el archivo enviado por los agentes.
Resumen El monitor inicia este caso de uso en la medida que van llegando los archivos con la información, en el momento que recibe el archivo se procede a leerlo para almacenar temporalmente la información en las estructuras creadas para ello. El caso de uso culmina una vez que se haya leído y estructurado toda la información.	
Referencias	R39, R40.
Precondiciones	El monitor debe estar en ejecución.
Post-condiciones	La información debe quedar estructurada correctamente.
Requisitos Especiales	
Prototipo	

Tabla 3.3: Descripción del caso de uso de sistema Recibir Información.

Caso de uso	Almacenar Información.
Actores	Monitor(inicia)
Propósito	Almacenar la información del archivo en la base de datos.
Resumen El monitor inicia este caso de uso cada vez que se recibe un archivo y el mismo ha sido leído y estructurado, seguidamente se procede a almacenar la información contenida en las estructuras hacia la base de datos culminando de esta forma el caso de uso.	
Referencias	R41.
Precondiciones	Debe existir una base de datos.
Post-condiciones	La información debe quedar guardada correctamente en la base de datos.
Requisitos Especiales	Debe estar instalado el driver ODBC para MySQL.
Prototipo	

Tabla 3.4: Descripción del caso de uso de sistema Almacenar Información.

Caso de uso	Activar alarma.
Actores	Monitor(inicia)
Propósito	Hacerle saber al administrador que ha ocurrido un cambio en el hardware de alguna PC.
Resumen El monitor inicia este caso de uso cada vez que se detecte un cambio en la información almacenada previamente con la información nueva que se recibe y culmina con un sonido de alarma y la emisión de un correo al administrador para alertarlo del cambio ocurrido.	
Referencias	R42, R43.
Precondiciones	Debe existir alguna información en la base de

	datos.
Post-condiciones	La nueva información queda almacenada en la base de datos.
Requisitos Especiales	Debe existir un servidor de correo.
Prototipo	

Tabla 3.5: Descripción del caso de uso de sistema Activar Alarma.

Caso de uso	Obtener Información.
Actores	Agente(inicia)
Propósito	Obtener información del hardware de la PC.
Resumen El caso de uso inicia una vez que se haya recibido por parte del monitor un mensaje de que ya se puede enviar el archivo, posteriormente se procede a capturar de cada componente hardware su información y culmina el caso de uso cuando dicha información quede almacenada en sus estructuras respectivas.	
Referencias	R3, R4, R5, R6, R7, R8, R9, R10, R11, R12, R13, R14, R15, R16, R17, R18, R19, R20, R21, R22, R23, R24, R25, R26, R27, R28, R29, R30, R31, R32, R33, R34, R35, R36.
Precondiciones	Los agentes deben estar en ejecución.
Post-condiciones	La información obtenida debe quedar estructurada.
Requisitos Especiales	
Prototipo	

Tabla 3.6: Descripción del caso de uso de sistema Obtener Información.

Caso de uso	Estructurar Información.
Actores	Agente(inicia)
Propósito	Crear el archivo que será enviado al monitor.
Resumen El Agente inicia el caso de uso una vez que la información obtenida del hardware haya sido estructurada, seguidamente se procede a crear el archivo que será enviado al monitor. El caso de uso termina cuando se haya generado el archivo.	
Referencias	R37.
Precondiciones	La información debe estar previamente estructurada.
Post-condiciones	La información debe quedar en el archivo creado.
Requisitos Especiales	
Prototipo	

Tabla 3.7: Descripción del caso de uso de sistema Estructurar Información.

Caso de uso	Enviar Información.
Actores	Agente(inicia)
Propósito	Enviar al monitor el archivo creado.
Resumen	El Agente inicia el caso de uso una vez que haya sido creado el archivo, seguidamente se procede a enviar al monitor dicho archivo por el puerto acordado, culminando de esta forma el caso de uso.
Referencias	R38.
Precondiciones	El archivo debe estar creado.
Post-condiciones	
Requisitos Especiales	
Prototipo	

Tabla 3.8: Descripción del caso de uso de sistema Enviar Información.

Caso de uso	Cambiar Contraseña
Actores	Administrador(inicia)
Propósito	Permitir que un usuario cambie su clave de acceso al sistema.
Resumen	El caso de uso se inicia cuando algún Administrador desea cambiar su contraseña. El sistema le muestra un formulario en el que debe introducir el nombre de usuario, la anterior contraseña y la nueva; confirmando esta última a fin de evitar posibles equivocaciones. Una vez que introduce los datos, el sistema confirma que estén completos y sean válidos. Si no lo son muestra mensaje de error y no realiza la modificación. En caso de que no existan errores, la contraseña es cambiada satisfactoriamente.
Referencias	R52
Precondiciones	Debe existir información almacenada del usuario registrado y este debe estar activo.
Postcondiciones	La contraseña se actualiza en caso de no existir ningún error.
Requisitos Especiales	
Prototipo	Ver Anexo B.1

Tabla 3.9: Descripción del caso de uso de sistema Cambiar Contraseña.

Caso de uso	Autenticar
Actores	Administrador(inicia)
Propósito	Proteger el acceso a la información.
Resumen	El caso de uso inicia cuando algún Administrador desea entrar al sistema. Para ello debe ingresar su login y su contraseña y a continuación el sistema chequea. Si los datos son válidos, el usuario podrá acceder a las opciones del sistema que le corresponden, en caso contrario el sistema muestra un mensaje de error denegando el acceso y finalizando así el caso de uso.
Referencias	R44

Precondiciones	El usuario debe estar registrado en sistema.
Postcondiciones	El usuario accede a la información del sistema que le corresponde.
Requisitos Especiales	
Prototipo	Ver Anexo 2

Tabla 3.10: Descripción del caso de uso de sistema Autenticar.

Caso de uso	Gestionar Usuario
Actores	Administrador(inicia)
Propósito	Insertar y eliminar usuario para proteger la información.
Resumen	El caso de uso se inicia cuando Administrador selecciona <i>Usuario</i> , el sistema muestra lo que se desea hacer <i>Insertar Usuario</i> y <i>Eliminar Usuario</i> . Este elige la operación a realizar. En caso de elegir <i>Insertar Usuario</i> se debe escribir el nombre del usuario a insertar teniendo en cuenta de que no puede haber nombre de usuario iguales y debe escribir la contraseña y confirmarla para que no vaya a ver equivocaciones, de haber algún problema el sistema le dará una alerta en caso contrario el usuario quedara insertado en el sistema. En caso de elegir la opción <i>Eliminar Usuario</i> le saldrán todos los usuarios del sistema que puede eliminar debe tener en cuenta que el usuario <i>root</i> el sistema no lo dejara eliminar por ser el súper administrador del sistema, en otro caso elija el que desee eliminar y presione OK.
Referencias	R50,R51
Precondiciones	Debe existir información almacenada del usuario registrado y este debe estar activo.
Postcondiciones	La contraseña se actualiza en caso de no existir ningún error.
Requisitos Especiales	
Prototipo	Ver Anexo B.3

Tabla 3.11: Descripción del caso de uso de sistema Gestionar Usuario.

Caso de uso	Gestionar Locales
Actores	Administrador(inicia)
Propósito	Insertar, actualizar y eliminar locales para tener una mayor organización a la hora de emitir los reportes.
Resumen	El caso de uso se inicia cuando Administrador selecciona <i>Configuración</i> y luego <i>Locales</i> , el sistema muestra lo que se desea hacer <i>Insertar</i> , <i>Actualizar</i> y <i>Eliminar locales</i> . En caso de elegir <i>Insertar</i> se debe escribir el nombre del Local a insertar y luego el IP inicial del local y el IP final. En caso de elegir <i>Actualizar</i> busca el local que desea actualizar y lo cargas y entonces cambias lo que deseas y luego lo salvas. En caso de elegir la opción <i>Eliminar</i> buscas el usuario el cual quieres eliminar y le das OK y ya.
Referencias	R47,R48,R49

Precondiciones	Debe existir algún local.
Postcondiciones	Queda insertado, eliminado o actualizado algún local.
Requisitos Especiales	
Prototipo	Ver Anexo B.4

Tabla 3.12: Descripción del caso de uso de sistema Gestionar Locales.

Caso de uso	Configurar Monitor
Actores	Administrador(inicia)
Propósito	Configurar el Monitor en cuanto a tiempo de pedido de información de los Agentes.
Resumen	El caso de uso se inicia cuando Administrador selecciona <i>Configuración y luego Configurar</i> , el sistema muestra donde usted debe poner el IP donde se encuentra el servidor y luego eliges el tiempo que desea ponerle en la configuración del Monitor.
Referencias	R46
Precondiciones	Debe haber algún Monitor ejecutado.
Postcondiciones	Queda configurado el Monitor en cuanto al tiempo de actualización.
Requisitos Especiales	
Prototipo	Ver Anexo B.5

Tabla 3.13: Descripción del caso de uso de sistema Configurar Monitor.

Caso de uso	Mostrar Todo
Actores	Administrador(inicia)
Propósito	Mostrar de todas las PC que hay en ese momento en el sistema, toda la información recopilada.
Resumen	El caso de uso se inicia cuando Administrador se autentifica en el sistema. Sale una pantalla donde aparecen todas las PC con la correspondiente información recopilada.
Referencias	R53,R54,R55,R56,R57,R58,R59,R60,R61,R62,R63,R64,R65,R66,R67,R68,R69,R70,R71,R72,R73,R74,R75,R76,R77,R78,R79,R80,R81,R82,R83,R84,R85
Precondiciones	Debe haber alguna PC en el sistema.
Postcondiciones	Muestra los valores de todas las PC en el sistema.
Requisitos Especiales	
Prototipo	Ver Anexo B.6

Tabla 3.14: Descripción del caso de uso de sistema Mostrar Todo.

Caso de uso	Mostrar Placabase
Actores	Administrador(inicia)
Propósito	Mostrar el placabase tanto de una PC, como de un rango de IP o de todas las PC en el sistema.
Resumen El caso de uso se inicia cuando Administrador selecciona <i>Consulta</i> y luego <i>Mostrar</i> hay le saldrá donde poner el IP de la maquina la cual quiere ver la placabase o para determinar si desea un rango de IP o de todas las PC. Luego chequea que desea ver placabase y presiona OK.	
Referencias	R53,R60,R61,R62,R63
Precondiciones	Debe haber alguna PC en el sistema.
Postcondiciones	Muestra los datos de la placabase de las PC en el sistema.
Requisitos Especiales	
Prototipo	Ver Anexo B.8

Tabla 3.15: Descripción del caso de uso de sistema Mostrar PlacaBase.

Caso de uso	Mostrar Procesador
Actores	Administrador(inicia)
Propósito	Mostrar el procesador tanto de una PC, como de un rango de IP o de todas las PC en el sistema.
Resumen El caso de uso se inicia cuando Administrador selecciona <i>Consulta</i> y luego <i>Mostrar</i> hay le saldrá donde poner el IP de la maquina la cual quiere ver el procesador o para determinar si desea un rango de IP o de todas las PC. Luego chequea que desea ver procesador y presiona OK.	
Referencias	R53, R55, R56, R57, R58, R59
Precondiciones	Debe haber alguna PC en el sistema.
Postcondiciones	Muestra los datos del procesador de las PC en el sistema.
Requisitos Especiales	
Prototipo	Ver Anexo B.7

Tabla 3.16: Descripción del caso de uso de sistema Mostrar Procesador.

Caso de uso	Mostrar Video
Actores	Administrador(inicia)
Propósito	Mostrar el video tanto de una PC, como de un rango de IP o de todas las PC en el sistema.
Resumen El caso de uso se inicia cuando Administrador selecciona <i>Consulta</i> y luego <i>Mostrar</i> hay le saldrá donde poner el IP de la maquina la cual quiere ver el video o para determinar si desea un rango de IP o de todas las PC. Luego chequea que desea ver video y presiona OK.	
Referencias	R53,R66,R67,R68,R69
Precondiciones	Debe haber alguna PC en el sistema.

Postcondiciones	Muestra los datos del video de las PC en el sistema.
Requisitos Especiales	
Prototipo	Ver Anexo B.9

Tabla 3.17: Descripción del caso de uso de sistema Mostrar Video.

Caso de uso	Mostrar Bios
Actores	Administrador(inicia)
Propósito	Mostrar el bios tanto de una PC, como de un rango de IP o de todas las PC en el sistema.
Resumen	El caso de uso se inicia cuando Administrador selecciona <i>Consulta</i> y luego <i>Mostrar</i> hay le saldrá donde poner el IP de la maquina la cual quiere ver el bios o para determinar si desea un rango de IP o de todas las PC. Luego chequea que desea ver bios y presiona OK.
Referencias	R53,R64,R65
Precondiciones	Debe haber alguna PC en el sistema.
Postcondiciones	Muestra los datos del bios de las PC en el sistema.
Requisitos Especiales	
Prototipo	Ver Anexo B.10

Tabla 3.18: Descripción del caso de uso de sistema Mostrar Bios.

Caso de uso	Mostrar HDD
Actores	Administrador(inicia)
Propósito	Mostrar el hdd tanto de una PC, como de un rango de IP o de todas las PC en el sistema.
Resumen	El caso de uso se inicia cuando Administrador selecciona <i>Consulta</i> y luego <i>Mostrar</i> hay le saldrá donde poner el IP de la maquina la cual quiere ver el hdd o para determinar si desea un rango de IP o de todas las PC. Luego chequea que desea ver hdd y presiona OK.
Referencias	R53,R70,R71,R72,R73
Precondiciones	Debe haber alguna PC en el sistema.
Postcondiciones	Muestra los datos del hdd de las PC en el sistema.
Requisitos Especiales	
Prototipo	Ver Anexo B.11

Tabla 3.19: Descripción del caso de uso de sistema Mostrar HDD.

Caso de uso	Mostrar Red
Actores	Administrador(inicia)
Propósito	Mostrar el hdd tanto de una PC, como de un rango de IP o de todas las PC en el sistema.
Resumen	El caso de uso se inicia cuando Administrador selecciona <i>Consulta</i> y luego <i>Mostrar</i> hay le saldrá donde poner el IP de la maquina la cual quiere ver la red

o para determinar si desea un rango de IP o de todas las PC. Luego chequea que desea ver red y presiona OK.	
Referencias	R53,R74,R75,R76,R77,R78,R79
Precondiciones	Debe haber alguna PC en el sistema.
Postcondiciones	Muestra los datos de la red de las PC en el sistema.
Requisitos Especiales	
Prototipo	Ver Anexo B.12

Tabla 3.20: Descripción del caso de uso de sistema Mostrar Red.

Caso de uso	Mostrar Ram
Actores	Administrador(inicia)
Propósito	Mostrar la ram y memoria virtual tanto de una PC, como de un rango de IP o de todas las PC en el sistema.
Resumen El caso de uso se inicia cuando Administrador selecciona <i>Consulta</i> y luego <i>Mostrar</i> hay le saldrá donde poner el IP de la maquina la cual quiere ver la ram y la memoria virtual o para determinar si desea un rango de IP o de todas las PC. Luego chequea que desea ver ram y presiona OK.	
Referencias	R53,R80,R81,R82,R83,R84,R85
Precondiciones	Debe haber alguna PC en el sistema.
Postcondiciones	Muestra los datos de la ram y la memoria virtual de las PC en el sistema.
Requisitos Especiales	
Prototipo	Ver Anexo B.13

Tabla 3.21: Descripción del caso de uso de sistema Mostrar Ram.

Caso de uso	Mostrar Reportes
Actores	Administrador(inicia)
Propósito	Realizar los reportes para luego imprimirlo.
Resumen El caso de uso se inicia cuando Administrador selecciona <i>Reporte</i> y luego <i>Reportes</i> hay le saldrá donde poner el rango de IP del cual desea ver los reportes o para seleccionar el local. Luego chequeas lo que desees ver en el reporte y presiona OK.	
Referencias	R87,R88,R89
Precondiciones	
Postcondiciones	Emite los reportes necesarios para la facultad
Requisitos Especiales	
Prototipo	Ver Anexo B.14

Tabla 3.22: Descripción del caso de uso de sistema Emitir Reportes.

3.4— Construcción del sistema.

3.4.1— Diagrama de clases del diseño.

Un Diagrama de Clases de Diseño muestra la especificación para las clases de una aplicación. Incluye la siguiente información:

- Clases, asociaciones y atributos.
- Interfaces, con sus operaciones y constantes.
- Métodos.
- Navegabilidad.
- Dependencias.

A diferencia del Modelo Conceptual, un Diagrama de Clases de Diseño muestra definiciones de entidades software más que conceptos del mundo real. [6]

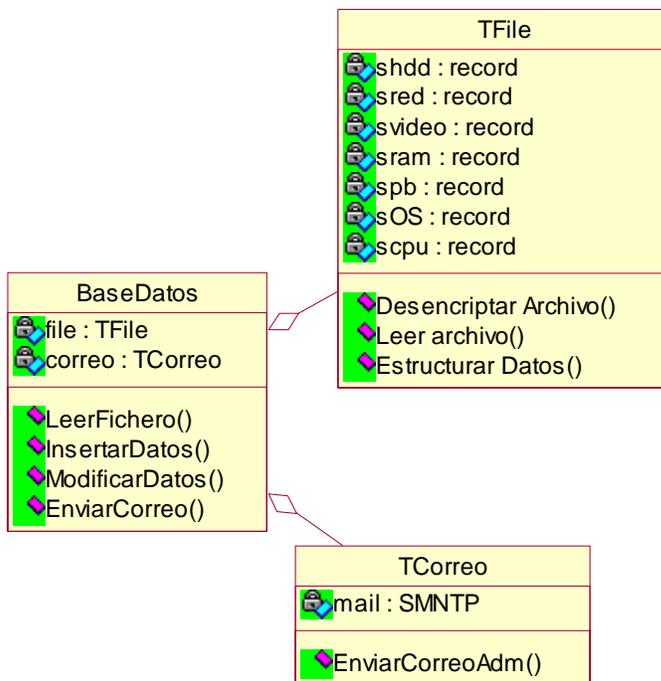


Figura 5. Diagrama de clases del diseño perteneciente al monitor (servidor).

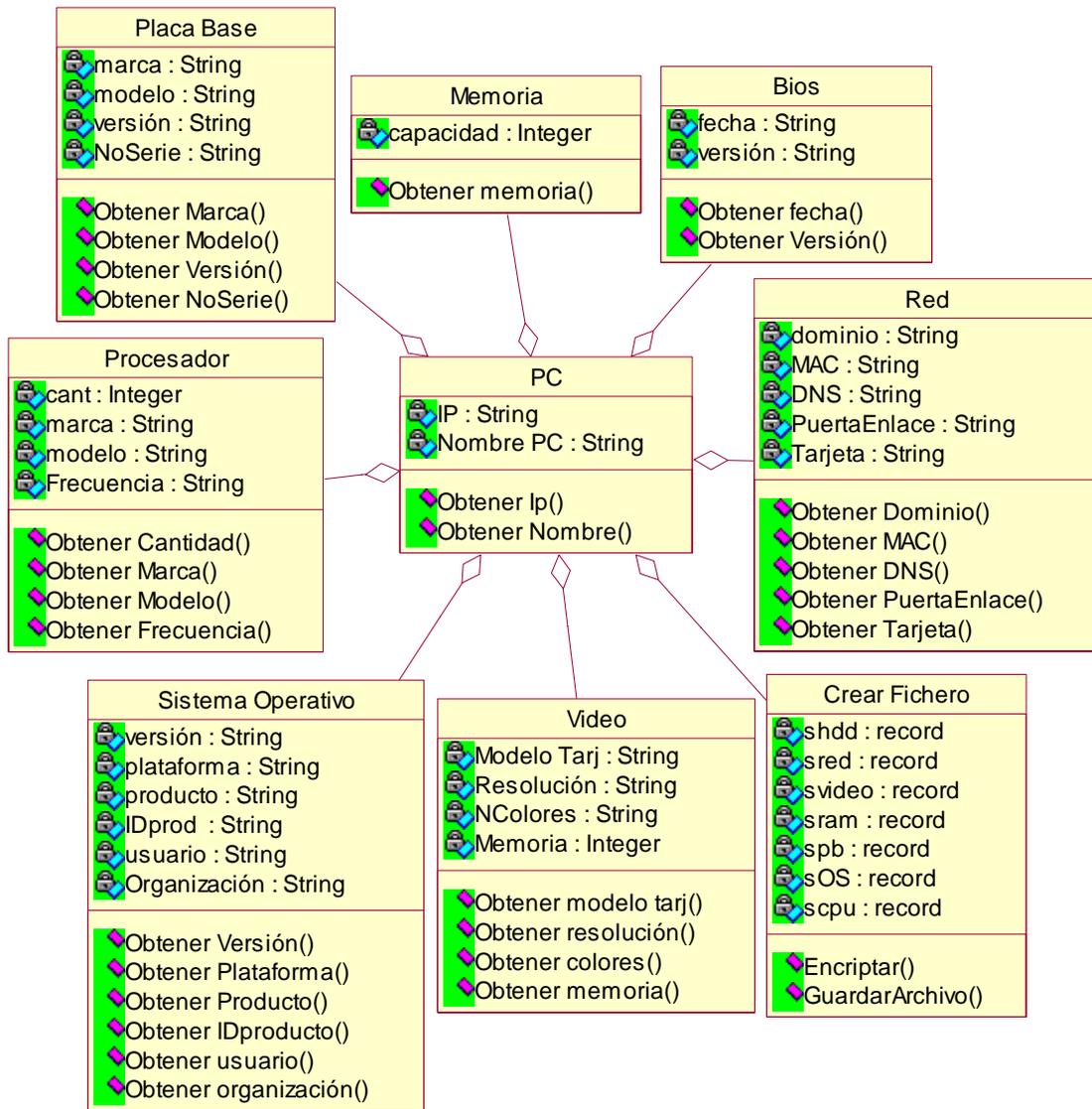


Figura 6. Diagrama de clases del diseño perteneciente al agente (clientes).

3.4.2— Diagrama del modelo lógico de datos.

El modelo lógico de la base de datos determina cómo se estructuran los datos de forma lógica mediante tablas y relaciones. Este diseño puede tener también una gran repercusión en el rendimiento de la aplicación. [7]

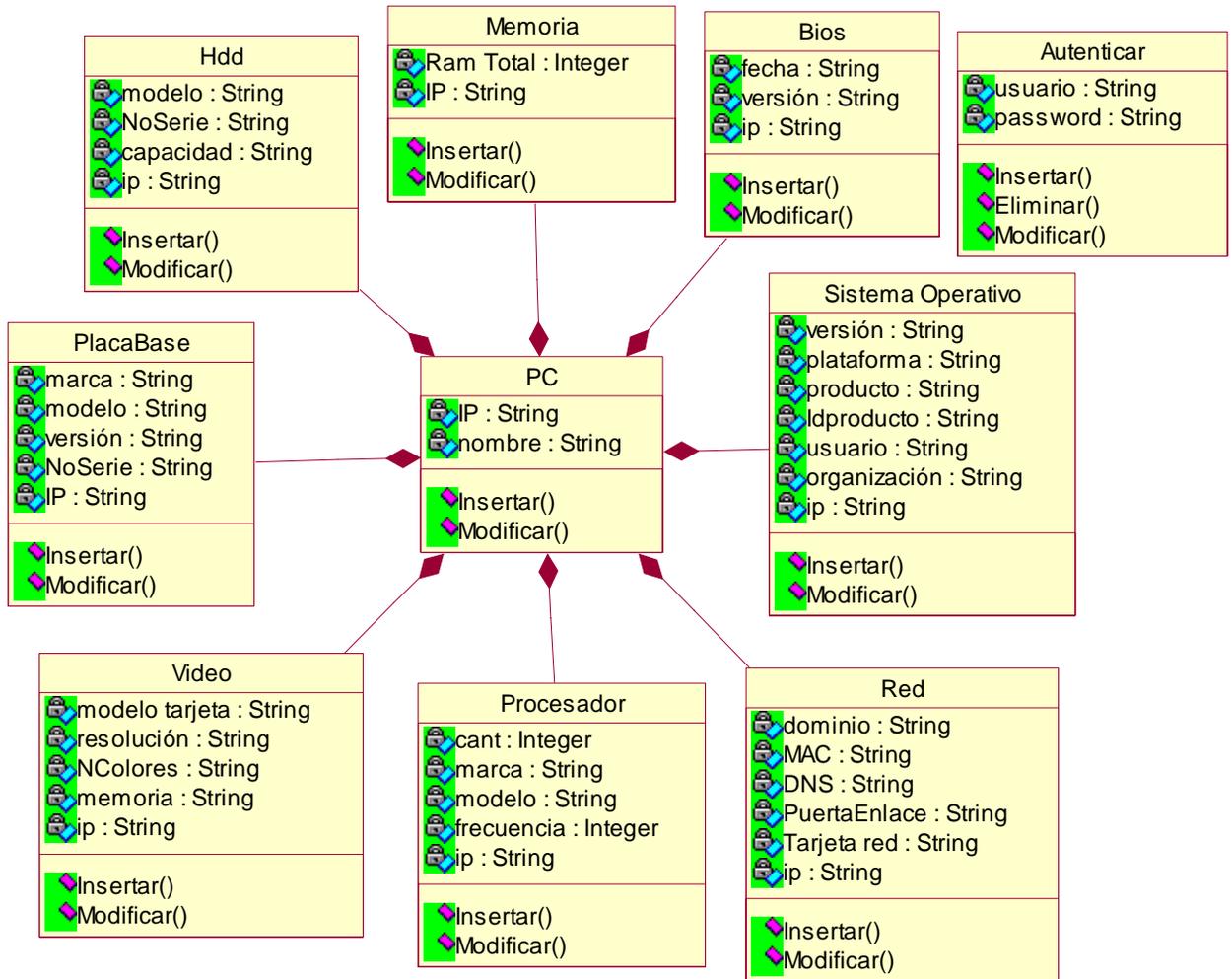


Figura 7. Diagrama modelo lógico de datos.

3.4.3— Diagrama del modelo físico de datos.

El modelo físico de datos incluye todos los aspectos de diseño de un modelo de base de datos que se pueden modificar sin cambiar los componentes de la aplicación. [7]

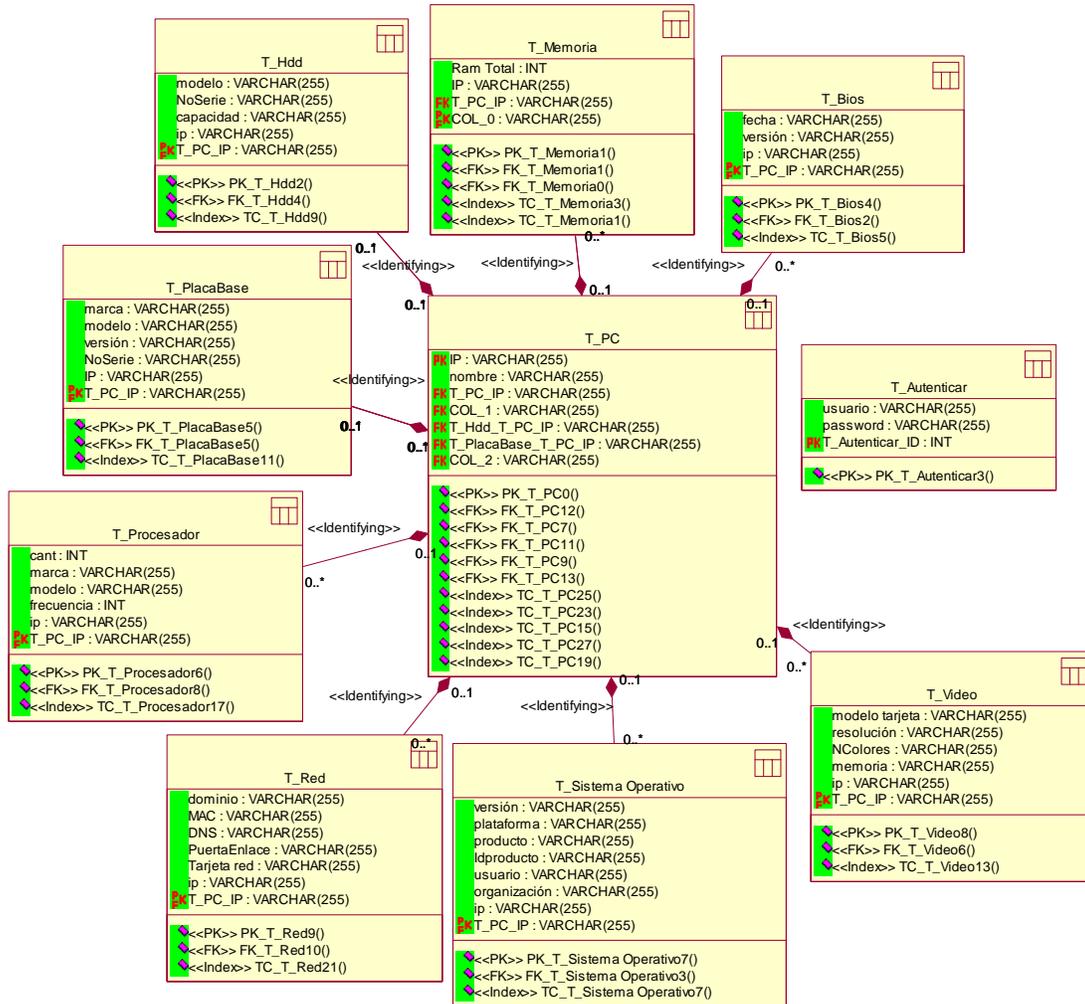


Figura 8. Diagrama del modelo físico de datos.

3.4.4— Diagramas de implementación.

El modelo de implementación describe la forma en que los elementos del modelo de diseño, como las clases, se implementan en términos de componentes. Describe también como se organizan los componentes de acuerdo con los mecanismos de estructuración y modularización disponibles en el entorno de implementación y en el lenguaje o lenguajes de programación utilizados y como dependen los componentes unos de otros. [1]

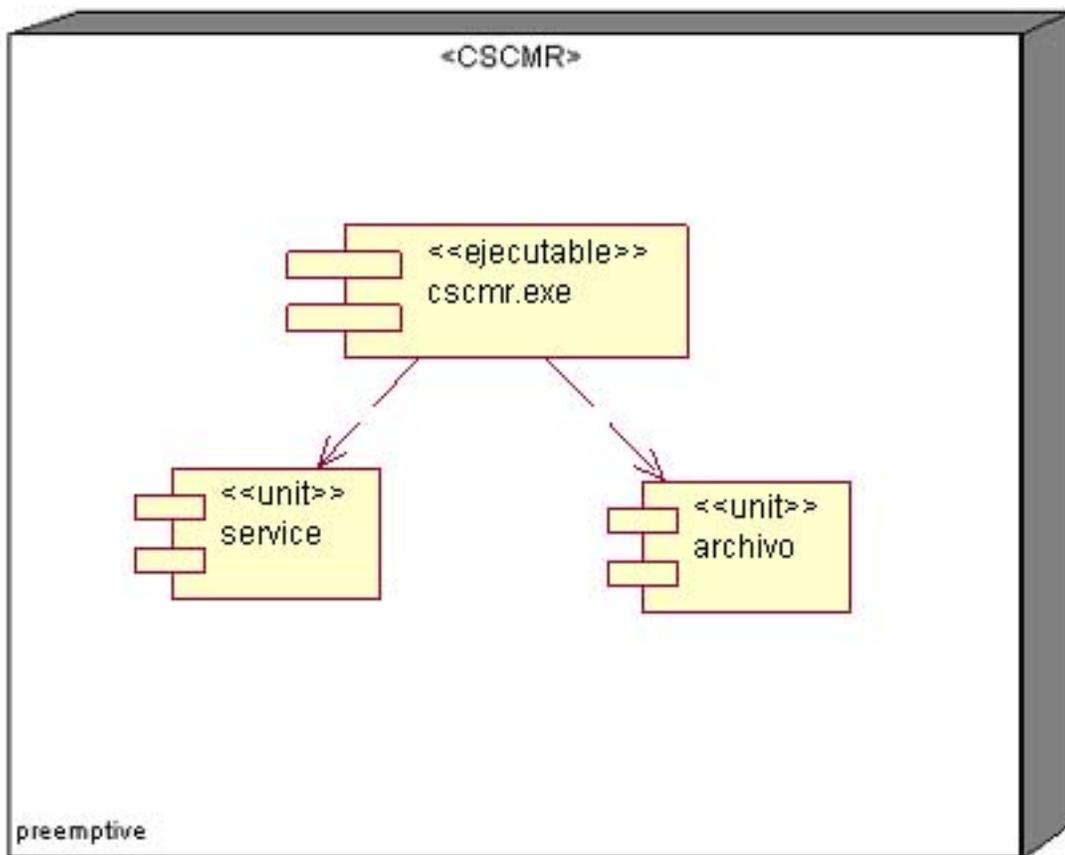


Figura 9. Diagrama de implementación del agente.

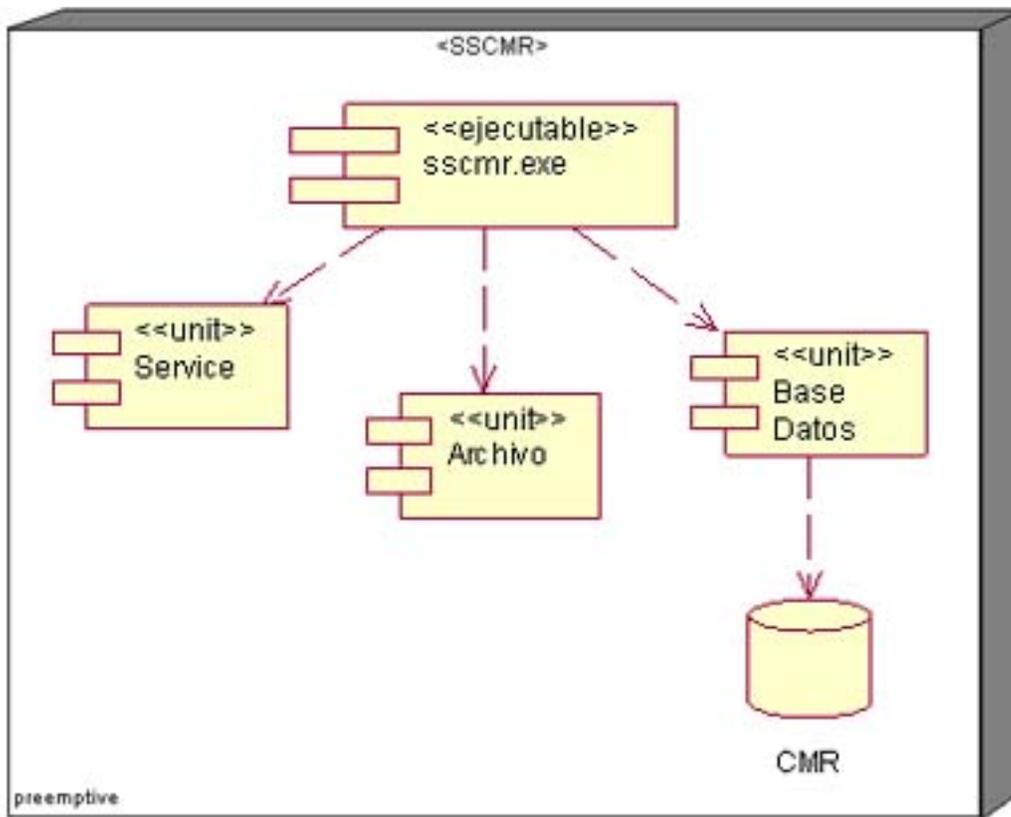


Figura 10. Diagrama de implementación del monitor.

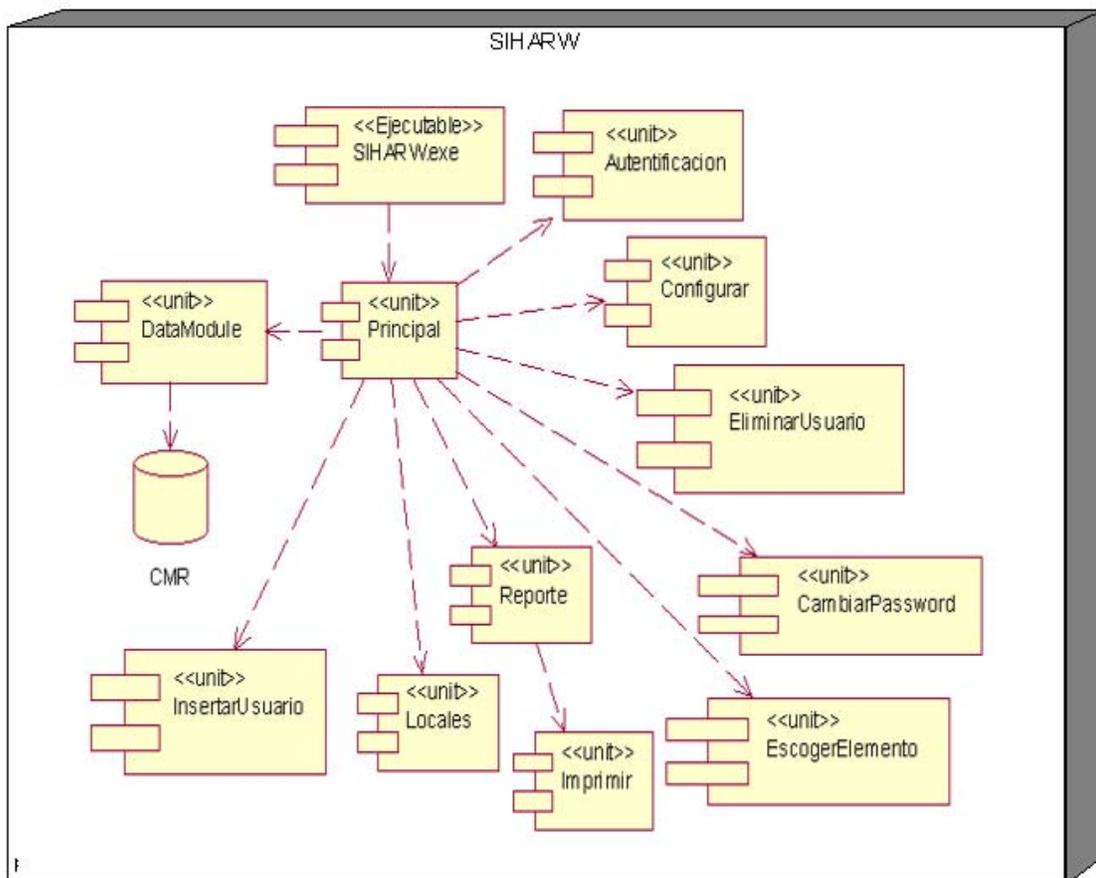


Figura 11. Diagrama de implementación de la consola.

3.5— Principios de diseño del sistema.

3.5.1— Diseño de la interfaz.

La interfaz fue diseñada basada en el estándar de ventanas de Windows. El tipo de letra utilizada fue MS Sans Serif de estilo Normal y tamaño de 8 pts. La carga visual esta concebida para especialistas en la materia, para los cuales las opciones de acceso se han hecho de forma muy familiar teniendo en cuenta los objetivos buscados por los usuarios. Para el acceso rápido a las opciones del sistema se cuenta con una barra de herramientas con la mayor parte de las opciones de la aplicación.

3.5.2— Concepción general de la ayuda.

El sistema cuenta con una ayuda general para el usuario, donde se explica detalladamente las opciones que brindan la aplicación.

3.5.3— Tratamiento de errores.

El sistema está concebido para minimizar los errores que puedan ser introducidos por los usuarios a la hora de manipular la información.

3.5.4— Estándares de codificación.

Los nombres de las variables, los controles, los procedimientos y funciones fueron adoptados lo más explicativos posibles, siempre respondiendo a su propósito. Para cada tipo de control se tuvo en cuenta el uso de los prefijos más utilizados, una muestra de esto se muestra en la siguiente tabla:

Control	Prefijo	Ejemplo
Button	Btn	btnCancelar
Form	Frm	frmVehiculos
Menu	Mn	mnPrincipal
LabelEdit	Le	leMarca
ToolButton	Tb	tbSolape
ListBox	Ltb	lbListaVehiculos
Timer	Tm	tmPrincipal
ToolBar	Tb	tbCicloBasico
SkinData	Kd	kdPrincipal
ActionList	Al	AlPrincipal

3.6 – Conclusiones

Como resultado de este capítulo se describió el modelo de sistema del objeto de automatización sobre la base de las especificaciones de la metodología RUP. Se identificaron los requerimientos funcionales y no funcionales, se definieron los actores del sistema y los servicios o funcionalidades que a su disposición se colocan (los casos de uso del sistema). Además, se detallaron una serie de diagramas que colaboran con la implementación del modelo de sistema, como son: el diagrama de casos de uso del sistema, el diagrama de clases del diseño, el diagrama del modelo físico y lógico de datos y el diagrama de implementación.

Capítulo IV – Estudio de Factibilidad

En el presente capítulo se aborda el tema relativo al estudio de la factibilidad del producto, se ofrece una descripción de la planificación del proyecto, así como los costos asociados al mismo, los beneficios tangibles e intangibles que reportaría su elaboración y finalmente el análisis entre los costos y los beneficios para concluir si es o no factible el desarrollo del sistema.

4.1 – Planificación

En el desarrollo de este capítulo utilizamos el método de Puntos de Características para la estimación del esfuerzo, el tiempo de desarrollo y el costo del proyecto.

Para realizar el cálculo de los costos de desarrollo del sistema se deben obtener primero las instrucciones fuentes. Analizándose para esto las cantidades de entradas, salidas, peticiones, archivos lógicos e interfaces externas preliminares que tiene el sistema. Para calcular la cantidad de instrucciones fuentes hay que tener en cuenta también que la conversión al Delphi y MySQL lenguajes seleccionados para implementar la aplicación, es de 29 y 37 puntos respectivamente.

Después de este estudio se llegó a los siguientes resultados:

Entrada Externa	Cantidad Ficheros	Cantidad elementos datos	Clasificación
Configuración	1	2	Bajo
Insertar Locales	1	3	Bajo
Actualizar Locales	1	3	Bajo
Eliminar Locales	1	3	Bajo
Insertar Usuario	1	2	Bajo

Eliminar Usuario	1	2	Bajo
Cambiar Contraseña	1	3	Bajo
Insertar PC	1	4	Bajo
Insertar Bios	1	3	Bajo
Insertar HDD	1	4	Bajo
Insertar PlacaBase	1	5	Bajo
Insertar Procesador	1	5	Bajo
Insertar Ram	1	2	Bajo
Insertar Red	1	7	Bajo
Insertar Sistema Operativo	1	7	Bajo
Insertar Video	1	5	Bajo
Insertar Alarma	1	9	Bajo

Tabla 4.1 Entradas externas

Salida externa	Cantidad ficheros	Cantidad Elementos datos	Clasificación
Mostrar Procesador	1	5	Bajo
Mostrar Placabase	1	5	Bajo
Mostrar Bios	1	3	Bajo
Mostrar Video	1	5	Bajo
Mostrar HDD	1	4	Bajo
Mostrar Red	1	7	Bajo
Mostrar Ram	1	2	Bajo
Mostrar reporte de procesador	1	2	Bajo
Mostrar reporte de placabase	1	2	Bajo
Mostrar reporte de disco duro	1	2	Bajo
Mostrar reporte de procesador y placabase	2	3	Bajo
Mostrar reportes de procesador y	2	3	Bajo

disco duro			
Mostrar reportes de placabase y disco duro	2	3	Bajo
Mostrar reporte de procesador, placabase y disco duro	3	4	Bajo

Tabla 4.2 Salidas externas

Petición	Cantidad ficheros	Cantidad Elementos datos	Clasificación
Imprimir reporte de procesador	1	2	Bajo
Imprimir reporte de placabase	1	2	Bajo
Imprimir reporte de disco duro	1	2	Bajo
Imprimir reporte de procesador y placabase	2	3	Bajo
Imprimir reporte de procesador y disco duro	2	3	Bajo
Imprimir reporte de placabase y disco duro	2	3	Bajo
Imprimir reporte de procesador, placabase y disco duro	3	4	Bajo

Tabla 4.3 Peticiones

Nombre del fichero interno	Cantidad ficheros	Cantidad Elementos datos	Clasificación
Alarma	1	9	Bajo
Bios	1	3	Bajo
HDD	1	4	Bajo
PC	1	4	Bajo

Placabase	1	5	Bajo
Procesador	1	5	Bajo
Ram	1	2	Bajo
Red	1	7	Bajo
Sistema Operativo	1	7	Bajo
Usuario	1	2	Bajo
Video	1	5	Bajo
Usuarios	1	4	Bajo

Tabla 4.4 Ficheros lógicos internos

Elementos	Bajos	X Peso	Medios	X Peso	Altos	X Peso	Subtotal puntos función
Ficheros lógicos internos	14	7	0	10	0	15	98
Entradas externas	17	3	0	4	0	6	51
Salidas externas	14	4	0	5	0	7	56
Peticiones	7	3	0	4	0	16	21
Total							226

Tabla 4.5 Puntos de función

Características	Valor	
Puntos de función desajustados	226	
Lenguaje	MySQL	Delphi
Instrucciones fuentes por puntos de función	37	29

Por ciento de la aplicación en cuanto a requerimientos funcionales	30%	70%
Instrucciones fuentes	2508,6	4587,8
Total de Instrucciones fuentes		7096.4

Tabla 4.6 Miles de Instrucciones fuentes

4.2 – Costos.

Cálculo del esfuerzo, tiempo de desarrollo, cantidad de hombres y costo.

Factores de escala

Cálculo de:	Valor	Justificación
RCPX	1,00	BD moderada, no se requiere de amplia documentación. La aplicación Web tiene una moderada complejidad. (Nominal)
RUSE	1,00	Se implementa código reutilizable para el aprovechamiento de este en toda la aplicación. (Nominal)
PDIF	1,00	No tiene grandes restricciones en cuanto al tiempo de ejecución ya que el software podrá estar trabajando varias horas. EL Software no tiene limitación de memoria impuesta. La plataforma de aplicación tiene gran estabilidad. (Nominal)
PERS	0,70	Hay poco movimiento del personal. (Alto)
PREX	0,84	El equipo tiene buen dominio y posee conocimiento del lenguaje de programación. Con una experiencia de aproximadamente un año. (Alto)
FCIL	0,87	Se utilizan herramientas de programación como: Macromedia Dreamweaver 2004, Zend Development Studio, así como la herramienta CASE Rational Rose para la documentación, empleando como notación UML. (Alto)
SCED	1,00	La planificación se hace con moderada frecuencia.

		(Nominal)
PREC	3,72	El equipo de desarrollo posee una comprensión considerable de los objetivos del producto, no tiene experiencia en la realización de software de este tipo. (Nominal)
FLEX	3,04	El sistema cuenta con alguna flexibilidad en relación con las especificaciones de los requerimientos preestablecidos y a las especificaciones de interfaz externa. (Nominal)
TEAM	1,10	El equipo que va a desarrollar el software es altamente cooperativo.
RESL	4,24	Teniendo en cuenta la alta experiencia que existe en el país acerca de este tipo de estudios existen algunos factores de riesgo. (Nominal)
PMAT	6,24	Nivel I Alto porque se encuentra en su primera etapa un poco avanzada. (Bajo)

Multiplicador de esfuerzos

$$EM = \prod_{i=1}^7 E_{mi} = RCPX * RUSE * PDIF * PERS * PREX * FCIL * SCED$$

$$EM = \prod_{i=1}^7 E_{mi} = 1,00 * 1,00 * 1,00 * 0,70 * 0,84 * 0,87 * 1,00 = 0,51156 \approx \mathbf{0,51}$$

Factores de escala

$$SF = \sum SF_i = PREC + FLEX + RESL + TEAM + PMAT$$

$$SF = \sum SF_i = 3,72 + 3,04 + 4,24 + 1,10 + 6,24 = \mathbf{18,34}$$

Valores de los coeficientes

$$A = 2,94; B = 0,91; C = 3,67; D = 0,24$$

$$E = B + 0,01 * SF$$

$$E = 0,91 + 0,01 * 18,34$$

$$E = 1,0934$$

$$F = D + 0,2 * (E - B)$$

$$F = 0,24 + 0,2 * (1,0934 - 0,91)$$

$$F = 0,27668$$

Esfuerzo

$$PM = A * (MF)^E * EM$$

$$PM = 2,94 * (7,0964)^{1,0934} * 0,51$$

$$PM = 12,78(\text{personas meses})$$

Cálculo del tiempo de desarrollo

$$TDEV = C * PM^F$$

$$TDEV = 3,67 * (12,78)^{0,27668}$$

$$TDEV = 7,43$$

Cálculo de la cantidad de hombres

$$CH = PM / TDEV$$

$$CH = 12,78 / 7,43$$

$$CH = 1,72$$

Recalculando

$$CH = PM / TDEV$$

$$2 = 12,78 / TDEV$$

$$TDEV = 6,38$$

$$TDEV \approx 6 \text{ meses} = \frac{1}{2} \text{ año}$$

Costo

Se asume como salario promedio mensual 275\$

$$CHM = 2 * \text{Salario Promedio}$$

$$CHM = 2 * 275$$

$$CHM = 550 \text{ \$/mes}$$

$$\text{Costo} = CHM * PM$$

$$\text{Costo} = 550 * 6$$

$$\text{Costo} = 3300$$

Los costos en los que se incurriría de desarrollarse el sistema serían:

Cálculo de:	Valor
Esfuerzo(PM)	12,78
Tiempo de desarrollo	6 meses
Cantidad de hombres	2

Costo	\$3300
Salario medio	\$275,0
RCPX	1,00
RUSE	1,00
PDIF	1,00
PREX	0,84
FCIL	0,87
SCED	1,00

4.3 – Análisis de costos y beneficios.

El desarrollo de un producto informático tiene asociado un costo y el llevarlo a cabo o no, está en dependencia de los beneficios que el mismo pudiese reportar. Los beneficios pueden ser económicos o de orden social.

La herramienta de software que se propone está dirigida fundamentalmente a proteger el hardware de las computadoras y a gestionar los reportes de la red de la facultad de Informática, por tanto su mayor beneficio es de orden económico.

El software reportará los siguientes beneficios:

- Toda la información del hardware de las máquinas quedará registrada en una Base de Datos.
- Los administradores serán alertados en caso de cambio o pérdida del hardware.
- Se agiliza la entrega de reportes.

Desde el punto de vista económico, se puede decir que este software traería consigo un considerable ahorro de dinero por concepto de compra de un producto extranjero.

4.4 – Conclusiones

La herramienta propuesta trae consigo una serie de beneficios sobre todo intangibles para la organización, pero no menos necesarios e importantes, porque va a contribuir a mejorar su funcionamiento, lo que indica que es factible implementar la herramienta propuesta. Una vez terminado el estudio de factibilidad del sistema, se estima un tiempo de 6 meses para su construcción por 2 hombres y su costo asciende a \$3300.

Conclusiones

Para culminar esta investigación y teniendo en cuenta los objetivos planteados arribamos a las siguientes conclusiones:

- Se investigaron los sistemas vinculados al campo de acción.
- Se creó una Base de Datos que garantiza el almacenamiento, la seguridad e integridad de la información.
- Se desarrolló una herramienta que de forma automática realiza el inventario del hardware de las computadoras de la red, alerta a los administradores de posibles cambios y realiza los reportes necesarios que deben ser entregados a la dirección de la Facultad.
- Se sometió el software elaborado a un período de prueba para la detección de posibles errores logrando una buena satisfacción del usuario con el resultado final del trabajo.

Recomendaciones

A pesar de que el software desarrollado cumple con los objetivos trazados recomendamos:

1. Dar continuidad al proyecto implementando una nueva versión desatendida, que desde el propio servidor puedan ser analizadas todas las máquinas de la red sin necesidad de instalar un cliente.
2. Lograr que en nuevas versiones se puedan monitorear computadoras con el sistema operativo Linux.

Referencias Bibliográficas

- [1] Jacobson, I. El Proceso Unificado de Desarrollo de software / Ivar Jacobson, G. Booch, J. Rumbaugh. —Addison-Wesley: [s.n], 2000. —p.25-38
- [2] Letelier Torres, Patricio. Desarrollo de Software Orientado a Objeto usando UML. Tomado De: <http://www.creangel.com/uml/creditos.php>, 11 de marzo de 2008.
- [3] Ruiz Bravo, Danaysi. MacoSoft, Software para la elaboración de Mapas Conceptuales.—Trabajo de Diploma, Instituto Superior Politécnico “José A. Echeverría” (Ch), 2004.— h.11
- [4] Rodríguez Jiménez, Damián. Segas, Software para el análisis y diseño de un sistema para el control del servicio de protección con agentes de seguridad de la agencia territorial SEPSA Cienfuegos.— Trabajo de Diploma Universidad de Cienfuegos (Cf), 2007.— h.26
- [5] Peraza Bello, Demis. SECDC, Sistema de Evaluación de Cualidades Dinámicas y de Consumo.— Trabajo de Diploma Universidad de Cienfuegos (Cf), 2006.— h.38
- [6] Comunidad de desarrolladores. Diagrama de Clases de Diseño. Tomado De: <http://www.clikear.com/manuales/uml/index.aspx> , 6 de abril del 2008.
- [7] Microsoft Corporation. Todos los derechos reservados. Tomado De: <http://www.microsoft.com/latam/technet/articulos/idc/idc5/default.asp#> ,11 de marzo del 2008.

Bibliografía

- ✓ Diagrama de Clases de Diseño. Tomado De:
<http://www.clikear.com/manuales/uml/index.aspx> , 6 de abril del 2008.
- ✓ Diagramas de clases: Definición y usos. Tomado De:
<http://www.osmosislatina.com/lenguajes/uml/clasesob.htm> , 2 de mayo del 2008.
- ✓ Jacobson, I. El Proceso Unificado de Desarrollo de software / Ivar Jacobson, G. Booch, J. Rumbaugh. —Addison-Wesley: [s.n], 2000. —356p.
- ✓ Letelier Torres, Patricio. Desarrollo de Software Orientado a Objeto usando UML. Tomado De: <http://www.creangel.com/uml/creditos.php>, 11 de marzo de 2008.
- ✓ Mendoza Sánchez, María A. Metodologías de Desarrollo de Software. Tomado De:http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_software_07062004.html ,2 de mayo del 2008.
- ✓ Rodríguez Jiménez, Damián. Segas, Software para el análisis y diseño de un sistema para el control del servicio de protección con agentes de seguridad de la agencia territorial sepsa Cienfuegos / Damián Rodríguez Jiménez; Laura Toledo Diez, Daimarelys Acevedo Cardoso, Reynel Domínguez Domínguez, tutores.— Trabajo de Diploma Universidad de Cienfuegos (Cf), 2007.— 135 h.
- ✓ Ruiz Bravo, Danaysi. MacoSoft, Software para la elaboración de Mapas Conceptuales / Danaysi Ruiz Bravo, Boris Piñero Suárez; Alfredo J. Simón Cuevas, tutor.—Trabajo de Diploma, Instituto Superior Politécnico “José A. Echeverría” (Ch), 2004.— 115 h.

Glosario de Términos

B

Benchmarks: En términos de ordenadores, un benchmark podría ser realizado en cualquiera de sus componentes, ya sea CPU, RAM, tarjeta gráfica, etc. También puede ser dirigido específicamente a una función dentro de un componente, por ejemplo, la unidad de coma flotante de la CPU; o incluso a otros programas.

Bios: sistema Básico de entrada/salida Basic Input-Output System es un código de interfaz que localiza y carga el sistema operativo en la RAM; es un software muy básico instalado en la placa base que permite que ésta cumpla su cometido. Proporciona la comunicación de bajo nivel, y el funcionamiento y configuración del hardware del sistema que, como mínimo, maneja el teclado y proporciona salida básica (emitiendo pitidos normalizados por el altavoz de la computadora si se producen fallos) durante el arranque.

D

DNS: Domain Name System es una base de datos distribuida y jerárquica que almacena información asociada a nombres de dominio en redes como Internet. Aunque como base de datos el DNS es capaz de asociar diferentes tipos de información a cada nombre, los usos más comunes son la asignación de nombres de dominio a direcciones IP y la localización de los servidores de correo electrónico de cada dominio.

F

Freeware: define un tipo de software de computadora que se distribuye sin costo y por tiempo ilimitado. El freeware suele incluir una licencia de uso, que permite su redistribución pero con algunas restricciones, como no modificar la aplicación en sí, ni venderla, y dar cuenta de su autor. También puede desautorizar el uso en una compañía con fines comerciales o en una entidad gubernamental.

FTP: *File Transfer Protocol* es un protocolo de transferencia de archivos entre sistemas conectados a una red TCP basado en la arquitectura cliente-servidor, de

manera que desde un equipo cliente nos podemos conectar a un servidor para descargar archivos desde él o para enviarle nuestros propios archivos independientemente del sistema operativo utilizado en cada equipo.

H

Host: Un ordenador de la red que ofrece servicios a otros equipos o una máquina conectada a una red de ordenadores y que tiene un nombre de equipo.

Hub: En informática un hub o concentrador es un equipo de redes que permite conectar entre sí otros equipos y retransmite los paquetes que recibe desde cualquiera de ellos a todos los demás. Los hubs han dejado de ser utilizados, debido al gran nivel de colisiones y tráfico de red que propician.

Hardware: es la parte física de un computador y más ampliamente de cualquier dispositivo electrónico. El hardware se refiere a todos los componentes físicos (que se pueden tocar), en el caso de una computadora personal serían los discos, unidades de disco, monitor, teclado, la placa base, el microprocesador.

I

IP: Protocolo de Internet (siglas en Inglés). Uno de los protocolos más representativos del estándar TCP/IP, es el responsable del esquema de direccionamiento utilizado en Internet y define la estructura y el intercambio de los datagramas IP entre redes distantes.

L

LAN: Local Area Network (siglas en inglés). Redes de área local.

M

MySQL: Lenguaje estructurado de consulta (siglas en Inglés). Lenguaje de programación que se utiliza para recuperar y actualizar la información contenida en una base de datos.

MHTML: (Abreviado también **MHT**) significa MIME HTML (Multipurpose Internet Mail Extension HTML o en español protocolo de transferencia de hipertexto Multiuso de la

Extensión del Correo del Internet). Es un estándar para incluir recursos que en páginas HTTP usualmente están enlazados externamente, tal como los archivos de imágenes y sonido, en el mismo archivo como el Código de Protocolo de Transferencia de Hipertexto.

O

Overclocking: es una técnica popular para conseguir un aumento en el funcionamiento de un sistema. En muchos casos puedes forzar tu CPU para funcione más rápido simplemente mediante un puente en la placa base. El Overclocking, sin embargo, tiene algunos riesgos como el recalentamiento, así que debes averiguar todos los pros y los contras antes de hacerlo.

P

PC: Personal Computer (en inglés). Computadora Personal.

R

RUP: Proceso Unificado de Desarrollo (Metodología de desarrollo de software).

Router: ruteador o encaminador es un dispositivo de hardware para interconexión de red de computadoras que opera en la capa tres (nivel de red). Este dispositivo permite asegurar el enrutamiento de paquetes entre redes o determinar la ruta que debe tomar el paquete de datos.

RAM: Random Access Memory, o más conocida como memoria RAM, se compone de uno o más chips y se utiliza como memoria de trabajo para programas y datos. Es un tipo de memoria temporal que pierde sus datos cuando se queda sin energía (por ejemplo, al apagar la computadora), por lo cual es una memoria volátil.

RFC: Request For Comments —petición de comentarios— son una serie de notas sobre Internet. Cada una de ellas individualmente es un documento cuyo contenido es una propuesta oficial para un nuevo protocolo de la red Internet (originalmente de ARPANET), que se explica con todo detalle para que en caso de ser aceptado pueda ser implementado sin ambigüedades. Cualquiera puede enviar una propuesta de RFC a la IETF, pero es ésta la que decide finalmente si el documento se convierte en

una RFC o no. Si luego resulta lo suficientemente interesante, puede llegar a convertirse en un estándar de Internet. Cada RFC tiene un título y un número asignado, que no puede repetirse ni eliminarse aunque el documento se quede obsoleto. Cada protocolo de los que hoy existen en Internet tiene asociado un RFC que lo define, y posiblemente otros RFC adicionales que lo amplían. Por ejemplo el protocolo IP se detalla en el RFC 791, el FTP en el RFC 959, y el HTTP (escrito por Tim Berners-Lee, entre otros) el RFC 2616.

S

Shareware: a una modalidad de distribución de software, tanto como juegos como programas utilitarios, para que el usuario pueda evaluar de forma gratuita el producto, por un tiempo especificado. Para adquirir una licencia de software que permite el uso del software de manera completa se requiere de un pago.

Software: Se refiere al **equipamiento lógico** o **soporte lógico** de un computador digital, comprende el conjunto de los componentes lógicos necesarios para hacer posible la realización de una tarea específica, en contraposición a los componentes físicos del sistema (hardware). Tales componentes lógicos incluyen, entre otras, aplicaciones informáticas tales como procesador de textos, que permite al usuario realizar todas las tareas concernientes a edición de textos; software de sistema, tal como un sistema operativo, el que, básicamente, permite al resto de los programas funcionar adecuadamente, facilitando la interacción con los componentes físicos y el resto de las aplicaciones, también provee una interface ante el usuario.

Switch: (en castellano "conmutador") es un dispositivo electrónico de interconexión de redes de ordenadores que opera en la capa 2 (nivel de enlace de datos) del modelo OSI (*Open Systems Interconnection*). Un conmutador interconecta dos o más segmentos de red, funcionando de manera similar a los puentes (bridges), pasando datos de un segmento a otro, de acuerdo con la dirección MAC de destino de los datagramas en la red.

Servicio de Windows: es una aplicación que está pensada para ejecutarse siempre que el ordenador está encendido, aunque el usuario no esté conectado.

Socket: es un método para la comunicación entre un programa del cliente y un programa del servidor en una red, una dirección de Internet, combinando una dirección IP (la dirección numérica única de cuatro partes que identifica a un ordenador particular en Internet) y un número de puerto (el número que identifica una aplicación de Internet particular, como FTP, Gopher, o WWW).

SIHARW: Sistema para Inventariar el Hardware de las Computadoras desde Windows.

SMTP: Protocolo Simple de Transferencia de Correo electrónico (siglas en Inglés). Protocolo de red basado en texto utilizado para el intercambio de mensajes de correo electrónico entre computadoras y/o distintos dispositivos.

SQL: Lenguaje estructurado de consulta (siglas en Inglés). Lenguaje de programación que se utiliza para recuperar y actualizar la información contenida en una base de datos.

SGBD: Sistema Gestor de base de datos, 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.

Así se trata de un software de propósito general. Ejemplo de SGBD son Oracle y SQL Server de Microsoft.

Service Pack: Grupo de parches que actualizan, corrigen y mejoran aplicaciones y sistemas operativos. Esta denominación fue popularizada por Microsoft cuando comenzó a empaquetar grupos de parches que actualizaban su sistema operativo Windows.

T

TCP/IP: Protocolo de Control de Transmisión/Protocolo de Internet (siglas en Inglés). Convenio de los protocolos de aplicación y transporte que se utiliza en el IP. Incluye FTP, TELNET, SMTP y UDP.

TIC: Tecnologías de la Informática y la Comunicaciones.

Token: Arquitectura de red desarrollada por IBM en los años 70's con topología lógica en anillo y técnica de acceso de paso de testigo. Token Ring se recoge en el estándar IEEE 802.5. En desuso por la popularización de Ethernet; no obstante, determinados escenarios, tales como bancos, siguen empleándolo.

U

UCF: Universidad de Cienfuegos.

UDP: User Datagram Protocol. Protocolo de nivel de transporte estándar de Internet. Añade flexibilidad al IP al no estar orientado a la conexión, puesto que se envía el paquete y no se espera confirmación.

UML: Lenguaje Unificado de Modelado (siglas en Inglés).

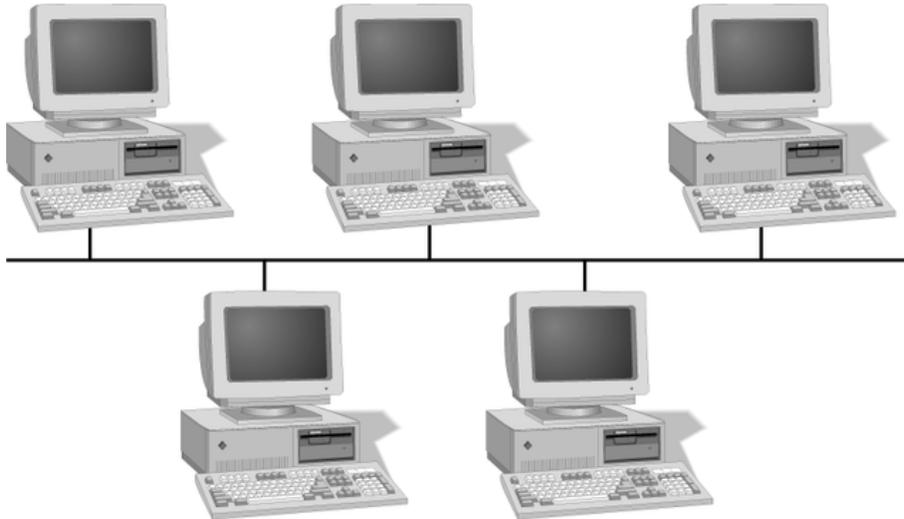
X

XML: sigla en inglés de *Extensible Markup Language* («lenguaje de marcas extensible»), es un metalenguaje extensible de etiquetas desarrollado por el World Wide Web Consortium (W3C). Es una simplificación y adaptación del SGML y permite definir la gramática de lenguajes específicos. Por lo tanto XML no es realmente un lenguaje en particular, sino una manera de definir lenguajes para diferentes necesidades. Algunos de estos lenguajes que usan XML para su definición son XHTML, SVG, MathML.

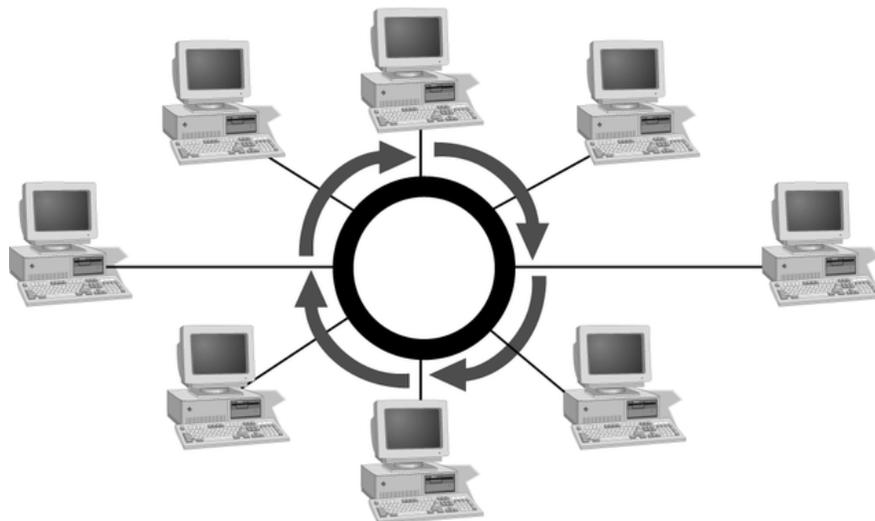
Anexos

ANEXO A. DOCUMENTACION ASOCIADA AL OBJETO DE ESTUDIO.

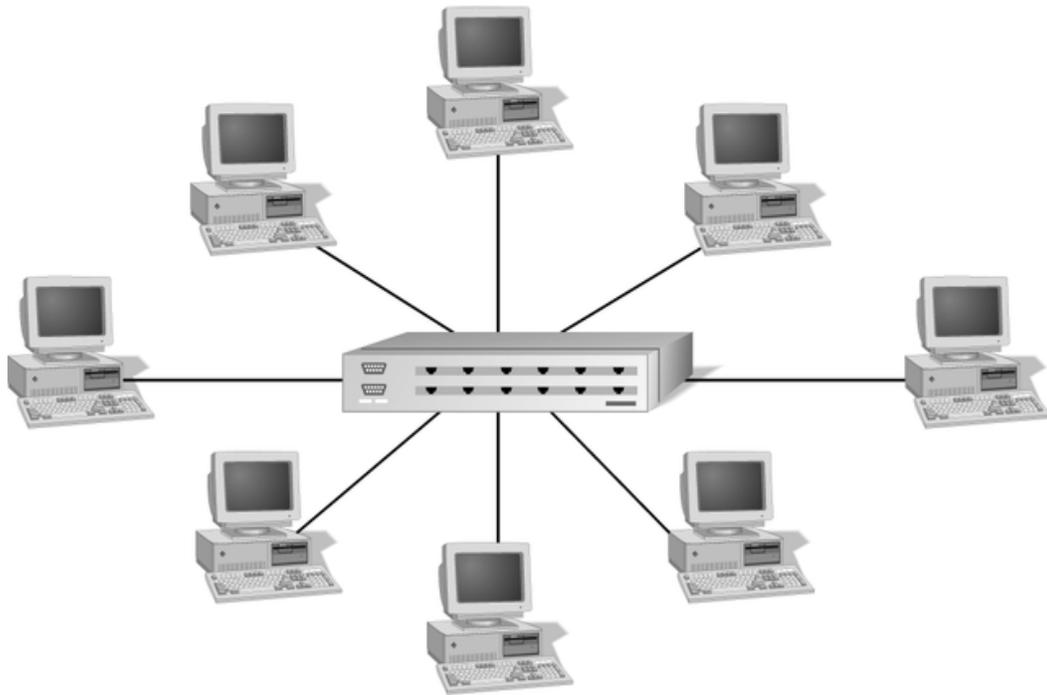
Anexo A.1: Topología de Bus.



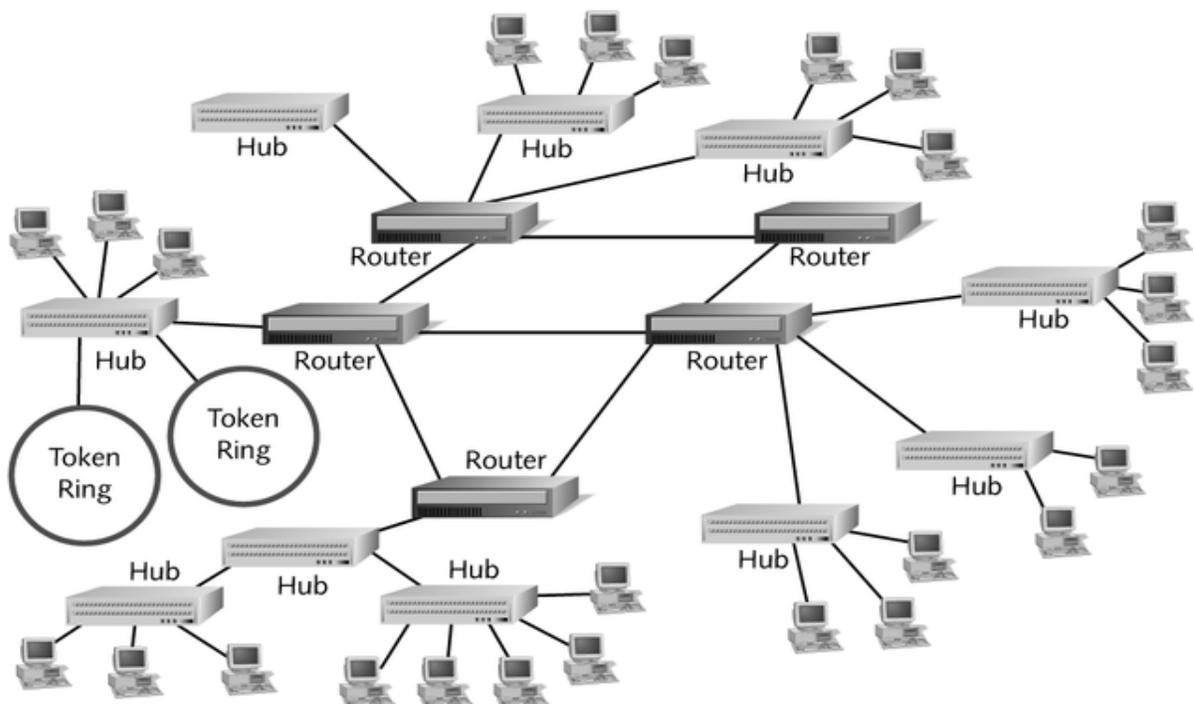
Anexo A.2: Topología de Anillo.



Anexo A.3: Topología de Estrella.



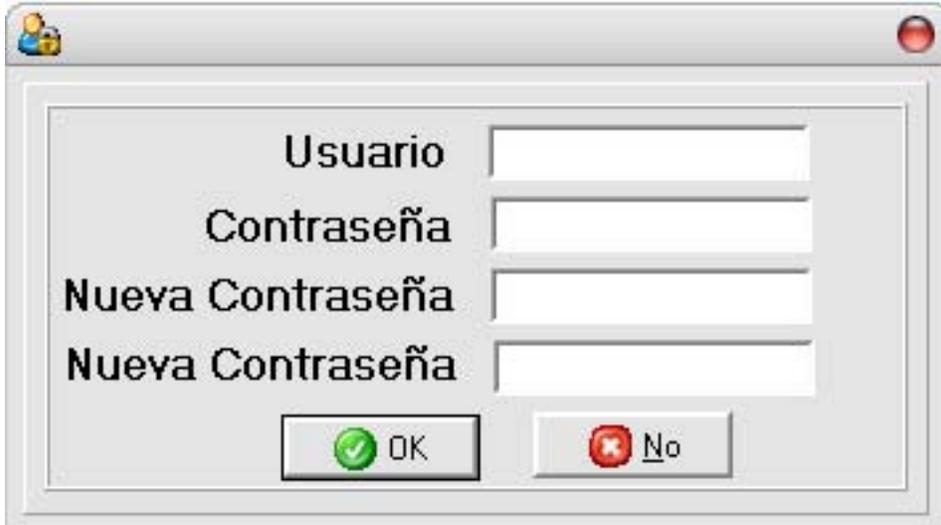
Anexo A.4: Backbone Distribuido interconectando varias LANs.



Anexo B. Prototipos

Prototipos de algunos de los casos de usos del sistema.

Anexo B.1: Caso de uso cambiar contraseña.



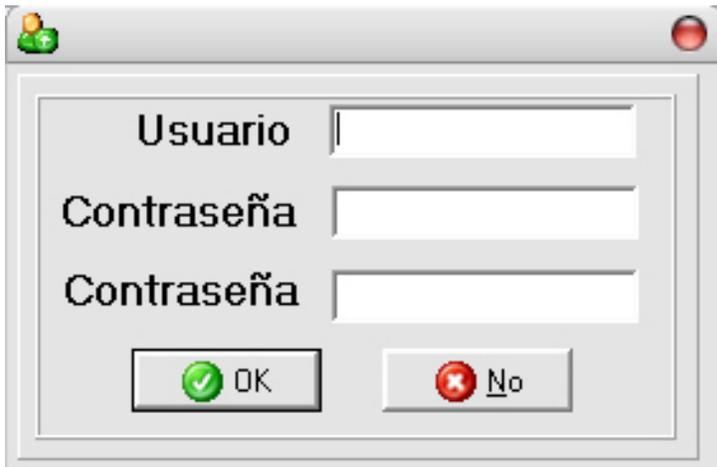
Este prototipo muestra un diálogo de usuario con un título que incluye un ícono de usuario. El diálogo contiene cuatro campos de texto etiquetados como 'Usuario', 'Contraseña', 'Nueva Contraseña' y 'Nueva Contraseña'. En la parte inferior, hay dos botones: 'OK' con un ícono de checkmark verde y 'No' con un ícono de X roja.

Anexo B.2: Caso de uso autenticar.

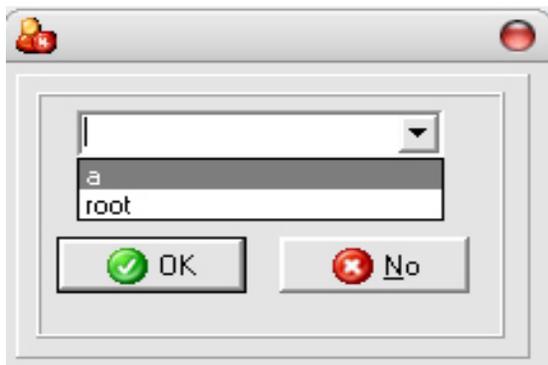


Este prototipo muestra una pantalla de aplicación con el título 'CMR'. El menú superior incluye 'Archivo', 'Configuración', 'Usuario', 'Lenguaje', 'Reporte', 'Consulta' y 'Ayuda'. Hay una barra de herramientas con íconos de archivo, configuración, usuario, lenguaje, reporte, consulta y ayuda. En el centro de la pantalla, hay un diálogo de autenticación con el título 'Autenticarse'. Este diálogo tiene dos campos de texto etiquetados como 'Usuario:' y 'Contraseña:', y un botón 'OK' debajo. En la parte inferior de la pantalla, hay un estado que muestra 'Fecha: 04/03/2008', 'Hora: 13:56:44' y 'Desarrollado por: DIGERS'.

Anexo B.3: Caso de uso gestionar usuario.



A dialog box with a title bar containing a help icon and a close button. The main area contains three text input fields. The first is labeled 'Usuario'. The second and third are both labeled 'Contraseña'. Below the fields are two buttons: 'OK' with a green checkmark icon and 'No' with a red 'X' icon.



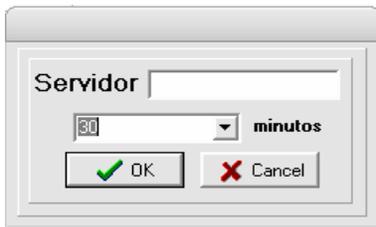
A dialog box with a title bar containing a help icon and a close button. The main area contains a dropdown menu with a downward arrow. The dropdown is open, showing two options: 'a' and 'root'. Below the dropdown are two buttons: 'OK' with a green checkmark icon and 'No' with a red 'X' icon.

Anexo B.4: Caso de uso gestionar locales.

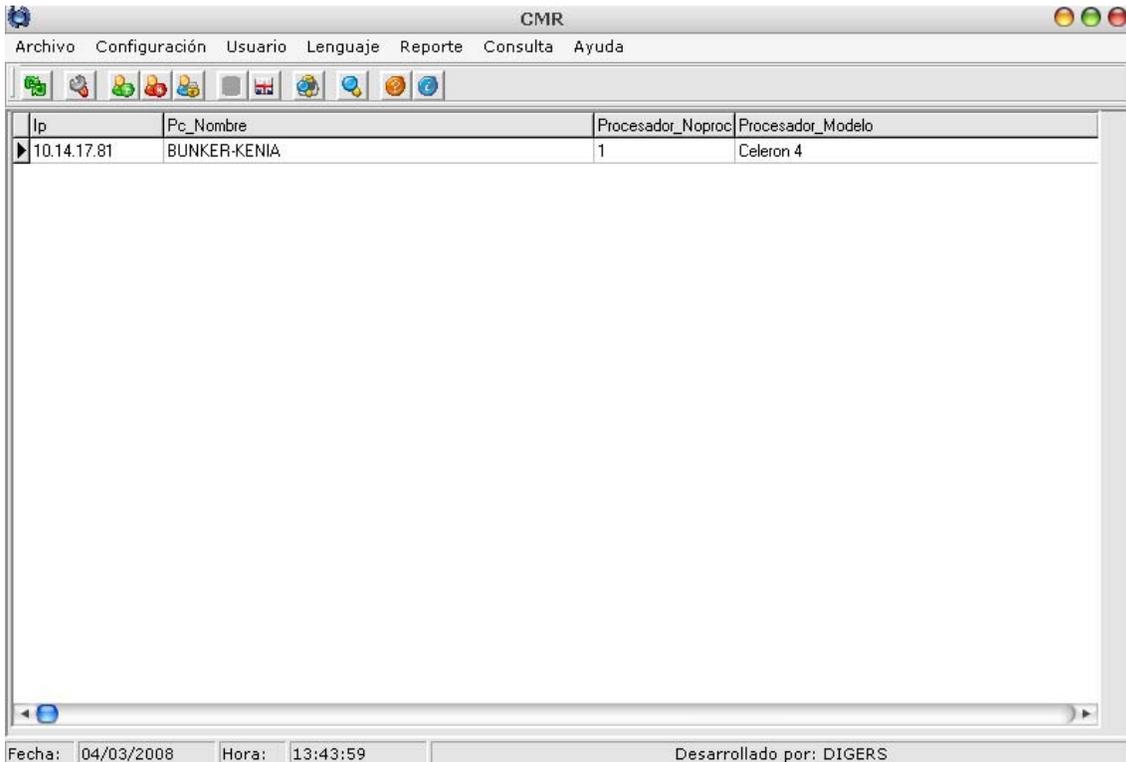


A dialog box with a title bar containing a help icon and a close button. At the top, there are three checkboxes: 'Insertar' (checked), 'Actualizar', and 'Eliminar'. To the right of these is the text 'Locales Activos:'. Below this are three text input fields: 'Nombre Local:', 'IP Inicial:', and 'IP Final:'. At the bottom are two buttons: 'OK' with a green checkmark icon and 'Cancel' with a red 'X' icon.

Anexo B.5: Caso de uso configurar monitor.



Anexo B.6: Caso de uso mostrar todo.



Anexo B.7: Caso de uso mostrar procesador.



Anexo B.8: Caso de uso mostrar placabase.

SIHARW					
Archivo Configuración Usuario Lenguaje Reporte Consulta Ayuda					
Ip	PlacabaseVersion	Placabase_Marca	Placabase_Modelo	Placabase_Noserie	
▶ 10.14.17.80	buena	asus	mx3s	1877e	
10.14.17.81	mala	asus	mx3w	145as	

Anexo B.9: Caso de uso mostrar video.

SIHARW					
Archivo Configuración Usuario Lenguaje Reporte Consulta Ayuda					
Ip	Video_Memoria	Video_Modelotari	Video_Nocolores	Video_Resolucion	
▶ 10.14.17.80	256	ati	24	800x600	
10.14.17.81	1024	all in wonder	32	1024x640	

Anexo B.10: Caso de uso mostrar bios.

SIHAR			
Archivo Configuración Usuario Lenguaje Reporte Consulta Ayuda			
Ip	Bios_Fecha	Bios_Version	
▶ 10.14.17.80	10/05/07	Phoenix	
10.14.17.81	10/06/06	Phoenix	

Anexo B.11: Caso de uso mostrar hdd.

SIHARW				
Archivo Configuración Usuario Lenguaje Reporte Consulta Ayuda				
Ip	Hdd_Modelo	Hdd_Noserie	Hdd_Capacidad	
▶ 10.14.17.80	Maxtor	1a87	40gb	
10.14.17.81	Seagate	147sdf	80gb	
10.14.17.80	Maxtor	21wefd	20gb	

Anexo B.12: Caso de uso mostrar red.

SIHARW						
Archivo Configuración Usuario Lenguaje Reporte Consulta Ayuda						
Ip	Red_Tarjetared	Red_Dominio	Red_Dns	Red_Mac	Red_Mascara	Red_Puertaenlace
▶ 10.14.17.80	planet	ucf	10.14.16.7	00-11	255.255.252.0	10.14.16.4
10.14.17.81	planet	ucf	10.14.16.7	125-45	255.255.252.0	10.14.16.4

Anexo B.13: Caso de uso mostrar ram.

SIHARW						
Archivo Configuración Usuario Lenguaje Reporte Consulta Ayuda						
Ip	Ram_total	Ram_libre	Ram_ocup	Virtual_total	Virtual_libre	Virtual_ocup
▶ 10.14.17.80	256	210	46	2000	1000	1000
10.14.17.81	256	246	10	3000	1500	1500

Anexo B.14: Caso de uso mostrar reportes.

Facultad de Informática
Universidad de Cienfuegos

Sistema de Inventario de Redes Windows

28/05/2008

Procesador	PlacaBase	Discos Duros	Cantidad
pentiumIII	mx3s	20gb	1
pentiumIII	mx3s	40gb	1
pentiumIV	mx3w	80gb	1

ANEXO C. Documento RFC.

PORT NUMBERS

(last updated 2008-05-22)

The port numbers are divided into three ranges: the Well Known Ports, the Registered Ports, and the Dynamic and/or Private Ports.

The Well Known Ports are those from 0 through 1023.

DCCP Well Known ports SHOULD NOT be used without IANA registration. The registration procedure is defined in [RFC4340], Section 19.9.

The Registered Ports are those from 1024 through 49151

DCCP Registered ports SHOULD NOT be used without IANA registration. The registration procedure is defined in [RFC4340], Section 19.9.

The Dynamic and/or Private Ports are those from 49152 through 65535

A value of 0 in the port numbers registry below indicates that no port has been allocated.

```
*****
***
* PLEASE NOTE THE FOLLOWING:
*
*
* IESG STATEMENT TO THE IANA
*
* THE IESG BELIEVES THAT IANA MAY ALLOCATE AN ADDITIONAL PORT IN
* THE 'USER PORT' RANGE TO PROTOCOLS WHOSE CURRENT PORT ALLOCATION
* REQUIRES ACCESS TO A PRIVILEGED PORT. THIS ALLOCATION SHOULD NOT
* BE AUTOMATIC, BUT MAY OCCUR UPON APPLICATION BY AN INTERESTED
* PARTY WHOSE APPLICATION WOULD OTHERWISE FIT IANA'S POLICIES.
*
*
* 1. UNASSIGNED PORT NUMBERS SHOULD NOT BE USED. THE IANA WILL
ASSIGN *
* THE NUMBER FOR THE PORT AFTER YOUR APPLICATION HAS BEEN APPROVED.
*
*
*
```

```
* 2. ASSIGNMENT OF A PORT NUMBER DOES NOT IN ANY WAY IMPLY AN
*
* ENDORSEMENT OF AN APPLICATION OR PRODUCT, AND THE FACT THAT NETWORK
*
* TRAFFIC IS FLOWING TO OR FROM A REGISTERED PORT DOES NOT MEAN THAT
*
* IT IS "GOOD" TRAFFIC. FIREWALL AND SYSTEM ADMINISTRATORS SHOULD
*
* CHOOSE HOW TO CONFIGURE THEIR SYSTEMS BASED ON THEIR KNOWLEDGE OF
*
* THE TRAFFIC IN QUESTION, NOT WHETHER THERE IS A PORT NUMBER
*
* REGISTERED OR NOT.
*
*****
***
```

WELL KNOWN PORT NUMBERS

The Well Known Ports are assigned by the IANA and on most systems can only be used by system (or root) processes or by programs executed by privileged users.

Ports are used in the TCP [RFC793] to name the ends of logical connections which carry long term conversations. For the purpose of providing services to unknown callers, a service contact port is defined. This list specifies the port used by the server process as its contact port. The contact port is sometimes called the "well-known port".

To the extent possible, these same port assignments are used with the UDP [RFC768].

The range for assigned ports managed by the IANA is 0-1023.

Port Assignments:

Keyword	Decimal	Description	References
-----	-----	-----	-----
#	4144	Unassigned	
vvr-control	4145/tcp	VVR Control	
vvr-control	4145/udp	VVR Control	
#		Ming Xu <ming&veritas.com>	
tgconnect	4146/tcp	TGConnect Beacon	
tgconnect	4146/udp	TGConnect Beacon	
#		Brian Becker <bbecker&tg-c-usa.com>	October 2006
vrxpervman	4147/tcp	Multum Service Manager	
vrxpervman	4147/udp	Multum Service Manager	
#		Scott Mager <Scott.Mager&cerner.com>	October 2006

```

hhb-handheld 4148/tcp   HHB Handheld Client
hhb-handheld 4148/udp   HHB Handheld Client
#               Steven G. Loughner
<stevengloughner@eaton.com> March 2007
agslb        4149/tcp   A10 GSLB Service
agslb        4149/udp   A10 GSLB Service
#               John Chiong <jchiong@al0networks.com> 02 July
2007
PowerAlert-nsa 4150/tcp   PowerAlert Network Shutdown Agent
PowerAlert-nsa 4150/udp   PowerAlert Network Shutdown Agent
#               Mike Delgrosso <Mike_Delgrosso@tripplite.com>
02 July 2007
menandmice_noh      4151/tcp   Men & Mice Remote Control
menandmice_noh      4151/udp   Men & Mice Remote Control
#               Eggert Thorlacius <eggert@menandmice.com> 30
August 2007
idig_mux          4152/tcp   iDigTech Multiplex
idig_mux          4152/udp   iDigTech Multiplex
#               Robin Findley <rfindley@usa.net> May 2007
mbl-battd         4153/tcp   MBL Remote Battery Monitoring
mbl-battd         4153/udp   MBL Remote Battery Monitoring
#               Claudio Procida <claudio@emeraldion.it> May
2007
atlinks           4154/tcp   atlinks device discovery
atlinks           4154/udp   atlinks device discovery
#               Scott Griepentrog
<griepentrog@atlinks.com> October 2002
bzzr              4155/tcp   Bazaar version control system
bzzr              4155/udp   Bazaar version control system
#               Martin Pool <mbp@canonical.com> February 2007
stat-results      4156/tcp   STAT Results
stat-results      4156/udp   STAT Results
stat-scanner      4157/tcp   STAT Scanner Control
stat-scanner      4157/udp   STAT Scanner Control
stat-cc           4158/tcp   STAT Command Center
stat-cc           4158/udp   STAT Command Center
#               Darwin Ammala <dammala@harris.com> March 2007
nss               4159/tcp   Network Security Service
nss               4159/udp   Network Security Service
#               Dave Wierbowski <wierbows@us.ibm.com> August
2006
jini-discovery    4160/tcp   Jini Discovery
jini-discovery    4160/udp   Jini Discovery
#               Mark Hodapp <mark.hodapp@sun.com>
omscontact        4161/tcp   OMS Contact
omscontact        4161/udp   OMS Contact
omstopology       4162/tcp   OMS Topology
omstopology       4162/udp   OMS Topology
#               David Page <david.page@sun.com> August
2005
silverpeakpeer    4163/tcp   Silver Peak Peer Protocol
silverpeakpeer    4163/udp   Silver Peak Peer Protocol

```

```

#           Damon Ennis <damon&silver-peak.com> March
2007
silverpeakcomm      4164/tcp   Silver Peak Communication Protocol
silverpeakcomm      4164/udp   Silver Peak Communication Protocol
#           Damon Ennis <damon&silver-peak.com> March
2007
altcp               4165/tcp   ArcLink over Ethernet
altcp               4165/udp   ArcLink over Ethernet
#           Dmitry Brant
<dmitry_brant&lincolnelectric.com> March 2007
joost               4166/tcp   Joost Peer to Peer Protocol
joost               4166/udp   Joost Peer to Peer Protocol
#           Colm MacCarthaigh
<colm.maccarthaigh&joost.com> April 2007
ddgn                4167/tcp   DeskDirect Global Network
ddgn                4167/udp   DeskDirect Global Network
#           Laurie Charlwood
<laurie.charlwood&printsoft.com> 22 August 2007
pslicser            4168/tcp   PrintSoft License Server
pslicser            4168/udp   PrintSoft License Server
#           David Weisgerber
<david.weisgerber&printsoft.de> 27 September 2007
iadt                4169/tcp   Automation Drive Interface Transport
iadt                4169/udp   Automation Drive Interface Transport
#           Paul Suhler <paul.suhler&quantum.com> 15
February 2008
#           4170-4176 Unassigned
wello               4177/tcp   Wello P2P pubsub service
wello               4177/udp   Wello P2P pubsub service
#           Christian Westbrook <cw&wellohorld.com> 15
November 2007
storman             4178/tcp   StorMan
storman             4178/udp   StorMan
#           Werner Guertler <Werner.Guertler&fujitsu-
siemens.com> February 2007
MaxumSP             4179/tcp   Maxum Services
MaxumSP             4179/udp   Maxum Services
#           Greg Stine <greg.stine&siemens.com> 05 July
2007
httpx               4180/tcp   HTTPX
httpx               4180/udp   HTTPX
#           Paul McGough <pmcgough&2factor.com> February
2007
macbak              4181/tcp   MacBak
macbak              4181/udp   MacBak
#           Wes Peters <wes&softweyr.com> April 2007
pcptcpservice      4182/tcp   Production Company Pro TCP Service
pcptcpservice      4182/udp   Production Company Pro TCP Service
#           Ben McNeill <ben.mcneill&cinexec.com> May
2007
gmmp                4183/tcp   General Metaverse Messaging Protocol
gmmp                4183/udp   General Metaverse Messaging Protocol

```

```

#           Gareth Nelson <gareth@garethnelson.com> June
2007
universe_suite 4184/tcp  UNIVERSE SUITE MESSAGE SERVICE
universe_suite 4184/udp  UNIVERSE SUITE MESSAGE SERVICE
#           Gary ANDREWS <gan@orsyp.com> 07 January
2008
wcpp           4185/tcp  Woven Control Plane Protocol
wcpp           4185/udp  Woven Control Plane Protocol
#           Christopher LILJENSTOLPE <cdl@asgaard.org>
14 April 2008
#           4186-4198  Unassigned
eims-admin     4199/tcp  EIMS ADMIN
eims-admin     4199/udp  EIMS ADMIN
#           Glenn Anderson <glenn@qualcomm.co.nz>
vrml-multi-use 4200-4299 VRML Multi User Systems
#           Mitra <mitra@earth.path.net>
corelccam      4300/tcp  Corel CCam
corelccam      4300/udp  Corel CCam
#           Jason Aiken <jasona@corelcomputer.com>
d-data         4301/tcp  Diagnostic Data
d-data         4301/udp  Diagnostic Data
d-data-control 4302/tcp  Diagnostic Data Control
d-data-control 4302/udp  Diagnostic Data Control
#           Jon March <jon.march@paget-analyst-
services.co.uk> September 2006
srcp           4303/tcp  Simple Railroad Command Protocol
srcp           4303/udp  Simple Railroad Command Protocol
#           Matthias Trute <mtrute@web.de> January 2007
owserver       4304/tcp  One-Wire Filesystem Server
owserver       4304/udp  One-Wire Filesystem Server
#           Paul Alfille <paul.alfille@gmail.com> January
2007
batman         4305/tcp  better approach to mobile ad-hoc networking
batman         4305/udp  better approach to mobile ad-hoc networking
#           Simon Wunderlich <siwu@hrz.tu-chemnitz.de> 30
August 2007
pinghgl        4306/tcp  Hellgate London
pinghgl        4306/udp  Hellgate London
#           David Berk <dberk@ping-0.com> 30 August 2007
visicron-vs    4307/tcp  Visicron Videoconference Service
visicron-vs    4307/udp  Visicron Videoconference Service
#           Alexey Vlaskin <avlaskin@visicron.com> 22
October 2007
compX-lockview 4308/tcp  CompX-LockView
compX-lockview 4308/udp  CompX-LockView
#           John Payson <john@circad.com> 22 October 2007
dserver        4309/tcp  Exsequi Appliance Discovery
dserver        4309/udp  Exsequi Appliance Discovery
#           Angelo Masci <angelomasci@yahoo.co.uk>
January 2007
mirrtex        4310/tcp  Mir-RT exchange service
mirrtex        4310/udp  Mir-RT exchange service

```

```

#                               Sylvain Robert
<sylvain.robert@adadis.com> 14 February 2008
#                               4311-4319  Unassigned
fdt-rcatp      4320/tcp    FDT Remote Categorization Protocol
fdt-rcatp      4320/udp    FDT Remote Categorization Protocol
#                               Russell P. Holsclaw, Fast Data Technology,
Inc. <rholsclaw@fastdatatech.com> March 2006
rwhois         4321/tcp    Remote Who Is
rwhois         4321/udp    Remote Who Is
#                               Mark Kosters <markk@internic.net>
trim-event     4322/tcp    TRIM Event Service
trim-event     4322/udp    TRIM Event Service
trim-ice       4323/tcp    TRIM ICE Service
trim-ice       4323/udp    TRIM ICE Service
#                               Siva Poobalasingam
<Siva.Poobalasingam@towersoft.com.au> February 2007
balour         4324/tcp    Balour Game Server
balour         4324/udp    Balour Game Server
#                               Konstantin Schauwecker <hi@konstantin-
schauwecker.de> February 2007
geognosisman  4325/tcp    Cadcorp GeognoSIS Manager Service
geognosisman  4325/udp    Cadcorp GeognoSIS Manager Service
geognosis     4326/tcp    Cadcorp GeognoSIS Service
geognosis     4326/udp    Cadcorp GeognoSIS Service
#                               Martin Daly <martin.daly@cadcorp.com>
November 2006
jaxer-web     4327/tcp    Jaxer Web Protocol
jaxer-web     4327/udp    Jaxer Web Protocol
#                               Uri Sarid <uri@aptana.com> 07 January 2008
jaxer-manager 4328/tcp    Jaxer Manager Command Protocol
jaxer-manager 4328/udp    Jaxer Manager Command Protocol
#                               Uri Sarid <uri@aptana.com> 22 January 2008
#                               4329-4339  Unassigned
gaia          4340/tcp    Gaia Connector Protocol
gaia          4340/udp    Gaia Connector Protocol
#                               Philippe Detournay
<philippe.detournay@acrosoft.be> 12 November 2007
lisp-data     4341/tcp    LISP Data Packets
lisp-data     4341/udp    LISP Data Packets
#                               Dino Farinacci <dino@cisco.com> 12 November
2007
lisp-cons     4342/tcp    LISP-CONS Control
lisp-control  4342/udp    LISP Data-Triggered Control
#                               Dino Farinacci <dino@cisco.com> May 2007
unicall       4343/tcp    UNICALL
unicall       4343/udp    UNICALL
#                               James Powell <james@enghp.unidata.comp>
vinainstall   4344/tcp    VinaInstall
vinainstall   4344/udp    VinaInstall
#                               Jay Slupesky <js@vina-tech.com>
m4-network-as 4345/tcp    Macro 4 Network AS
m4-network-as 4345/udp    Macro 4 Network AS

```

```

#                               Paul Wren <Paul.Wren&macro4.com>
elanlm          4346/tcp        ELAN LM
elanlm          4346/udp        ELAN LM
#                               Paul Ballew <ballew&projtech.com>
lansurveyor    4347/tcp        LAN Surveyor
lansurveyor    4347/udp        LAN Surveyor
#                               Michael Swan <swan&neon.com>
itose          4348/tcp        ITOSE
itose          4348/udp        ITOSE
#                               Michael Haeuptle <Michael_Haeuptle&hp.com>
fspportmap    4349/tcp        File System Port Map
fspportmap    4349/udp        File System Port Map
#                               Ron Minnich <rminnich&sarnoff.com>
net-device    4350/tcp        Net Device
net-device    4350/udp        Net Device
#                               Glenn Peterson <glennp&microsoft.com>
plcy-net-svcs 4351/tcp        PLCY Net Services
plcy-net-svcs 4351/udp        PLCY Net Services
#                               J.J. Ekstrom
pjlink        4352/tcp        Projector Link
pjlink        4352/udp        Projector Link
#                               Mitsuo Kodama <kodama&jbmia.or.jp> June
2005
f5-iquery     4353/tcp        F5 iQuery
f5-iquery     4353/udp        F5 iQuery
#                               Tom Kee <t.kee&f5.com>
qsnet-trans   4354/tcp        QSNet Transmitter
qsnet-trans   4354/udp        QSNet Transmitter
qsnet-workst  4355/tcp        QSNet Workstation
qsnet-workst  4355/udp        QSNet Workstation
qsnet-assist  4356/tcp        QSNet Assistant
qsnet-assist  4356/udp        QSNet Assistant
qsnet-cond    4357/tcp        QSNet Conductor
qsnet-cond    4357/udp        QSNet Conductor
qsnet-nucl    4358/tcp        QSNet Nucleus
qsnet-nucl    4358/udp        QSNet Nucleus
#                               Neer Kleinman <neer&qsr.co.il>
omabcastltkm 4359/tcp        OMA BCAST Long-Term Key Messages
omabcastltkm 4359/udp        OMA BCAST Long-Term Key Messages
#                               Frank Hartung <Frank.Hartung&ericsson.com> 07
September 2007
#                               4360-4367 Unassigned
wxbrief       4368/tcp        WeatherBrief Direct
wxbrief       4368/udp        WeatherBrief Direct
#                               Kim Alan Waggoner <kimw&weatherbank.com>
November 2006
epmd          4369/tcp        Erlang Port Mapper Daemon
epmd          4369/udp        Erlang Port Mapper Daemon
#                               Erlang/OTP support <epmd&erix.ericsson.se>
December 2004
elpro_tunnel  4370/tcp        ELPRO V2 Protocol Tunnel
elpro_tunnel  4370/udp        ELPRO V2 Protocol Tunnel

```

```
#                Harry Courtice
<harry.courtice@elprotech.com> 14 April 2008
#                4371-4372  Unassigned
remctl          4373/tcp    Remote Authenticated Command Service
remctl          4373/udp    Remote Authenticated Command Service
#                Russ Allbery <rra@stanford.edu> 30 August
2007
#                4374-4375  Unassigned
bip             4376/tcp    BioAPI Interworking
bip             4376/udp    BioAPI Interworking
```

DYNAMIC AND/OR PRIVATE PORTS

The Dynamic and/or Private Ports are those from 49152 through 65535

REFERENCES

- [RFC768] Postel, J., "User Datagram Protocol", STD 6, RFC 768, USC/Information Sciences Institute, August 1980.
- [RFC793] Postel, J., ed., "Transmission Control Protocol - DARPA Internet Program Protocol Specification", STD 7, RFC 793, USC/Information Sciences Institute, September 1981.
- [RFC3077] Duros, E., W. Dabbous, H. Izumiyama, N. Fujii, and Y. Zhang, "A Link-Layer Tunneling Mechanism for Unidirectional Links", RFC 3077, March 2001.
- [RFC3340] M. Rose, G. Klyne, and D. Crocker, "The Application Exchange Core", RFC 3340, July 2002.
- [RFC3547] M. Baugher, B. Weis, T. Hardjono and H. Harney, "The Group Domain of Interpretation", RFC 3547, July 2003.
- [RFC3576] M. Chiba, G. Dommety, M. Eklund, D. Mitton, and B. Aboba, "Dynamic Authorization Extensions to Remote Authentication Dial In User Service (RADIUS)", RFC 3576, July 2003.
- [RFC3588] P. Calhoun, J. Loughney, E. Guttman, G. Zorn and J. Arkko, "Diameter Base Protocol", RFC 3588, September 2003.
- [RFC3656] R. Siemborski, "The MUPDATE Distributed Mailbox Database Protocol", RFC 3656, December 2003.
- [RFC3620] D. New, "The TUNNEL Profile", RFC 3620, October 2003.
- [RFC3720] J. Satran, K. Meth, C. Sapuntzakis, M. Chadalapaka, and

- E. Zeidner, "iSCSI", RFC 3720, April 2004.
- [RFC3767] S. Farrell, Ed., "Securely Available Credentials Protocol", RFC 3767, June 2004.
- [RFC3807] E. Weilandt, N. Khanchandani, and S. Rao, "V5.2-User Adaptation Layer (V5UA)", RFC 3807, June 2004.
- [RFC3821] M. Rajagopal, E. Rodriguez, and R. Weber, "Fibre Channel Over TCP/IP (FCIP)", RFC 3821, July 2004.
- [RFC3887] T. Hansen, "Message Tracking Query Protocol", RFC 3887, September 2004.
- [RFC3920] P. Saint-Andre, Ed., "Extensible Messaging and Presence Protocol (XMPP): Core", RFC 3920, October 2004.
- [RFC3947] T. Kivinen, A. Huttunen, B. Swander, and V. Volpe, "Negotiation of NAT-Traversal in the IKE", RFC 3947, January 2005.
- [RFC3983] A. Newton and M. Sanz, "IRIS - Using the Internet Registry Information Service (IRIS) over the Blocks Extensible Exchange Protocol (BEEP)", RFC 3983, January 2005.
- [RFC4065] J. Kempf, "Instructions for Seamoby and Experimental Mobility Protocol IANA", RFC 4065, July 2005.
- [RFC4165] T. George, B. Bidulock, R. Dantu, H. J. Schwarzbauer, and K. Morneault, "Signaling System 7 (SS7) Message Transfer Part 2 (MTP2) - User Peer-to-Peer Adaptation Layer (M2PA)", RFC 4165, September 2005.
- [RFC4171] J. Tseng, K. Gibbons, F. Travostino, C. Du Laney, and J. Souza, "Internet Storage Name Service (iSNS)", RFC 4171, September 2005.
- [RFC4204] J. Lang, Ed., "Link Management Protocol (LMP)", RFC 4204, October 2005.
- [RFC4535] H. Harney, U. Meth, A. Colegrove, G. Gross, "GSAKMP: Group Secure Association Group Management Protocol", RFC 4535, June 2006.

- [RFC4340] E. Kohler, M. Handley and S. Floyd, "Datagram Congestion Control Protocol (DCCP)", RFC 4340, March 2006.
- [RFC4409] R. Gellens and J. Klensin, "Message Submission for Mail", RFC 4409, April 2006.
- [RFC4430] S. Sakane, K. Kamada, M. Thomas and J. Vilhuber, "Kerberosized Internet Negotiation of Keys (KINK)", RFC 4430, March 2006.
- [RFC4656] S. Shalunov, A. Karp, J.W. Boote and M.J. Zekauskas, "A One-way Active Measurement Protocol (OWAMP)", RFC 4656, September 2006.
- [RFC4540] M. Stiemerling, J. Quittek and C. Cadar, "NEC's Simple Middlebox Configuration (SIMCO) Protocol Version 3.0", RFC 4540, May 2006.
- [RFC4744] E. Lear and K. Crozier, "Using the NETCONF Protocol over Blocks Extensible Exchange Protocol (BEEP)", RFC 4744, December 2006.
- [RFC4742] M. Wasserman and T. Goddard, "Using the NETCONF Configuration Protocol over Secure Shell (SSH)", RFC 4742, December 2006.
- [RFC4743] T. Goddard, "Using the Network Configuration Protocol (NETCONF) Over the Simple Object Access Protocol (SOAP)", RFC 4743, December 2006.
- [RFC4727] B. Fenner, "Experimental values In IPv4, IPv6, ICMPv4, ICMPv6, UDP and TCP Headers", RFC 4727, November 2006.
- [RFC4712] A. Siddiqui, D. Romascanu, E. Golovinsky, M. Rahman and Y. Kim, "Transport Mappings for Real-time Application Quality of Service Monitoring (RAQMON) Protocol Data Unit (PDU)", RFC 4712, October 2006.
- [RFC4767] B. Feinstein and G. Matthews, "The Intrusion Detection Exchange Protocol (IDXP)", RFC 4767, March 2007.
- [RFC4934] S. Hollenbeck, "Extensible Provisioning Protocol (EPP) Transport over TCP", RFC 4934, May 2007.

- [RFC4960] Randall Stewart, "Stream Control Transmission Protocol",
RFC 4960,
September 2007.
- [RFC4975] B. Campbell, R. Mahy and C. Jennings, "The Message Session
Relay Protocol",
RFC 4975, September 2007.
- [RFC4992] A. Newton, "XML Pipelining with Chunks for the Information
Registry
Information Service", RFC 4992, August 2007.
- [RFC4993] A. Newton, "A Lightweight UDP Transfer Protocol for the
Internet
Registry Information Service", RFC 4993, August 2007.
- [RFC5087] Y(J). Stein, R. Shashoua, R. Insler and M. Anavi, "Time
Division
Multiplexing over IP (TDMoIP)", December 2007.
- [RFC-ietf-dccp-rtp-07.txt]
C. Perkins, "RTP and the Datagram Congestion Control
Protocol
(DCCP)", RFC XXXX, Month Year.
- [RFC5191] D. Forsberg, Y. Ohba (Ed.), B. Patil, H. Tschofenig and A.
Yegin,
"Protocol for Carrying Authentication for Network Access
(PANA)",
RFC 5191, May 2008.
- [RFC-ietf-manet-iana-07.txt]
I. Chakeres, "IANA Allocations for MANET Protocols", RFC
XXXX,
Month Year.