



**Universitat Autònoma
de Barcelona**

Gestión centralizada de software de una red Sanitaria en Clientes Windows

Memoria del proyecto
de Ingeniería Técnica en
Informática de Sistemas
realizado por
Sergio Sánchez Gutiérrez
y dirigido por
Yolanda Benítez Fernández

Escuela de Ingeniería
Sabadell, septiembre de 2012

El Bajo firmante, Yolanda Benítez Fernández,
Profesor de la Escuela de Ingeniería de la UAB,

CERTIFICA:

Que el trabajo al que corresponde la presente memoria
ha sido realizado bajo su dirección por Sergio Sánchez
Gutiérrez.

Y para que conste firma la presente.

Sabadell, Septiembre de 2012

Firmado: Yolanda Benítez Fernández

Director

**HOJA DE RESUMEN – PROYECTO FIN DE CARRERA DE L'ESCOLA
D'ENGINYERIA**

Título del proyecto: Gestión centralizada de software de una red Sanitaria en Clientes Windows	
Autor: Sergio Sánchez Gutiérrez	Fecha: Septiembre 2012
Tutor: Yolanda Benítez Fernandez	
Titulación: Ingeniería Técnica de Sistemas.	
Palabras clave (mínimo 3) <ul style="list-style-type: none">• Catalán: Opsi, gestió, administració, sistema, xarxa.• Castellano: Opsi, gestión, administración, sistema, red.• Inglés: Opsi, management, administration, system, network.	
Resumen del proyecto (extensión máxima 100 palabras) <ul style="list-style-type: none">• Catalán: El projecte que es presenta tracta de l'estudi i desenvolupament per a l'implementació d'un sistema de gestió centralitzada de programari d'una xarxa sanitària a Clients Windows. Aquest projecte sorgeix de la necessitat, en una empresa concertada donat el creixement de la xarxa, de poder administrar els hosts d'una manera més automatitzada i més eficient.• Castellano: El proyecto que se presenta trata del estudio y desarrollo para la implementación de un sistema de gestión centralizada de software de una red Sanitaria en Clientes Windows. Este proyecto surge de la necesidad, en una empresa concertada dado el crecimiento de la red, de poder administrar los host de una manera más automatizada y más eficiente.• Inglés: The current project is a research and development for the implementation of a centralized management system of a health network software on Windows clients. This project arises from the need, in a concerted company given the growth of the network, the host can manage in a more automated and more efficient.	

Índice

1.	Introducción	2
1.1-	Presentación	2
1.2-	Objetivos	2
1.3-	Estado del arte.....	3
1.4-	Estructura de la memoria	3
2.	Estudio de viabilidad.....	6
2.1-	Introducción	6
2.1-1.	Tipología y palabras clave.....	6
2.1-2.	Descripción.....	6
2.1-3.	Objetivos del proyecto	6
2.1-4.	Definiciones, acrónimos y abreviaciones	7
2.1-5.	Partes interesadas (Stakeholders).....	8
2.1-6.	Referencias.....	8
2.1-7.	Producto y documentación del proyecto	8
2.2-	Estudio de la situación actual	9
2.2-1.	Contexto	9
2.2-2.	Lógica del sistema	9
2.2-3.	Descripción Física	10
2.2-4.	Usuarios y/o personal del sistema.....	10
2.2-5.	Diagnóstico del sistema	10
2.2-6.	Normativas y legislación.....	11
2.3-	Requisitos del sistema.....	11
2.3-1.	Requisitos funcionales.....	11
2.3-2.	Requisitos no funcionales.....	11
2.3-3.	Restricciones del sistema.....	12
2.3-4.	Catalogación i priorización de los requisitos.....	12
2.4-	Alternativas y selección de la solución	13
2.4-1.	Alternativa 1	13
2.4-2.	Alternativa 2.....	14
2.4-3.	Solución propuesta	15
2.5-	Conclusiones.....	16
3.	Plan de proyecto	18
3.1-	Introducción	18
3.1-1.	Descripción.....	18

3.1-2.	Definiciones, acrónimos y abreviaciones	18
3.1-3.	Referencias.....	18
3.2-	WBS (Work Breakdown Structure)	19
3.2-1.	Fases y actividades del proyecto.....	19
3.2-2.	Diagrama WBS.....	20
3.2-3.	Milestone	20
3.3-	Recursos del Proyecto	21
3.3-1.	Recursos del proyecto	21
3.3-2.	Calendario de los recursos	22
3.4-	Planificación Temporal.....	22
3.4-1.	Calendario del proyecto.....	22
3.4-2.	Dependencias	22
3.4-3.	Cuadro de tareas del proyecto.....	23
3.4-4.	Calendario temporal.....	24
3.5-	Evaluación de riesgos.....	25
3.5-1.	Lista de riesgos.....	25
3.5-2.	Catalogación de riesgos.....	26
3.5-3.	Plan de Contingencia.....	26
3.6-	Presupuesto.....	27
3.6-1.	Estimación costes de personal	27
3.6-2.	Estimación costo de los recursos.....	28
3.6-3.	Estimación de otros costes.....	28
3.6-4.	Resumen i análisis coste beneficio.....	29
3.7-	Conclusiones.....	29
4.	Análisis.....	31
4.1-	Introducción	31
4.2-	Conceptos teóricos	31
4.2-1.	¿Qué es GNU/Linux?	31
4.2-2.	¿Qué es el proyecto GNU?.....	32
4.2-3.	¿Qué es el software libre?.....	32
4.2-4.	Licencia General de GNU	33
4.2-4.1.	GPL versión 3	33
4.2-5.	¿Qué es un sistema de Gestión Centralizado de software?	34
4.3-	La red en BSA	34
4.4-	Software utilizado.....	36

4.4-1.	CentOS	36
4.4-2.	OPSI	37
4.4-2.1.	OPSI servidor.....	37
4.4-2.2.	Interfaz de gestión	37
4.4-2.3.	Instalación de SO	38
4.4-2.4.	Implementación de software	39
4.4-2.5.	OPSI cliente-agente	39
4.4-2.6.	Patchmanagement	40
4.4-2.7.	Inventario de hardware	40
4.4-2.8.	Inventario de software	40
4.4-2.9.	Soporte de múltiples ubicaciones	41
4.4-2.10.	Gestión de licencias (*)	41
4.4-3.	Mysql.....	42
4.4-4.	Webmin	43
4.5-	Diagramas de caso de uso	43
4.5-1.	Login	44
4.5-2.	Añadir host.....	44
4.5-3.	Eliminar Host	45
4.5-4.	Distribuir Software a Host.....	46
4.5-5.	Eliminación software de Host.....	46
4.5-6.	Añadir paquete a repositorio	47
4.5-7.	Eliminar paquete de repositorio	48
4.5-8.	Distribuir hardware	48
4.5-9.	Eliminar hardware	49
5.	Implementación	51
5.1-	Introducción	51
5.2-	Arquitectura	51
5.3-	Servidor	52
5.4-	Herramientas OPSI	54
5.4-1.	Herramienta: OPSI-setup	54
5.4-2.	Herramienta: Interfaz de gestión OPSI-configed	55
5.4-3.	Herramienta: opsi-package-manager	56
5.4-4.	Herramienta: opsi-product-updater	57
5.4-5.	Herramientas: opsi-admin / interfaz configuración OPSI	58
5.5-	Ficheros de configuración	58

5.5-1.	opsiconfd.conf	58
5.5-2.	smb.conf	59
5.5-3.	site.py	61
5.5-4.	dispatch.conf	62
5.6-	Funcionamiento cliente servidor.....	62
5.7-	Interfaces.....	64
5.7-1.	Línea de comando	64
5.7-2.	Navegador web	65
6.	Conclusiones	67
6.1-	Introducción	67
6.2-	Conclusiones.....	67
6.3-	Futuras ampliaciones	67
6.4-	Problemas encontrados	68
6.5-	Valoración General	69
7.	Bibliografía	71
7.1-	Direcciones de internet.....	71
8.	Glosario	73

Índice de Tablas

Tabla 1: Criticidad de los objetivos.....	7
Tabla 2: Stakeholders	8
Tabla 3: Descripción física	10
Tabla 4: Prioridad Requisitos Funcionales.....	12
Tabla 5: Prioridad Requisitos no funcionales	12
Tabla 6: Relación entre los objetivos y los requisitos	13
Tabla 7: Solución propuesta.....	15
Tabla 8: Fases y actividades del proyecto	19
Tabla 9: Milestone	21
Tabla 10: Recursos del proyecto	21
Tabla 11: Recursos del proyecto	23
Tabla 12: Diagrama de Gantt	24
Tabla 13: Catalogación de Riesgos	26
Tabla 14: Planes de Contingencia	27
Tabla 15: Costes de personal	28
Tabla 16: Coste de recursos.....	28
Tabla 17: caso de uso Login.....	44
Tabla 18: caso de uso Añadir host	45
Tabla 19: caso de uso Eliminar Host.....	46
Tabla 20: caso de uso Distribuir Software a Host	46
Tabla 21: caso de uso Eliminación software de Host	47
Tabla 22: caso de uso Añadir paquete a repositorio	48
Tabla 23: caso de uso Eliminar paquete de repositorio.....	48
Tabla 24: caso de uso Distribuir hardware.....	49
Tabla 25: caso de uso Eliminar hardware	49

Índice de Ilustraciones

Ilustración 1: Lógica del sistema.....	9
Ilustración 2: Diagrama WBS	20
Ilustración 3: Red BSA	35
Ilustración 4: Distribución de los técnicos de soporte	35

Ilustración 5: Esquema OPSI.....	52
Ilustración 6: Vmware en BSA.....	53
Ilustración 7: servdesk.....	53
Ilustración 8: Webmin en servdesk	54
Ilustración 9: OPSI configed en servdesk	56
Ilustración 10: opsi-package manager en servdesk	57
Ilustración 11: Conexión cliente con servidor servdesk	63
Ilustración 12: Conexión servdesk mediante ssh.....	64

Introducción

1. Introducción

1.1- Presentación

El proyecto que se presenta trata del estudio y desarrollo para la implementación de un sistema de gestión centralizada de software de una red Sanitaria en Clientes Windows.

Este proyecto surge de la necesidad, en una empresa concertada dado el crecimiento de la red, de poder administrar los host de una manera más automatizada y más eficiente.

Dado que la empresa consta de aproximadamente 850 host, el estudio y desarrollo será un piloto de unos host elegidos, ya que el tiempo del proyecto es limitado así como los recursos. Por tanto una vez satisfecho el proyecto universitario el proyecto empresarial seguirá desarrollándose.

1.2- Objetivos

Los objetivos básicos que debe cubrir la aplicación son:

- Inventario de software específico de la empresa.
- Inventario de software general.
- Inventario de host de la empresa.
- Inventario de hardware (drivers).
- Instalación remota de software.
- Instalación remota de hardware (drivers).

Cabe destacar que aparte de estos objetivos hay otros que por necesidades de la empresa podrían ser cubiertos. Esto vendrá dado por la herramienta escogida y las posibilidades que se nos ofrezca.

1.3- Estado del arte

Dado que en la actualidad existen varias aplicaciones orientadas a la gestión de software y hardware, deberá valorarse el coste y las funcionalidades de las mismas, en comparación al desarrollo de un proyecto a medida.

1.4- Estructura de la memoria

Incluyendo el apartado que nos ocupa, la memoria está formada por 8 partes donde se elabora un detallado análisis del proyecto que se va a llevar a cabo.

- **Presentación:** Describiremos en líneas generales el proyecto que se va a desarrollar, los objetivos que debe cumplir y el contexto donde se situará este proyecto.

- **Estudio de Viabilidad:** Analizaremos la viabilidad del proyecto dentro de la situación actual teniendo en cuenta las especificaciones más técnicas.

- **Plan del Proyecto:** Determinaremos prioridades y criterios, cobertura de equipamientos y disposición de recursos, su previsión presupuestaria y horizonte temporal.

- **Análisis:** Detallaremos el software utilizado en el desarrollo del proyecto, y analizaremos el diseño del sistema, así como sus casos de uso.

- **Implementación:** Capítulo dedicado a los detalles relativos a la implementación del sistema.

Explicaremos detalladamente la arquitectura de la aplicación, así como los detalles técnicos relacionados con el funcionamiento de la misma.

- **Conclusiones:** Conclusiones resultantes de la elaboración de este proyecto, y analizaremos posibles líneas futuras de ampliación del mismo.

- **Bibliografía:** Referencia de la documentación consultada.

Estudio de Viabilidad

2. Estudio de viabilidad

2.1- Introducción

2.1-1. Tipología y palabras clave

Tipología: Administración y gestión.

Palabras clave: Gestión, administración, sistema, red, desatendida, S.O. (sistema operativo).

2.1-2. Descripción

Este Proyecto surge de la necesidad de inventariar una red sanitaria de 850 hosts. Concretamente la empresa es BSA (Badalona Serveis Assitencials). La empresa consta de un hospital, 7 ambulatorios, 1 clínica socio sanitaria, 1 centro de atención a la salud sexual, 1 edificio de consultas externas, 1 almacén y 2 edificios de oficinas.

Esta red está consta de 850 hosts con clientes Windows. Y pretendemos automatizar todo la distribución de software y hardware (drivers) en dichos hosts, y tener un repositorio estándar de software tanto propietario de la empresa como externo y llevar un control detallado de ello.

Para ello y debido al alcance del proyecto, se hará un piloto y en caso de tener éxito se procederá a la implantación a toda la organización, pero fuera del proyecto universitario.

2.1-3. Objetivos del proyecto

- O1. Inventario de software específico de la empresa.
- O2. Inventario de software general.
- O3. Inventario de host de la empresa.

- O4. Inventario de hardware (drivers).
- O5. Instalación remota de software.
- O6. Instalación remota de hardware (drivers).
- O7. Gestión de Licencias.
- O8. Instalación de SO.
- O9. Sistema de ayuda

<i>Objetivo</i>	<i>Crítico</i>	<i>Prioritario</i>	<i>Secundario</i>
1		X	
2	X		
3	X		
4	X		
5			X
6	X		
7			X
8			X
9	X		

Tabla 1: Criticidad de los objetivos

2.1-4. Definiciones, acrónimos y abreviaciones

LOPD: Ley orgánica de protección de datos.

OPSI: Open pc server integration.

2.1-5. Partes interesadas (Stakeholders)

Stakeholders		
Nombre	Descripción	Responsabilidad
A	Administrador del sistema	Aprobación del proyecto. Participa en su definición y hace el seguimiento del proyecto.
B	Operador del sistema	Persona encargada de desarrollar el proyecto. Implementará la solución
C	Técnico de soporte	Participará en la instalación de los clientes.

Tabla 2: Stakeholders

2.1-6. Referencias

- 1 - Normativa de proyectos de ingeniería técnica.
http://www.uab.cat/Document/541/595/Normativa_PFCNo_vembre2010.pdf
- 2 - LOPD:
<https://www.agpd.es/portalweb/canaldocumentacion/legislacion/estatal/index-ides-idphp.php>

2.1-7. Producto y documentación del proyecto

1. Se instalará un servidor para proveer el software y la monitorización de este.
2. Se entregará un manual de usuario.
3. Se elaborará una memoria del proyecto.

2.2- Estudio de la situación actual

2.2-1. Contexto

En la actualidad, en la empresa BSA, se dispone del aplicativo WSUS de Microsoft para la distribución de actualizaciones críticas y de sistema operativo. Esto resulta muy útil, pero a la vez, insuficiente, porque con esto queda muy reducido la distribución del software.

El resto de aplicativos requiere una intervención por parte del personal de soporte técnico y dado la cantidad de hosts disponibles y las diferentes sedes, cada cambio de software resulta muy laborioso y poco práctico.

El coste personal es muy alto y el personal de soporte se tiene que dedicar también a otros menesteres, por lo que si la distribución y administración fuese corporativa se ganaría muchísimo tiempo y dinero, aparte también de tener un inventario actualizado y en tiempo real.

Es por todo esto que el administrador de sistemas de BSA hace la petición de arrancar dicho proyecto, con el beneplácito de la Universitat Autònoma de Barcelona y la tutora Yolanda Benítez como tutora.

2.2-2. Lógica del sistema

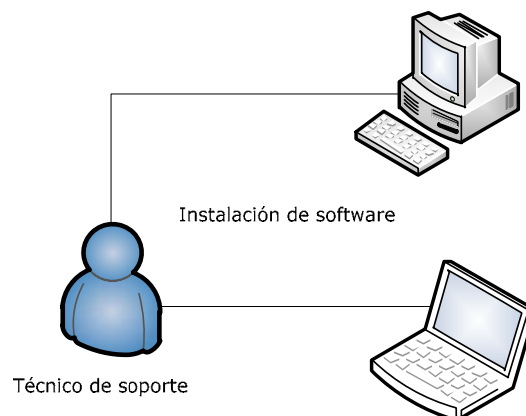


Ilustración 1: Lógica del sistema

2.2-3. Descripción Física

<i>Cliente</i>	<i>Servidor</i>
<ul style="list-style-type: none"> • Procesador: Pentium Dual Core 2GhZ • Disco Duro: 120 Gb • DVD • TFT • Tarjeta de red • Windows Xp Profesional 	<ul style="list-style-type: none"> • Procesador: 2 Procesadores Virtuales • Memoria: 4058 MB • Disco Duro: 16 GB • S.O. : CentOS 5.0

Tabla 3: Descripción física

2.2-4. Usuarios y/o personal del sistema

<i>Usuarios y/o personal del sistema</i>		
<i>Nombre</i>	<i>Descripción</i>	<i>Responsabilidad</i>
A	Administrador del sistema	Aprobación del proyecto. Participa en su definición y hace el seguimiento del proyecto.
B	Operador del sistema	Persona encargada de desarrollar el proyecto. Implementará la solución
C	Técnico de soporte	Participará en la instalación de los clientes.

Tabla 4: Usuarios y/o personal del sistema

2.2-5. Diagnóstico del sistema

Deficiencias:

- El sistema actual es muy dependiente de las personas.
- Es propenso a errores i a falta de software.
- Falta de inventario de los hosts.

Mejoras:

- Acceso eficiente al inventario.
- Reducción drástica de costes de personal.
- Eficiencia en la administración de hardware /software.

2.2-6. Normativas y legislación

- LOPD: Ley orgánica de protección de datos.
- Normativa de proyectos de final de carrera de EI.
- Ley de propiedad intelectual.

2.3- Requisitos del sistema

2.3-1. Requisitos funcionales

RF1. Inventario de software específico de la empresa.

RF2. Inventario de software general.

RF3. Inventario de host de la empresa.

RF4. Inventario de hardware (drivers).

RF5. Instalación remota de software.

RF6. Instalación remota de hardware (drivers).

RF7. Gestión de Licencias.

RF8. Instalación de SO.

Rf9. Sistema de ayuda.

2.3-2. Requisitos no funcionales

RNF1. Cumplimiento de la LOPD por lo que hace referencia a los ficheros de datos i a los derechos de los clientes.

RNF2. Los recursos utilizados por la aplicación han de estar ajustados a la medida de la entidad.

RNF3. Tolerancia a fallos i a acciones incorrectas.

RNF4. Control de todas las entradas de usuarios.

RNF5. La seguridad de los datos: el servidor tendrá que tener restricciones de acceso físico.

2.3-3. Restricciones del sistema

RS1. El sistema ha de adaptarse al sistema físico disponible en la entidad.

RS2. Este sistema ha de ser virtualizado.

RS3.El proyecto ha de estar finalizado el 14 de de septiembre de 2012.

2.3-4. Catalogación i priorización de los requisitos

Requisitos Funcionales:

	RF1	RF2	RF3	RF4	RF5	RF6	RF7	RF8	RF9
Esencial	X	X	X	X	X	X			X
Condicional							X		
Opcional								X	

Tabla 4: Prioridad Requisitos Funcionales

Requisitos No funcionales:

	RNF1	RNF2	RNF3	RNF4	RNF5
Esencial	X	X		X	X
Condicional			X		
Opcional					

Tabla 5: Prioridad Requisitos no funcionales

Relaciones entre los objetivos y los requisitos:

	RF1	RF2	RF3	RF4	RF5	RF6	RF7	RF8	RF9	RNF1	RNF2	RNF3	RNF4	RNF5
O1	X								X	X	X	X	X	X
O2		X							X	X	X	X	X	X
O3			X						X	X	X	X	X	X
O4				X					X	X		X	X	X
O5					X				X	X		X		X
O6						X			X	X		X		X
O7							X		X	X		X		X
O8								X	X	X			X	X
O9									X	X	X			X

Tabla 6: Relación entre los objetivos y los requisitos

2.4- Alternativas y selección de la solución

2.4-1. **Alternativa 1**

System Center Configuration Manager 2012

(<http://www.microsoft.com/systemcenter/es/es/default.aspx>)

Microsoft System Center Configuration Manager, frecuentemente abreviado como SCCM o CofigMgr por los expertos, es la evolución de un producto anterior llamado System Management Server (SMS).

System Center Configuration Manager es un producto diseñado para administrar la configuración de los hosts, servidores y dispositivos móviles (con Windows Mobile) de una organización, ver como están, inventariar y en caso de ser necesario modificarlo.

Características:

- Tener un inventario de software y hardware de todas las máquinas.
- Cruzar el inventario de software el contrato de licencias Microsoft para ver cuánto queda y que falta.
- Distribuir actualizaciones de software, a diferencia de WSUS no solo de Microsoft sino de cualquier fabricante.
- Distribuir software remotamente de manera optimizada, incluso a través de canales de baja velocidad.
- Instalar o actualizar el sistema operativo de las máquinas remotamente.
- Configurar cuarenta de red: habilitar NAP tecnología que permite garantizar que las máquinas que se conecten a la red cumplan con ciertas políticas de seguridad, por ejemplo que tengan el antivirus o una actualización de sistema operativo específica, en caso de que esta no este se le instala.
- Garantizar que todas las máquinas tengan la misma configuración de hardware.
- Tener un control de activos: si una máquina cambia su configuración, salta una alerta

Costes:

- Licencia por cliente.
- Licencia de aplicación
- Servidor.

2.4-2. Alternativa 2

OPSI (<http://www.opsi.org/en>)

OPSI (*open pc server integration*) es una aplicación open source basada en Linux que permite gestionar de forma remota, la instalación y el mantenimiento del software relacionado con equipos Microsoft Windows.

Características:

- Distribución de software automatizada y la gestión de parches (agente OPSI cliente).

- Automatizado de instalación del SO (sin vigilancia o basada en la imagen).
- Inventarios de hardware y software.
- Soporte de múltiples ubicaciones.

Costes:

- Servidor.

2.4-3. Solución propuesta

	<i>Costes de adquisición</i>	<i>Costes de adaptación</i>	<i>Soporte</i>	<i>Nivel de integración</i>	<i>Complejidad</i>	<i>Formación</i>
Alternativa 1	60000€ + 80000€	Bajos.	Partner, contrato de mantenimiento	Muy alto.	Alta	Se desconoce.
Alternativa 2	0 €	Bajos.		Alto	Muy alta	Se desconoce.

Tabla 7: Solución propuesta

A la vista de las dos alternativas propuestas y el cuadro comparativo, se ha escogido la alternativa 2.

Se implementará con la aplicación OPSI, ya que el precio de la alternativa 1 es muy alto y B.S.A. No puede hacer frente, en estos tiempos de crisis a un desembolso tan alto.

Por otro lado, las diferencias funcionales entre las diferentes aplicaciones no son tan grandes y constituye un reto implementar el sistema en plataforma open source.

2.5- Conclusiones

Beneficios:

- Bajo coste.
- Reducción de costes de personal.
- Eficiencia a la hora de gestionar el software.
- Eficiencia a la hora de gestionar hardware.

Inconvenientes:

- Aplicación sin soporte oficial
- Fuerte inversión de personal inicialmente.

Una vez analizado lo que representa para B.S.A. la implantación de OPSI se determina que el proyecto es viable.

Beneficios + inconvenientes = Proyecto Viable

Plan de proyecto

3. Plan de proyecto

3.1- Introducción

3.1-1. Descripción

Este documento recoge el plan de proyecto, es decir, el conjunto de actividades que permiten desarrollar, ejecutar i controlar el proyecto.

Incluyen las tareas i los puntos de control del proyecto, los recursos del proyecto, el calendario del proyecto, la evaluación de los riesgos i el presupuesto del proyecto.

Se ha utilizado una metodología lineal en la elaboración del plan.

3.1-2. Definiciones, acrónimos y abreviaciones

1. Microsoft Project: Programa de Microsoft para la gestión de proyectos.
2. WBS: Work Breakdown Structure.
3. Milestone: Punto de control.
4. Diagrama de Gantt: Cronograma del proyecto.

3.1-3. Referencias

1. Normativa de proyectos de ingeniería técnica.
http://www.uab.cat/Document/541/595/Normativa_PFCNov
2. Microsoft Project.
<http://www.microsoft.com/project/>

3.2- WBS (Work Breakdown Structure)

Para poder planificar el trabajo, conocer con exactitud cuáles son las fases del proyecto, con qué concluye cada una de esas fases y la fecha prevista de su finalización se utilizará la herramienta WBS.

Se describirá cada una de las partes del proyecto y detallará cuáles son los puntos de control.

3.2-1. Fases y actividades del proyecto

Este proyecto se puede separar en las siguientes fases, cada una de ellas viene descrita con las actividades a realizar.

<i>Fases</i>	<i>Descripción</i>
Iniciación	Fase de iniciación. Incluye las actividades, definición del proyecto, asignación y matriculación.
Planificación	Incluye el Estudio de Viabilidad y el Plan de Proyecto.
Análisis	Análisis de los requisitos funcionales y no funcionales
Diseño	Incluye el diseño de la capa de datos, de control i la interface.
Desarrollo	Fase de desarrollo de la aplicación
Test i pruebas	Fase de prueba del sistema.
Implantación	La aplicación se instala en su entorno real. Incluye la formación de usuarios
Generación de Documentos	Fase de documentación del proyecto. Incluye manuales y memoria del proyecto.
Cierre del proyecto	Fase de cierre. El director del proyecto firma la aceptación y cierre del proyecto
Defensa del proyecto	Defensa del proyecto delante la comisión.

Tabla 8: Fases y actividades del proyecto

3.2-2. Diagrama WBS

La WBS o Estructura desglosada del trabajo, es una técnica de planificación mediante la cual podemos definir y cuantificar el trabajo a realizar en todo el proyecto.

Para poder visualizar de manera más clara cada una de estas fases se muestra el siguiente esquema. Los nodos intermedios nombran las fases del proyecto y los nodos terminales muestran los objetivos a conseguir.

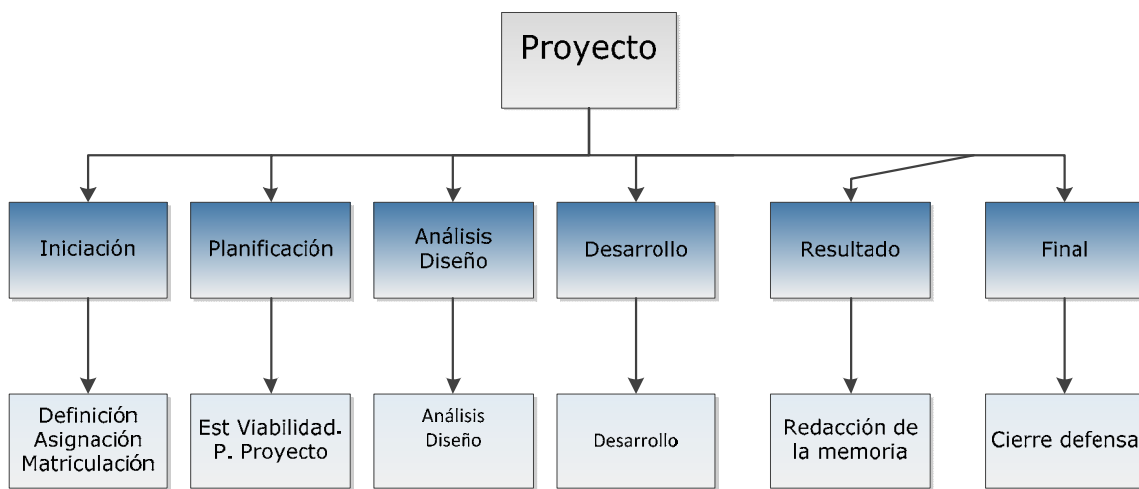


Ilustración 2: Diagrama WBS

3.2-3. Milestone

La siguiente tabla muestra las fechas en las que se hará un punto de control y que, además, tiene que estar finalizada cada fase.

<i>Nombre</i>	<i>Descripción</i>	<i>Fecha</i>
Inicio	Matriculación	3/10/11
E. Viabilidad	Aprobación	14/11/11
Plan de proyecto	Aprobación	29/11/11
Análisis	Aprobación	9/01/12
Diseño	Aprobación	12/03/12
Desarrollo y pruebas	Aprobación	4/05/12
Cierre del proyecto	Aceptación	14/09/12
Defensa del proyecto	Evaluación	18/09/12

Tabla 9: Milestone

3.3- Recursos del Proyecto

3.3-1. Recursos del proyecto

<i>Recursos humanos</i>	<i>Valoración</i>
Jefe de proyecto	100 €/h
Analista	50 €/h
Técnico de soporte	30 €/h

Tabla 10: Recursos del proyecto

Recursos materiales: Se utilizarán los recursos materiales disponibles en la entidad. Todo el desarrollo se hará utilizando programa de dominio público.

3.3-2. Calendario de los recursos

Los recursos humanos que se utilizarán en todo el proyecto:

- Jefe de proyecto: Iniciación, planificación, Generación de documentos, cierre y defensa. Puntos de control.
- Analista: Análisis y diseño. Implantación i puntos de control del análisis, diseño y desarrollo.
- Técnico de soporte: Fase de test, implantación en clientes.

Los recursos materiales se utilizarán principalmente en las fases de desarrollo, test e implantación.

3.4- Planificación Temporal

3.4-1. Calendario del proyecto

El proyecto se desarrollará desde Octubre de 2011 hasta Septiembre de 2012 con una dedicación de 8 horas semanales aproximadamente. El total de horas dedicadas al proyecto será de 416 horas.

- Fecha de inicio: 3 de octubre de 2011
- Fecha de finalización: 18 de Septiembre de 2012
- Planificación: Para la planificación se ha utilizado la herramienta Microsoft Project.

3.4-2. Dependencias

Todas las fases se desarrollan utilizando un modelo lineal, es decir, cada fase no se inicia hasta que la fase anterior no está completada.

La fase de generación de documentos se prevé al final porque incluirá los documentos elaborados durante el desarrollo del proyecto.

3.4-3. Cuadro de tareas del proyecto

T








Id		Nombre de tarea	Duración	Comienzo	Fin	Predecesoras
7		Aprobación Plan del Proyecto (Punto de control)	1 día	mar 29/11/11	mar 29/11/11	6
8		Análisis de la aplicación	28 días	mié 30/11/11	lun 09/01/12	
9		Análisis de requisitos (casos de uso)	2 días	mié 30/11/11	jue 01/12/11	7
10		Análisis de la seguridad y legalidad	1 día	vie 02/12/11	vie 02/12/11	9
11		Documentación del análisis	3 días	lun 05/12/11	mié 07/12/11	10
12		Aprobación del análisis	0 días	lun 09/01/12	lun 09/01/12	11
13		Diseño de la aplicación	45 días	lun 09/01/12	lun 12/03/12	
14		Instalación del servidor	8 días	lun 09/01/12	mié 18/01/12	12
15		Instalación de la aplicación	10 días	jue 19/01/12	mié 01/02/12	14
16		Instalación de los clientes	10 días	jue 02/02/12	mié 15/02/12	15
17		Configuración de la aplicación	7 días	jue 16/02/12	vie 24/02/12	16
18		Aprobación del diseño (Punto de control)	0 días	lun 12/03/12	lun 12/03/12	17
19		Test y pruebas	40 días	lun 12/03/12	vie 04/05/12	
20		Pruebas unitarias	9 días	lun 12/03/12	jue 22/03/12	18
21		Pruebas de integración	9 días	vie 23/03/12	mié 04/04/12	20
22		Pruebas de estrés (Incidencias y riesgos)	5 días	jue 05/04/12	mié 11/04/12	21
23		Documentación de desarrollo y test	5 días	jue 12/04/12	mié 18/04/12	22
24		Aprobación de desarrollo y pruebas (Punto de control)	0 días	vie 04/05/12	vie 04/05/12	23
25		Generación de documentos (memoria del proyecto)	10 días	lun 07/05/12	vie 18/05/12	24
26		Cierre del proyecto	0,5 días	vie 14/09/12	vie 14/09/12	25
27		Defensa del proyecto	1 día	mar 18/09/12	mar 18/09/12	26

Tabla 11: Recursos del proyecto

3.4-4. Calendario temporal

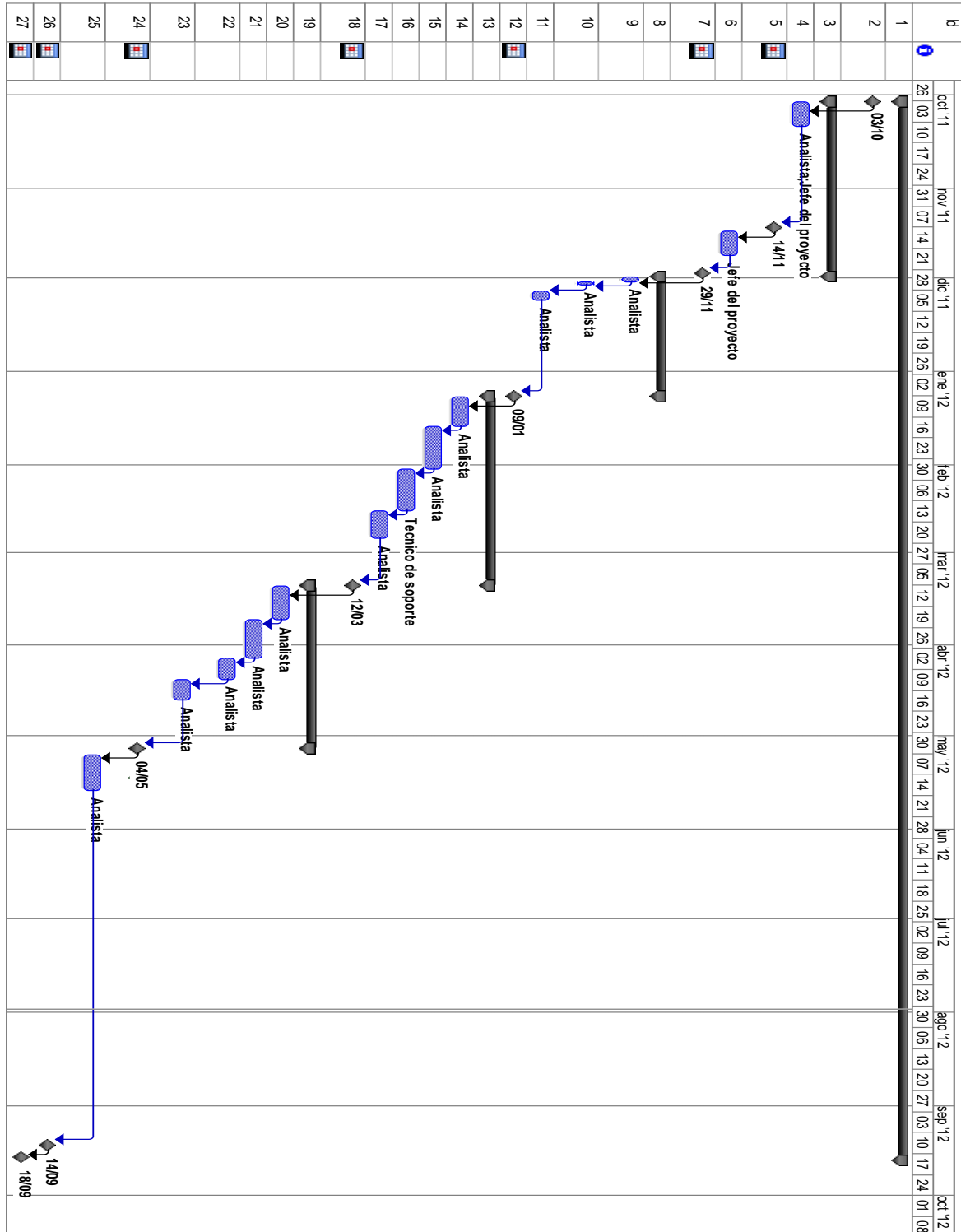


Tabla 12: Diagrama de Gantt

3.5- Evaluación de riesgos

Detectar las vulnerabilidades de un proyecto permite actuar con prevención sobre los posibles fallos, definir estos puntos así como un plan de contingencia para solucionarlos permite un mayor control del proyecto.

En este caso, se detallarán los riesgos que se pueden encontrar durante el transcurso del trabajo necesario para conseguir los objetivos marcados en los puntos anteriores, sus consecuencias y una solución para cada uno de ellos.

3.5-1. Lista de riesgos

Dentro de este proyecto se pueden encontrar diferentes situaciones que harán que no salga todo según lo planificado. Estos casos son:

- R1. Planificación temporal optimista: Plan del proyecto. No se acaba en la fecha prevista, aumentan los recursos.
- R2. Retraso de alguna tarea necesaria: Plan de proyecto. No se cumplen los objetivos del proyecto.
- R3. Presupuesto poco ajustado: Plan de proyecto. Menos calidad, pérdidas económicas.
- R4. Cambios de requisitos: Estudio de viabilidad, análisis. Retraso en la finalización del proyecto, no se cumplen los objetivos del proyecto.
- R5. Equipo del proyecto reducido: Plan de proyecto. Retraso en la finalización del proyecto, no se cumplen los objetivos del proyecto.
- R6. Herramientas en el desarrollo inadecuadas: desarrollo. Retraso en la finalización del proyecto.
- R7. Tareas de desarrollo inadecuadas: Desarrollo. Retraso en la finalización del proyecto, menos calidad en la aplicación.
- R8. No se hace correctamente la fase de test: Desarrollo, implantación. Falta de calidad, deficiencias en la operativa, insatisfacción usuarios.

- R9. Abandono del proyecto antes de la finalización: en cualquier fase. Suspenso del proyecto, frustración.

3.5-2. Catalogación de riesgos

Según el impacto que tengan sobre el proyecto, los riesgos definidos anteriormente se pueden catalogar de la siguiente forma:

	<i>Probabilidad</i>	<i>Impacto</i>
R1	Media	Crítico
R2	Alta	Crítico
R3	Media	Crítico
R4	Media	Medio
R5	Baja	Crítico
R6	Baja	Crítico
R7	Media	Crítico
R8	Media	Crítico
R9	Media	Catastrófico

Tabla 13: Catalogación de Riesgos

3.5-3. Plan de Contingencia

Una vez establecidos cuáles son los posibles riesgos que se pueden encontrar en el transcurso del proyecto se trazará un plan que ayude a resolver cada una de las vulnerabilidades.

<i>Riesgo</i>	<i>Solución</i>
R1	Aplazar o descartar alguna funcionalidad secundaria si el proyecto se encuentra lo suficientemente avanzado para la entrega.
R2	Aplazar o descartar alguna funcionalidad secundaria.
R3	Renegociar con el cliente, afrontar posibles pérdidas, hacer un seguro.
R4	Aplazar o descartar alguna funcionalidad secundaria si el proyecto se encuentra lo suficientemente avanzado para la entrega
R5	Buscar alternativas.
R6	Mejorar la formación del equipo.
R7	Buscar alternativas.
R8	Realizar pruebas según se avanza el desarrollo e ir validando partes con el fin de encontrar el mínimo de errores al finalizar la aplicación
R9	Sin solución

Tabla 14: Planes de Contingencia

3.6- Presupuesto

Dentro del presupuesto podemos contar el costo de personal, el costo de los recursos y otros costes.

3.6-1. Estimación costes de personal

Costes de personal imputables directamente al proyecto.

<i>Nombre del recurso</i>	<i>Tasa estándar</i>	<i>Costo</i>	<i>Tipo</i>
Jefe del proyecto	90,00 €/hora	10.080,00 €	Trabajo
Analista	40,00 €/hora	24.480,00 €	Trabajo
Técnico de soporte	25,00 €/hora	2.000,00 €	Trabajo
Director de proyecto	0,00 €/hora	0,00 €	Trabajo

Tabla 15: Costes de personal

Total: 36560 €

3.6-2. Estimación costo de los recursos

Costes de los recursos imputables directamente al proyecto.

	<i>Costo de Amortización</i>	<i>Costo unitario</i>	<i>Período de amortización</i>
Amortización de MS Office	300	600	36 m
Amortización de MS Project	100	250	36 m

Tabla 16: Coste de recursos

3.6-3. Estimación de otros costes

Servidor Virtual.....2000€

3.6-4. Resumen i análisis coste beneficio

Coste de desarrollo.....	36560 €
Coste de amortización de material.....	400 €
Coste del servidor.....	2000 €

Aunque el coste del proyecto es alto, la reducción de tiempo de los técnicos de soporte para el futuro, es considerable. Además, en la actualidad, la organización es caótica y desde la dirección se requiere un control minucioso de la red.

3.7- Conclusiones

Con este estudio hemos podido comprobar que los objetivos se pueden llegar a cumplir. Se ha determinado las fases, las actividades principales i los puntos de control del proyecto.

Se ha generado el calendario y evaluado los riesgos del proyecto y se ha preparado un plan de contingencia.

Por último se ha determinado el presupuesto del proyecto y los beneficios esperados.

Análisis

4. Análisis

4.1- Introducción

En este capítulo se describirá algunos conceptos teóricos que abarca el desarrollo de este proyecto. Es de vital importancia, que además de explicar en qué consiste un sistema de Gestión de software cliente/servidor, aclaremos conceptos como software libre o GNU/LINUX.

Detallaremos cómo es la red que actualmente dispone la empresa BSA para dar un enfoque global del proyecto.

Además se repasaran las herramientas, lenguajes y tecnologías utilizadas analizándolas brevemente, para acabar exponiendo las cuestiones relacionadas con el diseño de la aplicación.

4.2- Conceptos teóricos

4.2-1. ¿Qué es GNU/Linux?

GNU/Linux es uno de los términos empleados para referirse a la combinación del núcleo o kernel libre similar a Unix denominado Linux con las herramientas del proyecto GNU. Su desarrollo es uno de los ejemplos más prominentes de software libre; todo su código fuente puede ser utilizado, modificado y redistribuido libremente por cualquiera bajo los términos de la GPL (Licencia Pública General de GNU, en inglés: General Public License) y otra serie de licencias libres.

A pesar de que Linux es, en sentido estricto, el sistema operativo, parte fundamental de la interacción entre el núcleo y el usuario (o los programas de aplicación) se maneja usualmente con las herramientas del proyecto GNU o de otros proyectos como GNOME.

Sin embargo, una parte significativa de la comunidad, así como muchos medios generales y especializados, prefieren utilizar el término Linux para referirse a la unión de ambos proyectos.

4.2-2. ¿Qué es el proyecto GNU?

Aunque hayamos comentado el binomio GNU/Linux, merece la pena hacer hincapié y resaltar la importancia del proyecto GNU, ya que habitualmente suele pasarse por alto, sin darle la importancia que le corresponde.

El proyecto GNU fue iniciado por Richard Stallman con el objetivo de crear un sistema operativo completamente libre: el sistema GNU.¹

Actualmente se usan ampliamente variantes del sistema operativo GNU, que usa el núcleo Linux.

4.2-3. ¿Qué es el software libre?

El software libre es la denominación del software que respeta la libertad de los usuarios sobre su producto adquirido y, por tanto, una vez obtenido puede ser usado, copiado, estudiado, modificado, y redistribuido libremente.

Según la Free Software Foundation, el software libre se refiere a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, modificar el software y distribuirlo modificado.

El software libre suele estar disponible gratuitamente, o al precio de costo de la distribución a través de otros medios; sin embargo no es obligatorio que sea así, por lo tanto no hay que asociar software libre a "software gratuito", ya que, conservando su carácter de libre, puede ser distribuido comercialmente.

Tampoco debe confundirse software libre con "software de dominio público". Éste último es aquel software que no requiere de licencia, pues sus derechos de explotación son para toda la humanidad, porque pertenece a

todos por igual. Cualquiera puede hacer uso de él, siempre con fines legales y consignando su autoría original.

4.2-4. Licencia General de GNU

La Licencia Pública General de GNU o más conocida por su nombre en inglés GNU General Public License o simplemente sus siglas del inglés GNU GPL, es una licencia creada por la Free Software Foundation en 1989 (la primera versión, escrita por Richard Stallman), y está orientada principalmente a proteger la libre distribución, modificación y uso de software.

Su propósito es declarar que el software cubierto por esta licencia es software libre y protegerlo de intentos de apropiación que restrinjan esas libertades a los usuarios.

4.2-4.1. GPL versión 3

La versión oficial fue publicada el día 29 de junio de 2007 y es accesible a través del Portal de GNU.

La nueva versión contempla los siguientes aspectos:

- Las diversas formas en que alguna persona podría quitar libertades a los usuarios.
- Prohibir el uso de software libre en sistemas que utilizan la llamada "gestión digital de derechos" o DRM, sistema criticado por la comunidad del software libre.
- Resolver ambigüedades y aumentar la compatibilidad de GPLv3 con otras licencias.
- Facilitar su adaptación a otros países.
- Incluir cláusulas que defiendan a la comunidad de software libre del uso indebido de las patentes de software.
- Mostrar usuarios registrados.

4.2-5. ¿Qué es un sistema de Gestión Centralizado de software?

Administrar una plataforma de varios centenares de equipos de una red es una tarea compleja. Los comportamientos impredecibles de los equipos que corren en Windows, su mantenimiento con la responsabilidad del buen funcionamiento de aplicaciones particulares y de los grupos de trabajo, hace necesario por parte de la Administración de Sistemas de algún software para su administración centralizada.

Algunas funciones básicas si pueden ser realizadas desde el Active Directory, que dispone de un grupo de GPOs a través de las cuales se pueden automatizar muchas tareas de forma centralizada. Pero no es suficiente y se necesita ese nivel que aportan las herramientas referidas que unifican en una consola de administración todas las tareas de mantenimiento software y soporte.

4.3- La red en BSA

Como se ha descrito anteriormente la empresa consta de un hospital, 7 ambulatorios, 1 clínica socio sanitaria, 1 centro de atención a la salud sexual, 1 edificio de consultas externas, 1 almacén y 2 edificios de oficinas.

En el siguiente esquema se puede apreciar con mayor claridad cómo están distribuidas las diferentes sedes de la organización, así como las velocidades entre sedes.

Estas sedes están distribuidas por las poblaciones de Badalona, Montgat y Tiana. Cada sede Dispone de unos 50 hosts, excepto el Hospital que concentra unos 350 hosts.

Como se puede comprobar habiendo tan pocos técnicos de soporte, tantas sedes, y tantos hosts, cada cambio de software es muy costoso y se tarda mucho tiempo.

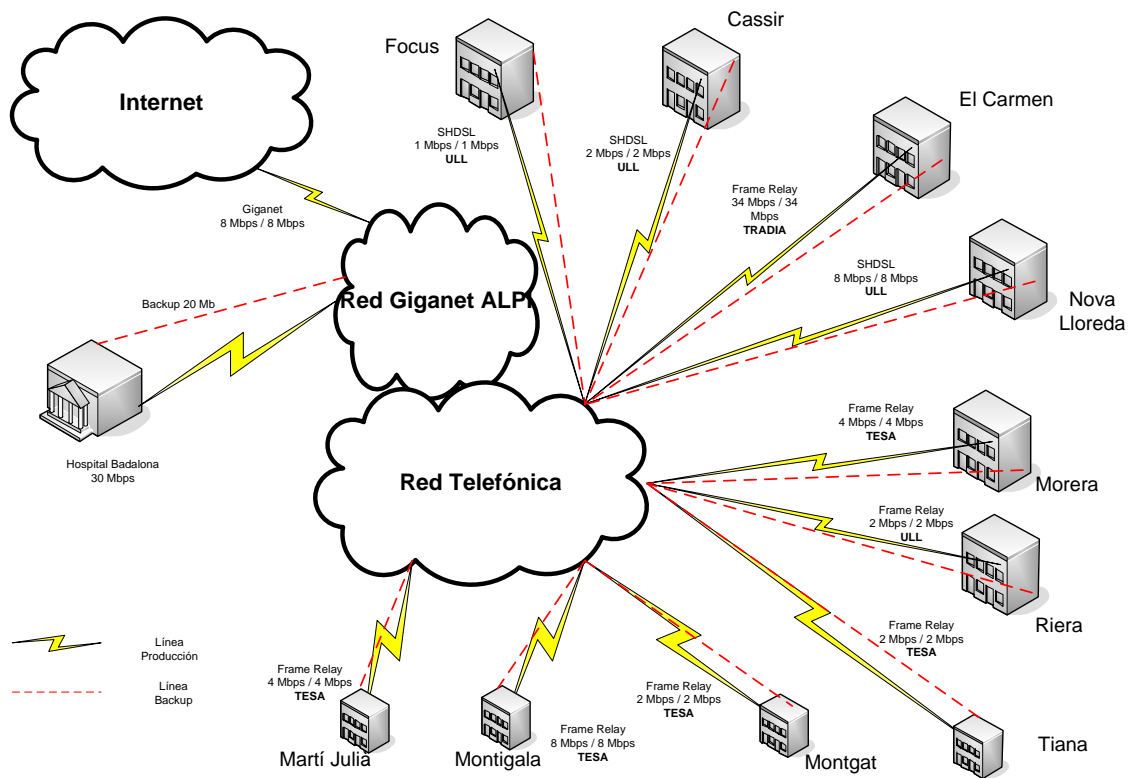


Ilustración 3: Red BSA

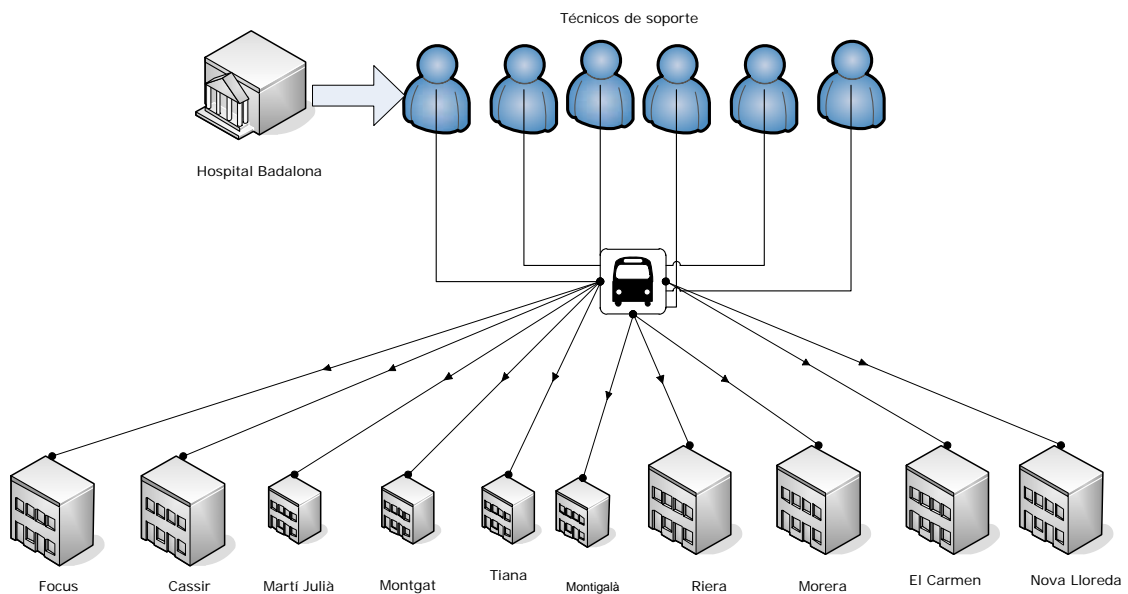


Ilustración 4: Distribución de los técnicos de soporte

4.4- Software utilizado

4.4-1. CentOS

CentOS ha sido la distribución elegida para desarrollar el proyecto. Se ha elegido CentOS 5.8.

CentOS (Community ENTERprise Operating System) es una bifurcación a nivel binario de la distribución Linux Red Hat Enterprise Linux RHEL, compilado por voluntarios a partir del código fuente liberado por Red Hat.

Red Hat Enterprise Linux se compone de software libre y código abierto, pero se publica en formato binario usable (CD-ROM o DVD-ROM) solamente a suscriptores pagados. Como es requerido, Red Hat libera todo el código fuente del producto de forma pública bajo los términos de la Licencia pública general de GNU y otras licencias.

Los desarrolladores de CentOS usan ese código fuente para crear un producto final que es muy similar al Red Hat Enterprise Linux y está libremente disponible para ser bajado y usado por el público, pero no es mantenido ni asistido por Red Hat. Existen otras distribuciones también derivadas de los fuentes de Red Hat.

Con CentOS se obtiene una distribución empresarial que será entre otras cosas:

- Sumamente estable, nada de códigos no probados por el upstream.
- Rápida liberando actualizaciones.
- El soporte de actualizaciones dura 7 años.
- Soporte de muchos programas, comerciales o no, que soportan RedHat Enterprise Linux.
- CentOS tiene varios repos que en suma alcanzan más de los 10mil paquetes. CentOS usa yum para bajar e instalar las actualizaciones, herramienta también utilizada por Fedora.

4.4-2. OPSI

OPSI (Open Pc Server Integration) es un sistema que gestiona y distribuye software para clientes Microsoft Windows, basado en servidores GNU/Linux.

El código fuente de OPSI está licenciado bajo GPLv3, con la excepción de las partes de código que pertenecen a un módulo co-financiado por el período para el que la refinanciación no ha terminado todavía.

El código fuente de OPSI es propiedad de la UIB GmbH, de Alemania.

La estructura de datos de OPSI y los métodos están orientados a objetos. Cada objeto tiene un conjunto de operadores que se pueden utilizar para trabajar con este objeto.

Las características indicadas con (*) son parte de un proyecto de cofinanciación y la tarifa se requiere en este momento.

4.4-2.1. OPSI servidor

- OPSI servidor se ejecuta en Debian, Ubuntu, OpenSuse, SLES, UCS, CentOS, Red Hat Enterprise Linux.
- Puede ser utilizado en máquinas virtuales (VMware / Xen / VirtualBox / KVM) para entornos productivos.
- Backends de datos posibles son: archivo basado, basado en LDAP, MySQL.
- Interfaz gráfica de las estadísticas.

4.4-2.2. Interfaz de gestión

El configed OPSI (editor de configuración) es una interfaz de administración gráfica. Se puede usar en distintas plataformas a través de navegador web ya que se implementa a través de Java.

La lista de OPSI-clientes se pueden clasificar por nombre de cliente, descripción, fecha, número IP, número de inventario, y "visto por última vez".

- Gestión de grupos
- Selección y transformación, según grupos de clientes.
- Guardar y cargar grupos de clientes.
- Jerarquía de gestión de grupos (*).
- Wake on LAN.
- Presentación de software instalable y las versiones.
- Visualización y análisis de las dependencias de los productos y los pedidos de instalación.
- Muestra el progreso de la instalación, resultados de la instalación y la última acción.
- Visualización y edición de propiedades específicas del producto del cliente, que se utilizará durante el proceso de instalación.
- Presentación de los datos de inventario de hardware.
- Presentación de los datos de inventario de software.
- Visualización de los archivos de registro específicos de los clientes por escrito por el servidor, el agente del cliente y la BootImage.
- Documentación de la API.
- Administración por línea de comandos y scripts disponibles.
- Administración por los servicios web (JSON-RPC) disponibles.

4.4-2.3. Instalación de SO

Sistemas operativos compatibles:

- Windows XP (32/64 bits).
- MS-Server 2003 (32/64 bits).
- Windows Vista (32/64 bits).
- MS-Server 2008 (32/64 bits).
- Windows 7 (32/64 bits).
- MS-Server 2008R2 (64 bits).

Permite la instalación a través de PXE-Boot,CD.

Integración sencilla y detección automática de nuevos controladores de hardware de cliente.

4.4-2.4. Implementación de software

- El software cliente se puede asignar a máquinas individuales y / o grupos dedicados de máquinas.
- Instalación automática.
- Software de instalación y desinstalación a través de diferentes métodos.
- Programa de instalación con comandos especiales para:
 - Inicio de los programas de detección de código.
 - Detección de la ejecución del sistema operativo, configuración de idioma, así como la evaluación de los archivos *.ini, archivos de texto, entradas de registro y variables de entorno.
 - Edición de Registro, entradas del menú y de escritorio, archivos INI, archivos XML y archivos de texto.
- Edición de entradas de perfil de usuario específicas, de registro y archivos (en caso de no usar "perfiles móviles").
- Llamadas a programas y scripts externos.
- Copia de archivos con control de versiones.
- Instalación software de 32/64 bits.
- Reinicio del sistema y control de apagado.
- Tareas administrativas.

4.4-2.5. OPSI cliente-agente

- Varios servidores pueden ser configurados para una mayor fiabilidad y reparto de carga.
- Los mensajes pueden ser enviados a los clientes del escritorio (pop-ups).
- El cliente agente de OPSI se conecta en el proceso de inicio de sesión de Windows y se inicia como un servicio del sistema. El software puede ser instalado antes del inicio de sesión del usuario.
- Altamente configurable, con informes de error y advertencias, las

posibilidades de ruptura de la instalación y reinicio del sistema.

- Posibilidad de personalización de un diseño corporativo.
- Las tareas administrativas, por ejemplo, a través de llamadas WMI.

4.4-2.6. Patchmanagement

- Implementación de revisiones sistema operativo (*).
- Implementación de MS-Office revisiones (*).
- Implementación de actualizaciones y parches de seguridad para los paquetes OPSI estándar (*).

4.4-2.7. Inventario de hardware

- Detección detallada de la información de hardware a través de WMI y otros comandos.
- Puede ser configurado qué tipo de información será detectado.
- Detección y almacenamiento de los cambios en el hardware (la historia).
- Almacenamiento de datos en una base de datos SQL con la estructura de datos documentados para la generación de informes de estilo libre.
- Presentación de los datos de los inventarios en la interfaz de administración.
- Consultas sencillas sobre los datos de inventario a través de la interfaz de administración.

4.4-2.8. Inventario de software

- Recolección de datos de inventario de software del registro (Windows).
- Soporte de 64 bits.
- La detección de licensekeys.
- Almacenamiento de datos en una base de datos SQL con la estructura documentada de datos para informes.
- Presentación de los datos de los inventarios en la interfaz de

administración.

4.4-2.9. Soporte de múltiples ubicaciones

- Central de almacenamiento y administración de todos los datos de configuración.
- Repositorios de software descentralizadas con la dirección central.
- Administración de clientes en diferentes lugares mediante una única interfaz de administración central.
- Los datos de configuración para todos los clientes se pueden almacenar en una sola ConfigServer (MasterServer).
- Todos los clientes se conectan a través de servicio web con la MasterServer y obtener la información de configuración a partir de ahí.
- El Software puede ser depositado en los servidores descentralizados y está disponible para la instalación de un servidor central a través de la red.
- Soporte de múltiples servidores de depósito para la instalación y desinstalación de paquetes de software.
- Detección de contradicciones entre depósitos principales y otros depósitos mediante metadatos.
- Los clientes conectados a los depósitos diferentes se pueden gestionar en un solo paquete, si los depósitos implicados están sincronizados (que todos los paquetes de producto estén en versiones idénticas).
- Los clientes se pueden mover de un servidor de almacén a otro lugar a través de una interfaz de gestión.
- El proceso de actualización de paquetes OPSI está diseñado para descargar e instalar paquetes OPSI de un repositorio o de otro servidor OPSI. Puede también ser utilizado para mantener el servidor de depósito en sincronía con el servidor de configuración.

4.4-2.10. Gestión de licencias (*)

- Integración de la gestión de licencias en el editor de configuración OPSI, que es la interfaz de usuario estándar OPSI, para la gestión de la configuración del software de los clientes OPSI.

- Software características de administración de licencias, que son la inserción, la reserva, (automatizado) de asignación, liberación y eliminación de claves de licencia.
- Varios tipos de licencias:
 - Las licencias estándar únicas (una clave de licencia única asignada a un único cliente).
 - Licencias por volumen (una única clave de licencia válida para una cierta cantidad de instalaciones).
 - Las licencias de campus (una única clave de licencia válida para un número ilimitado de las instalaciones dentro de la empresa / sitio).
 - El cliente vinculado licencias (que es una licencia única válida para un cliente específico, por ejemplo, sólo las licencias OEM).
- Liberación de asignación de licencias después de la desinstalación del software correspondiente.
- Manual de mantenimiento opcional de la base de datos de licencias en relación con instalaciones manuales.
- Informes sobre las licencias existentes, utilizando las licencias de acuerdo con la implementación de software y de acuerdo con el inventario de software.

Hay una ventana separada para la gestión de licencias en el editor de OPSI-config. Se puede demostrar haciendo clic en el área del interruptor llamado "licencias" en la ventana principal. Esta ventana sólo está disponible si el módulo de gestión de licencias se activa.

4.4-3. Mysql

MySQL es un sistema de gestión de bases de datos relacional, multihilo y multiusuario. MySQL AB —desde enero de 2008 una subsidiaria de Sun Microsystems y ésta a su vez de Oracle Corporation desde abril de 2009— desarrolla MySQL como software libre en un esquema de licenciamiento dual.

Por un lado se ofrece bajo la GNU GPL para cualquier uso compatible con esta licencia, pero para aquellas empresas que quieran incorporarlo en productos privativos deben comprar a la empresa una licencia específica que les permita este uso. Está desarrollado en su mayor parte en ANSI C.

MySQL es patrocinado por una empresa privada, que posee el copyright de la mayor parte del código.

Se utilizará MySQL como base de datos de la aplicación OPSI.

4.4-4. Webmin

Webmin es una herramienta de configuración de sistemas accesible vía web para OpenSolaris, GNU/Linux y otros sistemas Unix. Con él se pueden configurar aspectos internos de muchos sistemas operativos, como usuarios, cuotas de espacio, servicios, archivos de configuración, apagado del equipo, etcétera, así como modificar y controlar muchas aplicaciones libres, como el servidor web Apache, PHP, MySQL, DNS, Samba, DHCP, entre otros.

Webmin está escrito en Perl, versión 5, ejecutándose como su propio proceso y servidor web. Por defecto se comunica mediante TCP a través del puerto 10000, y puede ser configurado para usar SSL si OpenSSL está instalado con módulos de Perl adicionales requeridos.

Está construido a partir de módulos, los cuales tienen una interfaz a los archivos de configuración y el servidor Webmin. Esto hace fácil la adición de nuevas funcionalidades sin mucho esfuerzo. Debido al diseño modular de Webmin, es posible para cualquier interesado escribir extensiones para configuración de escritorio.

4.5- Diagramas de caso de uso

A continuación se exponen los diagramas de los casos de uso diseñados para establecer los requisitos de comportamiento. El diseño de estos es primordial en el diseño de cualquier proyecto para permitirnos capturar los requisitos potenciales del sistema.

Cada caso de uso nos proporciona uno o más escenarios que indican cómo debería interactuar el sistema con el usuario o con otro sistema para

conseguir el objetivo específico de cada uno.

4.5-1. Login

Nombre:	Login
Fecha:	5.08.2012
Descripción: Permite acceder a la aplicación	
Actores: Operador, Administrador, Técnico de soporte.	
Precondiciones: El actor ha debido ejecutar el navegador con la dirección web.	
Flujo Normal: <ol style="list-style-type: none">1. El actor debe acceder al navegador.2. Debe acceder a la página de la aplicación.3. Introduce el usuario y la palabra clave.4. Es válido y accede a la aplicación.	
Flujo Alternativo: <ol style="list-style-type: none">5. El usuario y contraseña no es correcta.6. Vuelve a pedir usuario y contraseña.	
Post condiciones: <ol style="list-style-type: none">7. Se accede correctamente a la aplicación.	

Tabla 17: caso de uso Login

4.5-2. Añadir host

Nombre:	Añadir Host
Fecha:	5.08.2012
Descripción: Permite añadir un host a la base de datos.	

<p>Actores: Operador, Administrador, Técnico de soporte.</p> <p>Precondiciones:</p> <ol style="list-style-type: none"> 1. El actor ha debido ejecutar el navegador con la dirección web. 2. El actor se ha logado correctamente. 3. El host no existe en la base de datos. <p>Flujo Normal:</p> <ol style="list-style-type: none"> 4. Se accede a la pestaña de añadir host. 5. Se introduce el host. <p>Flujo Alternativo:</p> <ol style="list-style-type: none"> 6. Se accede a la unidad compartida. 7. Se ejecuta el OPSI client agent. <p>Post condiciones:</p> <ol style="list-style-type: none"> 8. El host se reinicia.
--

Tabla 18: caso de uso Añadir host

4.5-3. Eliminar Host

Nombre:	<i>Eliminar Host</i>
Fecha:	5.08.2012
<p>Descripción: Permite eliminar un host de la base de datos.</p> <p>Actores: Operador, Administrador, Técnico de soporte.</p> <p>Precondiciones:</p> <ol style="list-style-type: none"> 1. El actor ha debido ejecutar el navegador con la dirección web. 2. El actor se ha logado correctamente. 3. El host existe en la base de datos. <p>Flujo Normal:</p> <ol style="list-style-type: none"> 4. Se pulsa botón derecho sobre el host y se selecciona eliminar host. <p>Flujo Alternativo:</p>	

Post condiciones:

Tabla 19: caso de uso Eliminar Host

4.5-4. Distribuir Software a Host

Nombre:	<i>Distribuir Software a Host</i>
Fecha:	5.08.2012
Descripción:	Permite distribuir el software instalado en OPSI a los clientes.
Actores:	Operador, Administrador, Técnico de soporte.
Precondiciones:	<ol style="list-style-type: none"> 1. El actor ha debido ejecutar el navegador con la dirección web. 2. El actor se ha logado correctamente. 3. El host existe en la base de datos.
Flujo Normal:	<ol style="list-style-type: none"> 4. Se pulsa botón derecho sobre el host. 5. Se accede al software del host. 6. Se pone en cola el software seleccionado.
Flujo Alternativo:	
Post condiciones:	<ol style="list-style-type: none"> 7. El host se reinicia.

Tabla 20: caso de uso Distribuir Software a Host

4.5-5. Eliminación software de Host

Nombre:	<i>Eliminación Software</i>
Fecha:	5.08.2012
Descripción:	Permite eliminar el software de los clientes.

<p>Actores: Operador, Administrador, Técnico de soporte.</p> <p>Precondiciones:</p> <ol style="list-style-type: none"> 1. El actor ha debido ejecutar el navegador con la dirección web. 2. El actor se ha logado correctamente. 3. El host existe en la base de datos. <p>Flujo Normal:</p> <ol style="list-style-type: none"> 4. Se pulsa botón derecho sobre el host. 5. Se accede al software del host. 6. Se pone en cola de desinstalación el software seleccionado. <p>Flujo Alternativo:</p> <p>Post condiciones:</p> <ol style="list-style-type: none"> 7. El host se reinicia.
--

Tabla 21: caso de uso Eliminación software de Host

4.5-6. Añadir paquete a repositorio

Nombre:	<i>Añadir paquete a repositorio</i>
Fecha:	5.08.2012
<p>Descripción: Permite añadir un nuevo paquete al repositorio de OPSI.</p> <p>Actores: Administrador.</p> <p>Precondiciones:</p> <ol style="list-style-type: none"> 1. El actor ha debido ejecutar la consola ssh. 2. El actor se ha logado correctamente. 3. El paquete no está instalado. <p>Flujo Normal:</p> <ol style="list-style-type: none"> 4. Tenemos el paquete descargado correctamente 5. Ejecutamos el instalador. <p>Flujo Alternativo:</p>	

Post condiciones:

Tabla 22: caso de uso Añadir paquete a repositorio

4.5-7. Eliminar paquete de repositorio

Nombre:	<i>Eliminar paquete a repositorio</i>
Fecha:	5.08.2012
Descripción: Permite eliminar un nuevo paquete al repositorio de OPSI.	
Actores: Administrador.	
Precondiciones: <ol style="list-style-type: none"> 1. El actor ha debido ejecutar la consola ssh. 2. El actor se ha logado correctamente. 3. El paquete se encuentra instalado. 	
Flujo Normal: <ol style="list-style-type: none"> 4. Ejecutamos el instalador para desinstalarlo. 	
Flujo Alternativo:	
Post condiciones:	

Tabla 23: caso de uso Eliminar paquete de repositorio

4.5-8. Distribuir hardware

Nombre:	<i>Distribuir hardware</i>
Fecha:	5.08.2012
Descripción: Permite eliminar un nuevo paquete al repositorio de OPSI.	
Actores: Operador, Administrador, Técnico de soporte.	
Precondiciones:	

<p>5. El actor ha debido ejecutar el navegador con la dirección web.</p> <p>6. El actor se ha logado correctamente.</p> <p>7. El host existe en la base de datos.</p> <p>Flujo Normal:</p> <p>8. Se pulsa botón derecho sobre el host.</p> <p>9. Se accede al hardware del host.</p> <p>10. Se pone en cola el software asociado al hardware seleccionado.</p> <p>Flujo Alternativo:</p> <p>Post condiciones:</p> <p>11. El host se reinicia.</p>
--

Tabla 24: caso de uso Distribuir hardware

4.5-9. Eliminar hardware

Nombre:	<i>Eliminar Hardware</i>
Fecha:	5.08.2012
Descripción: Permite distribuir drivers referentes a un hardware.	
Actores: Operador, Administrador, Técnico de soporte.	
Precondiciones:	
<p>1. El actor ha debido ejecutar el navegador con la dirección web.</p> <p>2. El actor se ha logado correctamente.</p> <p>3. El host existe en la base de datos.</p>	
Flujo Normal:	
<p>4. Se pulsa botón derecho sobre el host.</p> <p>5. Se accede al hardware del host.</p> <p>6. Se pone en cola de desinstalación el software del hardware seleccionado.</p>	
Flujo Alternativo:	
Post condiciones:	
<p>7. El host se reinicia.</p>	

Tabla 25: caso de uso Eliminar hardware

Implementación

5. **Implementación**

5.1- **Introducción**

A lo largo de este capítulo, comentaremos los aspectos más técnicos relativos a la implementación del proyecto.

Comenzaremos detallando la arquitectura de la aplicación y su funcionamiento desde un punto de vista más general, para ir profundizando en cada uno de los apartados más significativos de la misma.

Una vez vista la arquitectura, nos centraremos las herramientas OPSI, los ficheros y la interfaz de usuario.

5.2- **Arquitectura**

Como hemos visto en capítulos anteriores, el proyecto consta de una aplicación central, y unas aplicaciones cliente. Todas ellas desarrolladas en python.

La configuración de OPSI requiere gestionar datos. Todos los componentes que no hacen de servidor usan un servicio web para el intercambio de datos con el servidor OPSI. Estos intercambian datos a través del `opsiconfd`, y la `opsiconfd` reenvía los datos al gestor de backend que pasa los datos al backend seleccionado.

OPSI soporta diferentes sistemas de apoyo (Backend):

- Basado en Archivo.
- Basado en LDAP.
- Basado en MySQL.

En nuestro caso se ha optado por utilizar una base de datos en MySQL, porque nos permite una forma más sencilla para manipular los datos y permite de una manera ágil tratar las copias de seguridad.

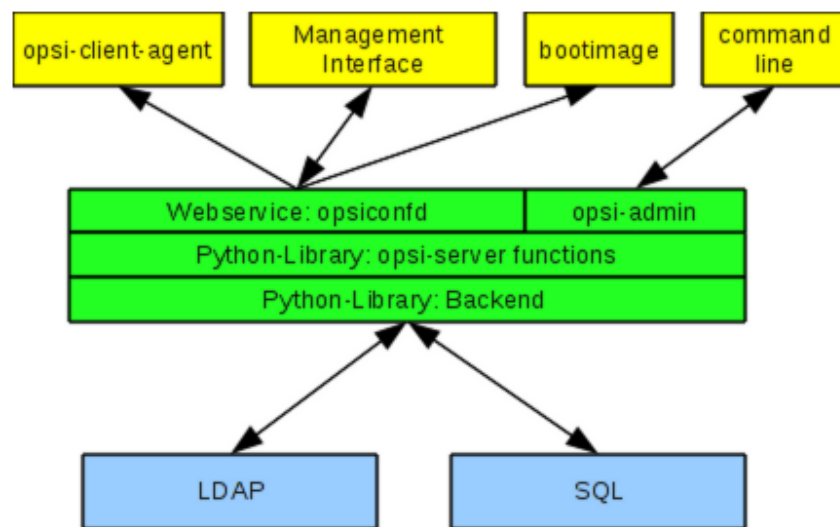


Ilustración 5: Esquema OPSI

Los archivos de configuración que se encuentran en `/etc/opsi/backends` definen los backends.

Qué backend se utiliza para que los datos, están configurados en el archivo `/etc/opsi/backendManager/dispatch.conf`. El archivo `/etc/opsi/backendManager/acl.conf` define quién tiene acceso a qué métodos.

Por debajo del directorio `/etc/opsi/backendManager/extend.d` pueden haber archivos que definen los métodos extendidos OPSI.

5.3- Servidor

La empresa dispone de un sistema BladeSystem Proliant. Dentro de éste sistema, tenemos varias máquinas virtualizadas.

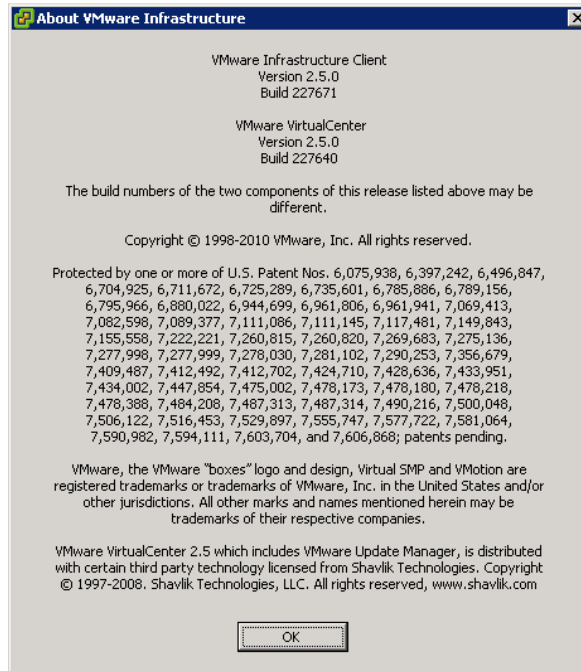


Ilustración 6: VMware en BSA

Se ha creado una máquina virtual llamada servdesk dentro del dominio bsa.es.

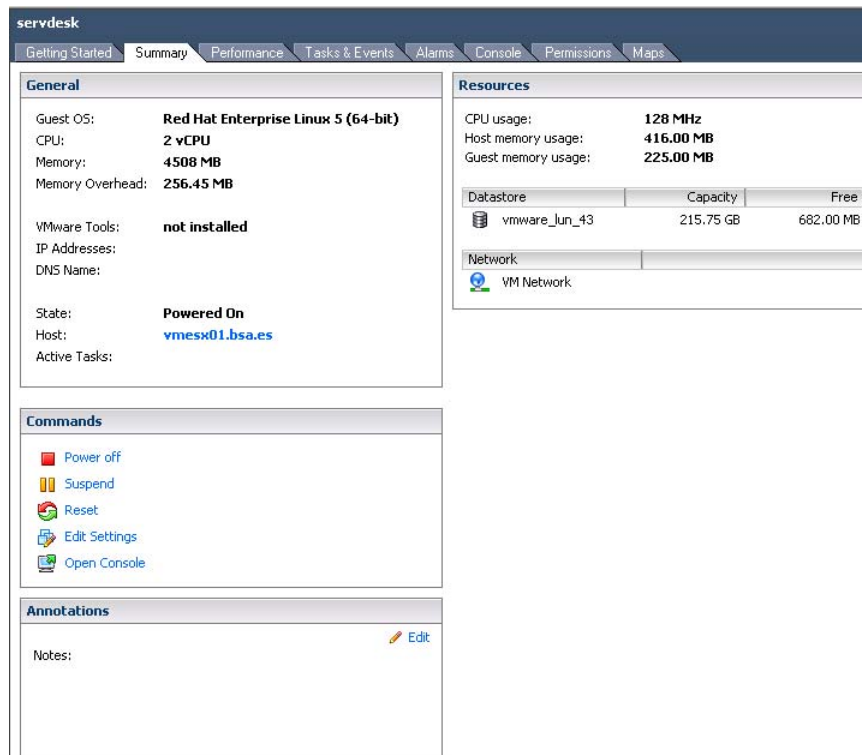


Ilustración 7: servdesk

Ésta máquina se encuentra dentro del dominio bsa.es y tiene instalado un gestor de base datos MySQL, OPSI, JAVA, la herramienta de configuración Webmin y un servicio samba para tener el repositorio que accederán los clientes.

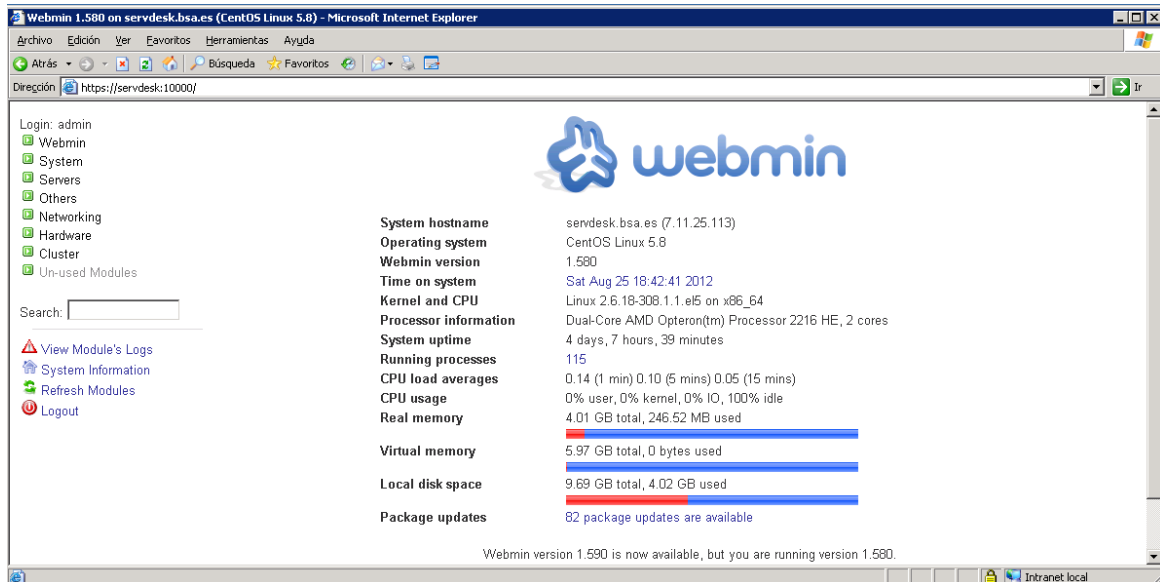


Ilustración 8: Webmin en servdesk

5.4- Herramientas OPSI

5.4-1. Herramienta: OPSI-setup

Este programa es algo así como la navaja suiza de la configuración OPSI. Es utilizado por las secuencias de instalación OPSI y puede ser también llamado por separado para mantenimiento y reparación.

Las tareas de OPSI-setup son:

- registrar un OPSI-servidor como servidor de almacén.
- correctos derechos de acceso a archivos.
- inicializar backends de almacenamiento de datos.

- actualizar backend.
- configuración del MySQL- backend.
- editar las configuraciones predeterminadas.
- limpieza del backend actual (s).
- configurar las acciones esenciales de samba.
- configurar las entradas esenciales dhcp.

El comando `opsi-setup --help` muestra las opciones del programa.

5.4-2. Herramienta: Interfaz de gestión *OPSI-configed*

OPSI-configed requiere Java 1.6 opiconfd corriendo en el servidor.

La mayoría de las veces OPSI-configed se llamará como applet en el navegador a través de: `https://servdesk:4447/configed`. Para un inicio de sesión satisfactorio el usuario siempre tiene que ser un miembro de *opsiadmin* unix-group.

La OPSI-configed como aplicación es también parte del producto OPSI OPSI-adminutils y entonces se puede iniciar a través del menú ventanas de inicio. En el servidor de la OPSI-configed está instalado como parte de la instalación OPSI-servidor. Puede ponerse en marcha con la entrada del menú o con el comando `/usr/bin/opsi-configed`.

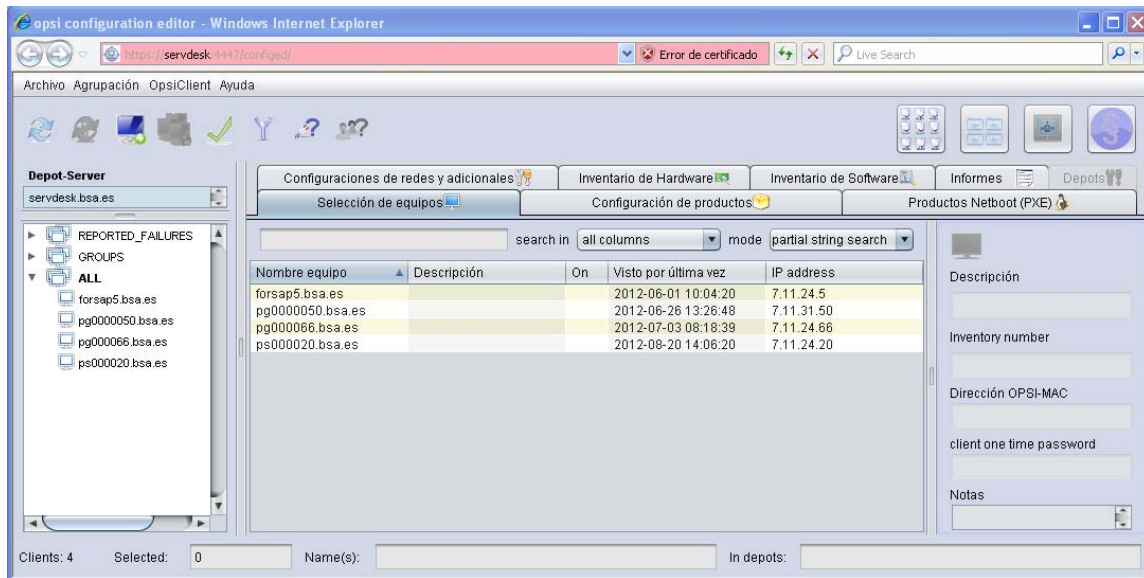


Ilustración 9: OPSI configed en servdesk

5.4-3. Herramienta: opsi-package-manager

Opsi-package-manager se usa para (des) instalación de productos OPSI en el servidor OPSI. Con el fin de instalar un paquete OPSI, este deberá ser legible para el usuario opsiconfd del sistema OPSI.

El archivo de registro del paquete opsi-package-manager se encuentra en: /var/log/OPSI/package.log

```

servdesk - default - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles

[root@servdesk webmin]# opsi-package-manager -l
-----
- servdesk.bsa.es -
-----
Product ID          Version            Name
-----
config-win-base     4.0.1-1           Windows Customizing
documentfoundation.libreoffice 3.5.3-go4         LibreOffice Productivity Suite
hwaudit             4.0.4-1           Client Hardwareinventarisierung
hwinvent            4.0-1             Hardware Inventarisierung
javavm              1.6.0.20-2        Sun Java Runtime Environment
jediit              4.5-3             jEdit programmer's text editor
memtest86           3.5b-1            Memtest86+
ntfs-restore-image  4.0-1             NTFS Image zurückschreiben
ntfs-write-image    4.0-1             NTFS Image erstellen
opsi-adminutils     4.0.1.12-1        some administration tools (not
opsi-client-agent   4.0.1-26          opsi.org client agent
opsi-template       4.0.1-5           opsi template product
opsi-template-with-admin 1.0-8            opsi template for installs with
opsi-winst          4.11.2.5-1        winst
opsi-winst-test     4.11.2-3          opsi-winst test
shutdownwanted      1.0-4             shutdown wanted
swaudit             4.0-11            Client software inventory
win2003             4.0.1-3           Windows 2003 Standard Edition -
win2003-r2-x64     4.0.1-3           Windows Server 2003 R2 - x64 -
win2008             4.0.1-11          Windows Server 2008 - x86
win2008-r2         4.0.1-11          Windows Server 2008 R2 - x64
win2008-x64        4.0.1-11          Windows Server 2008 - x64
win2k               4.0.1-3           Windows 2000 professional - x86
win7                4.0.1-11          Windows 7 - x86
win7-x64           4.0.1-11          Windows 7 - x64
winvista           4.0.1-11          Windows Vista - x86
winvista-x64       4.0.1-11          Windows Vista - x64
winxppro           4.0.1-3           Windows XP professional - x86 -
winxppro-x64      4.0.1-3           Windows XP professional - x64 -
wipedisk           1.0-2             Festplatte oder Partitionen loe

[root@servdesk webmin]#
Connected to servdesk                               SSH2 - aes128-cbc - hmac-md5 - none 122x38 NUM

```

Ilustración 10: opsi-package manager en servdesk

5.4-4. Herramienta: opsi-product-updater

La utilidad de línea de comandos `opsi-product-updater` está diseñada para descargar e instalar paquetes cómodos OPSI de un repositorio o en otro servidor OPSI. Puede también ser utilizado en una tarea programada para mantener el servidor de almacén en sincronización con el servidor de configuración.

Las principales características son:

- repositorios configurables.
- acciones configurables.

Toda la configuración se realiza en el fichero de configuración `/etc/opsi/opsi-product-updater.conf`.

5.4-5. Herramientas: `opsi-admin` / *interfaz configuración OPSI*

OPSI V3 introdujo una biblioteca OPSI propiedad python que proporciona una API para la configuración OPSI. `opsiconfd` proporciona esta API como un servicio web, mientras que `opsi-admin` es la interfaz de línea de comandos para esta API.

`opsi-admin` puede utilizar el servicio web o directamente OPSI operar en el backend de datos. Para trabajar con el servicio web tiene que proporcionar la dirección y un nombre de usuario y contraseña.

En la línea de comando `opsi-admin` proporciona una interfaz para la OPSI-API. Hay un modo interactivo y un modo no interactivo para el procesamiento por lotes dentro de scripts.

5.5- Ficheros de configuración

En este apartado se describirán los ficheros más importantes y la configuración que tienen.

5.5-1. `opsiconfd.conf`

El `opsiconfd` proporciona la API como JSON servicio web y tareas importantes. Por lo tanto el `opsiconfd` es el servicio OPSI central y hace toda la comunicación con los clientes.

```
/etc/opsi/opsiconfd.conf
```

```
[global]
```

```
backend config dir = /etc/opsi/backends
dispatch config file = /etc/opsi/backendManager/dispatch.conf
extension config dir = /etc/opsi/backendManager/extend.d
acl file = /etc/opsi/backendManager/acl.conf
admin networks = 0.0.0.0/0
multiprocessing = no
pid file = /var/run/opsiconfd/opsiconfd.pid
log file = /var/log/opsi/opsiconfd/%m.log
symlink logs = yes
log level = 9
log format = [%I] [%D] %M (%F|%N)
max execution statistics = 250
```

[service]

```
interface = 7.11.25.113
http port = 0
https port = 4447
```

[session]

```
session name = OPSISID
verify ip = no
update ip = no
max inactive interval = 120
max authentication failures = 5
```

[directories]

```
/ = /usr/share/opsiconfd/static (noauth)
configed = /usr/lib/configed (noauth)
```

Si modificamos el este fichero, deberemos reiniciar el servicio opsiconfd.

```
service opsiconfd restart
```

5.5-2. smb.conf

Éste fichero contiene la información del servidor samba.

```
/etc/samba/smb.conf
```

[global]

```
name resolve order = wins lmhosts hosts
idmap gid = 15000-20000
smb ports = 445
socket options = TCP_NODELAY SO_RCVBUF=8192 SO_SNDBUF=8192
wins server = 7.11.25.158
encrypt passwords = yes
winbind use default domain = yes
realm = BSA.ES
wins proxy = yes
dns proxy = yes
netbios name = servdesk
server string = Servidor OPSI
idmap uid = 15000-30000
disable netbios = yes
password server = 7.11.25.158
workgroup = BSA
security = ads
preferred master = no
winbind enum users = yes
winbind enum groups = yes
```

[datos]

```
comment = Unidad Virtual datos en Aries-2
path = /datos/public
writeable = yes
valid users = @"BSA\Usersmbhmb"
admin users = @"BSA\smbadminssc"
force create mode = 0777
```

[opt_pcbn]

```
available = yes
comment = opsi depot share
path = /opt/pcbun
oplocks = no
follow symlinks = yes
level2 oplocks = no
writeable = yes
invalid users = root
```

[opsi_depot]

```
oplocks = no
```

```
invalid users = root
path = /var/lib/opsi/depot
comment = opsi depot share (ro)
valid users = @"BSA\smbadminssc"
browsable = yes
level2 oplocks = no
```

[opsi_config]

```
available = yes
comment = opsi config share
path = /var/lib/opsi/config
writeable = yes
invalid users = root
```

[opsi_workbench]

```
available = yes
comment = opsi workbench
path = /home/opsiproducts
writeable = yes
invalid users = root
create mask = 0660
directory mask = 0770
```

5.5-3. site.py

Este fichero contiene la configuración entre otros de la codificación de python.

```
/usr/lib64/python2.4/site.py
```

```
def setencoding():
    """Set the string encoding used by the Unicode implementation. The
    default is 'ascii', but if you're willing to experiment, you can
    change this."""
    encoding = "utf-8" # Default value set by _PyUnicode_Init()
    if 0:
        # Enable to support locale aware default string encodings.
        import locale
        loc = locale.getdefaultlocale()
        if loc[1]:
            encoding = loc[1]
    if 0:
        # Enable to switch off string to Unicode coercion and implicit
```

```
# Unicode to string conversion.
encoding = "undefined"
if encoding != "ascii":
    # On Non-Unicode builds this will raise an AttributeError...
    sys.setdefaultencoding(encoding) # Needs Python Unicode build !
```

5.5-4. **dispatch.conf**

Este archivo configura qué métodos son enviados los backends correspondientes.

```
/etc/opsi/backendManager/dispatch.conf
```

```
backend_.*      : file, mysql, opsipxeconfd
host_.*        : file, opsipxeconfd
productOnClient_.* : file, opsipxeconfd
configState_.* : file, opsipxeconfd
license.*      : mysql
softwareLicense.* : mysql
audit.*        : mysql
.*             : file
```

Si modificamos el este fichero, deberemos reiniciar el servicio opsiconfd.

```
service opsiconfd restart
```

5.6- **Funcionamiento cliente servidor**

OPSI está basado en una arquitectura cliente-servidor. En cada una de las máquinas Windows que se quiera gestionar a través de Opsi se instalará un agente (wInst - Windows Installer). A su vez, dentro del servidor se depositará el software que se quiera desplegar en dichas máquinas y éste recogerá de forma centralizada la información de todos sus nodos.

Cuando arranque el sistema Windows, el cliente se logeará en el módulo `preLoginLoader` del servidor, y examinará si existe alguna actualización disponible. En caso afirmativo, `winst` comenzará de forma automática la tarea de instalación. El servidor proporcionará los scripts de

instalación y los paquetes software a través de un espacio de almacenamiento compartido (Samba).

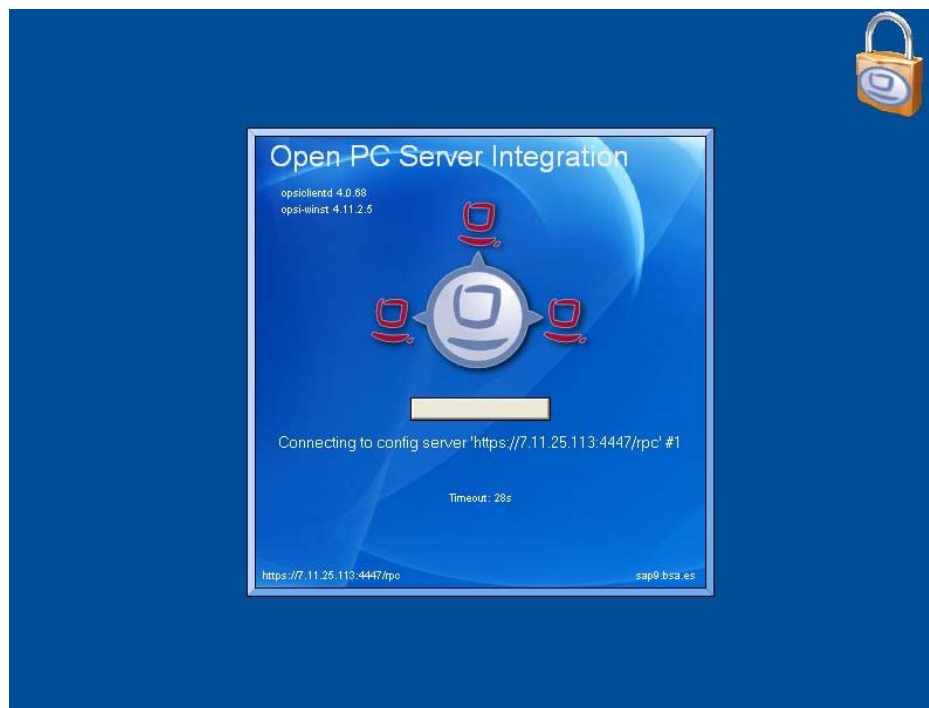


Ilustración 11: Conexión cliente con servidor servdesk

`opsi-client-agent` se instala en `%ProgramFiles%\opsi.org\opsi-client-agent`.

Este directorio contiene todos los programas OPSI, como por ejemplo, `opsiclientd`, el notificador `opsiclientd`, el `opsi-winst` y algunas bibliotecas necesarias. También encontraremos aquí los archivos de configuración y plantillas gráficas (skins) de los programas mencionados.

El directorio `%ProgramFiles%\opsi.org\opsi-client-agent` está protegido contra la manipulación por parte de los usuarios sin privilegios de administrador.

El directorio `%ProgramFiles%\opsi.org\opsi-client-agent\opsiclientd` contiene el archivo de configuración de `opsiclientd` y se necesitan privilegios de administrador para poder leerlo.

El directorio `c:\opsi.org` es usado para almacenar en caché los archivos de instalación y datos.

5.7- Interfaces

Como hemos comentado tenemos dos tipos de interfaces. Por línea de comando y por navegador web.

5.7-1. Línea de comando

Esta es la manera de acceder al servidor directamente. Accederemos vía ssh. Para la instalación del servidor, configuración de Java, de samba e instalación de OPSI esta será la manera de conectarse al servidor.

Herramientas como `opsi-package-manager`, `opsi-product-updater`, `opsi-setup`, `opsi-admin` solo se pueden configurar mediante ssh.

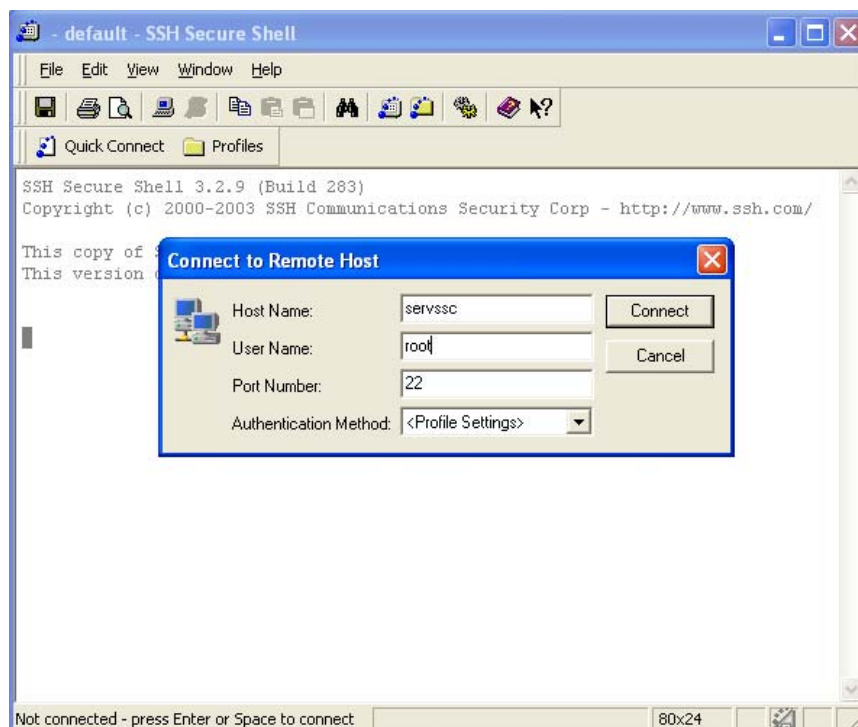
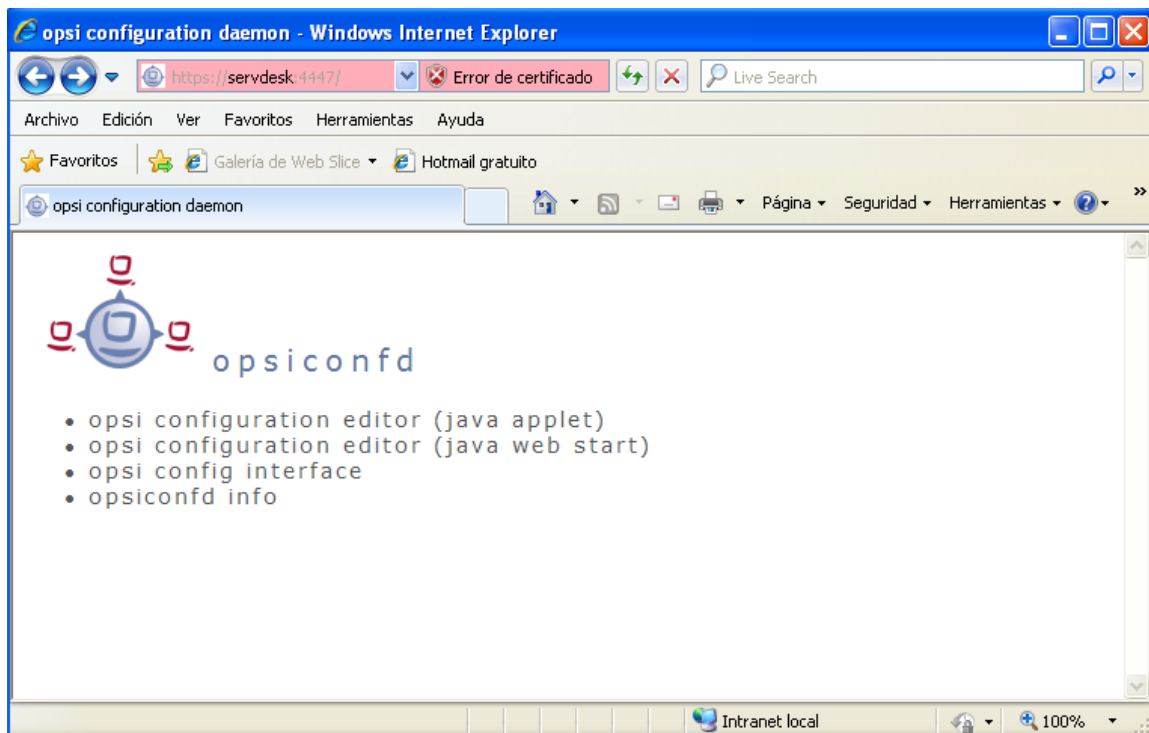


Ilustración 12: Conexión servdesk mediante ssh

5.7-2. Navegador web

Esta es la manera de acceder desde navegador web. Nos proporciona:

- Java applet para la configuración de equipos.
- La interfaz de configuración
- Opsiconfd info que nos muestra graficas detalladas de uso de CPU y valores de la configuración del sistema OPSI.



Conclusiones

6. Conclusiones

6.1- Introducción

En el último capítulo se presentan las conclusiones extraídas del estudio y desarrollo del proyecto.

Primero se indican los objetivos que se han conseguido de lo planteado, los problemas que han surgido durante el desarrollo de la aplicación y el modo de resolverlos.

Se indican las líneas a seguir del proyecto empresarial.

Finalmente, se expone una valoración general sobre la experiencia al plantear y desarrollar este proyecto.

6.2- Conclusiones

La finalización del proyecto se considera como buena, ya que tanto el planteamiento, como la metodología y el diseño se han mantenido. Se ha logrado adaptar desde un punto de inicio, una estructura sólida, ampliable y configurable para cualquier administrador que siga la parametrización establecida.

6.3- Futuras ampliaciones

Han sido bastantes las líneas de posibles ampliaciones futuras que se han ido detectando durante el desarrollo del proyecto. Esto es debido, como hemos comentado, a que el proyecto empresarial excede del proyecto universitario.

- Extender el cliente a todos los hosts de la empresa.

- Instalar todo el software de la empresa como paquetes OPSI.
- Instalar todo el hardware de la empresa como paquetes OPSI.
- Distribuir imágenes de S.O. en los hosts.
- Controlar las licencias.
- Encendido de hosts remotamente.

6.4- **Problemas encontrados**

Instalación: El problema principal. Esta aplicación ha requerido muchas horas de dedicación, debido a la gran cantidad de condicionantes y de integración. La instalación de CentOS, la configuración de samba, la instalación de webmin, de Java y por último de OPSI, que todo esto funcionase dentro de la red específica de BSA.

Documentación: No tiene soporte en castellano. Gran cantidad de dudas y problemas técnicos que principalmente están en foros alemanes.

Licencia: La licencia del código es GPL – dispone de una serie de módulos que no son gratuitos, que tras el concepto de cofundadores, no serán liberados hasta que no sean totalmente financiados por ellos, mientras tanto, únicamente aquellos que hagan la donación estipulada podrán disfrutar de los módulos. Dispone de un periodo de evaluación para probar estos módulos antes de tener que pagarlos.

Abasto del proyecto: Dado el tamaño de la empresa y al tener que satisfacer a la vez un proyecto empresarial y un proyecto académico, se ha optado finalmente de hacer una muestra de varios hosts con alguna aplicación ya que finalizar un proyecto de este tipo está fuera del ámbito académico por envergadura. A partir de ahora queda un trayecto donde las bases están puestas y quedan para futuras ampliaciones.

6.5- Valoración General

La realización de este proyecto ha resultado una gran satisfacción personal, ya que he tenido la oportunidad de aprender entre otras cosas una metodología a seguir para la implementación de un proyecto de sistemas, el estudio previo, muchas decisiones a tener en cuenta y una cosa primordial, una idea clara del objetivo a seguir.

Debido a la ilusión y la gran satisfacción que presenta, dentro de una empresa, desarrollar un sistema e integrarlo en el ámbito global, seguir adquiriendo nuevos conocimientos para mejorar y obtener mayores y mejores resultados.

Por lo tanto la valoración final es muy positiva, dado que se han cumplido en mayor o menor medida todos los objetivos establecidos inicialmente.

Bibliografía

7. **Bibliografía**

7.1- **Direcciones de internet**

opsi (open pc server integration) the open source client management system

Disponible:

<http://www.opsi.org/en>

Último acceso (23-08-2012)

www.centos.org - The Community Enterprise Operating System

Disponible

www.centos.org

Último acceso (20-08-2012)

uib gmbh

Disponible

<http://uib.de/en/home/index.html>

Último acceso (20-08-2012)

Wikipedia

Disponible

<http://es.wikipedia.org/>

Último acceso (23-08-2012)

Glosario

8. Glosario

Patchmanagement: herramienta que busca mantener al día los parches, actualizaciones y Service Packs de software instalado en las estaciones de trabajo de una organización.

Red: es un conjunto de equipos informáticos y software conectados entre sí por medio de dispositivos físicos que envían y reciben impulsos eléctricos, ondas electromagnéticas o cualquier otro medio para el transporte de datos, con la finalidad de compartir información, recursos y ofrecer servicios.

Opsi: es un cliente de código abierto del sistema de gestión de los clientes de Windows y se basa en servidores Linux

S.O.: es un programa o conjunto de programas que en un sistema informático gestiona los recursos de hardware y provee servicios a los programas de aplicación, ejecutándose en modo privilegiado respecto de los restantes.

Java: es un lenguaje de programación de alto nivel orientado a objetos.

Navegador: es una aplicación que opera a través de Internet, interpretando la información de archivos y sitios web para que podamos ser capaces de leerla (ya se encuentre ésta alojada en un servidor dentro de la World Wide Web o en un servidor local).

Software: equipamiento lógico o soporte lógico de un sistema informático, comprende el conjunto de los componentes lógicos necesarios que hacen posible la realización de tareas específicas, en contraposición a los componentes físicos, que son llamados hardware.

Hardware: corresponde a todas las partes tangibles de un sistema informático; sus componentes son: eléctricos, electrónicos, electromecánicos y mecánicos.

Licencia: es un contrato mediante el cual una persona recibe de otra el

derecho de uso, de copia, de distribución, de estudio y de modificación (en el caso del Software Libre) de varios de sus bienes, normalmente de carácter no tangible o intelectual, pudiendo darse a cambio del pago de un monto determinado por el uso de los mismos.

Host: es usado en informática para referirse a las computadoras conectadas a una red, que proveen y utilizan servicios de ella.

Samba: es una implementación libre del protocolo de archivos compartidos de Microsoft Windows (antiguamente llamado SMB, renombrado recientemente a CIFS) para sistemas de tipo UNIX.

Webmin: es una herramienta de configuración de sistemas accesible vía web para OpenSolaris, GNU/Linux y otros sistemas Unix.

Applet: Un applet es un componente de una aplicación que se ejecuta en el contexto de otro programa, por ejemplo un navegador web.

LDAP: son las siglas que hacen referencia a un protocolo a nivel de aplicación el cual permite el acceso a un servicio de directorio ordenado y distribuido para buscar diversa información en un entorno de red.

Sabadell 17 de septiembre de 2012

Sergio Sánchez Gutiérrez