

**Universidad de las Ciencias Informáticas**

**Facultad 1**

**Título:** Módulo de administración del servicio Proxy para HMAST.

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

**Autor:** Alexander Pérez Tasé

**Tutores:** Ing. Amaury Viera Hernández

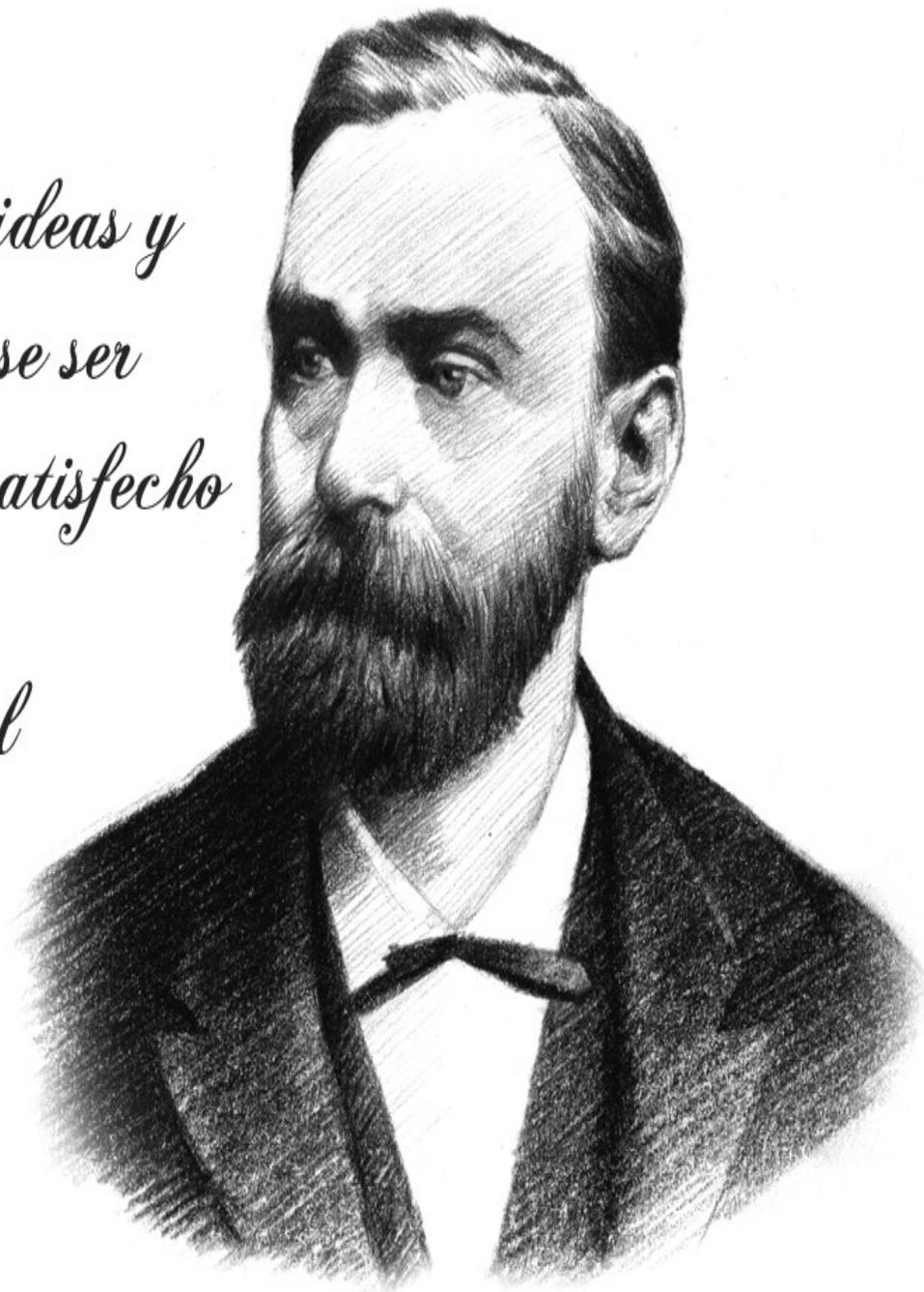
Ing. Alfredo Pérez Benitez

**La Habana**

**Junio 2013**

*Si tuviera mil ideas y  
sólo una resultase ser  
buena, estaría satisfecho*

*Aldred Nobel*  
(1833 - 1896)



**Declaración de Autoría**

Declaro ser el único autor de este trabajo y autorizo a la Universidad de las Ciencias Informáticas a hacer uso exclusivo del mismo en su beneficio. Para que así se conste firmo la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del año \_\_\_\_\_.

\_\_\_\_\_  
Alexander Pérez Tasé (Autor)

\_\_\_\_\_  
Ing. Amaury Viera Hernández (Tutor)

\_\_\_\_\_  
Ing. Alfredo Pérez Benitez (Tutor)

**Dedicatoria**

*A mi mamá por toda su entrega en la formación de sus dos hijos, su ejemplo y por sobre todas las cosas su amor.*

*A mi tía Magalis por ser mi segunda madre y guiarme por el buen camino.*

*A mi hermano Karel, para que el próximo año sea él quién escriba la dedicatoria.*

*A todas las personas que a lo largo de toda mi vida me han ayudado a ser lo soy hoy,*

## **Agradecimientos**

A mi mamá por todo su sacrificio, su ejemplo y sobre todo, por el amor brindado en estos años.

A mi tía Magalis por ser mi segunda madre y tener un peso importante en mi crecimiento y formación como persona.

A mi hermano Karel por su forma tan peculiar de darme consejos y siempre estar bromeando, aunque el momento no lo requiera.

A mi padre por su apoyo.

A Blas, por ser como un padre para nosotros y haber ayudado a mi mamá en el tiempo que más difícil era la situación. Muchas de las cosas que hoy tenemos son gracias ti.

A mi prima Katy por haberme apoyado en estos 5 años de la carrera.

A Juan y Yasiel por ser amigos en todo el sentido de la palabra, estar en las buenas y malas.

A las personas que se han convertido en imprescindibles y han sabido provocar una sonrisa cuando más me hacía falta, Fernando, Roberto, Joel, Henry, Felix, Yosle y Daniel. Gracias por todo.

A las personas que en estos años en la UCI de una forma u otra han estado cuando he necesitado una mano, no me han dicho que no, Yenny, Yinet, Osbel. En especial a Adrianet por ayudarme más que nadie aunque no piense así.

A mis compañeros del aula, Yadiel, Nurisel, Merlyn y todos los que han compartido conmigo estos 5 años. A Adianys, Yaris, que aunque no terminaron con nosotros siempre los recordamos con mucho cariño.

A mis compañeros del cuarto, Raciél, El Bolo, Ripoll, Diorge, los Chacha

A mis amigos de Bayamo, Yuri, Leandro, Alfre, Mirta, Nelsido, Reinier.

A mi profesor Henry, quien fue como un padre en los años que estuve en la EIDE y ESPA.

A mis tutores Alfredo y Amaury por confiar en mi.

## **Resumen**

El proceso de migración hacia tecnologías libres, ejecutado actualmente en el país y encabezado por el Departamento de Servicios Integrales en Migración, Asesoría y Soporte (SIMAYS), requiere en la mayoría de las instituciones involucradas, la migración desde plataformas privativas hacia plataformas libres de los servicios telemáticos que estas brindan. Una vez migrados estos servicios, son necesarias herramientas que permitan administrar los mismos, sin embargo, las aplicaciones existentes no cumplen, debido a las pocas prestaciones que brindan los servidores en estas instituciones, con varios aspectos importantes de estos servicios como son usabilidad y rendimiento. Con el objetivo de resolver estos problemas, se desarrolló en 2011 la Herramienta para la Migración y Administración de los Servicios Telemáticos (HMAST), que actualmente cuenta con los módulos para administrar el servicio DHCP y los que implementan el protocolo SSH.

En la presente investigación se desarrolla un módulo para HMAST, basado en tecnologías libres, que permite la administración del servicio proxy en instituciones migradas. Para ello se analizan los servidores proxy existentes en entornos de código abierto, seleccionando el servidor Squid como el más óptimo, así como las herramientas que administran dicho servicio. Además, se documentan las tecnologías, herramientas, lenguajes a utilizar en la construcción del módulo y la definición de los elementos necesarios para el exitoso desarrollo del mismo.

**Palabras claves:** administración, HMAST, servicio proxy, Squid, tecnologías libres.

## Índice

Introducción.....	1
Capítulo 1: Servicios Proxy.....	5
1.1 Conceptos asociados al servicio proxy.....	5
1.2 Funcionamiento de un servidor proxy.....	6
1.3 Servidores proxy de código abierto.....	7
Varnish Web Cache.....	7
Tinyproxy.....	8
Privoxy.....	9
Squid.....	11
1.4 Herramientas para la administración de servicios telemáticos.....	13
Zentyal.....	13
Webmin.....	14
HMAST.....	15
1.5 Lenguajes de programación.....	17
1.6 Metodología de desarrollo.....	19
1.7 Herramientas utilizadas.....	19
1.8 Librería para el trabajo con ficheros de configuración.....	20
Conclusiones parciales.....	21
Capítulo 2: Características del módulo a implementar.....	22
2.1 Propuesta del módulo a desarrollar.....	22
2.2 Lista de reserva del producto (LRP).....	22
2.3 Historias de Usuarios.....	28
2.4 Arquitectura del módulo.....	43
2.5 Diagrama de Paquetes.....	44
2.6 Patrones de diseño.....	47
Conclusiones parciales.....	48
Capítulo 3: Implementación y pruebas del módulo.....	49
3.1 Plan de iteraciones.....	49
3.2 Tareas de Ingeniería.....	49
3.3 Estándares de codificación.....	52
3.4 Pruebas de software.....	53
Conclusiones parciales.....	57
Conclusiones.....	59
Recomendaciones.....	60
Referencias bibliográficas.....	61
Bibliografía.....	63
Anexos.....	66
Glosario de términos.....	79

---

## Índice de tablas

Tabla 1: Comparación de servidores proxy en entornos de código abierto.....	13
Tabla 2: Lista de Reserva del Producto.....	28
Tabla 3: HU Instalar servicio Squid.....	29
Tabla 4: HU Administrar Servicio.....	30
Tabla 5: HU Configurar parámetros de red.....	31
Tabla 6: HU Configurar Lista de Control de Acceso(ACL).....	33
Tabla 7: HU Configurar ACL de tipo origen.....	34
Tabla 8: HU Configurar ACL de tipo destino.....	35
Tabla 9: HU Configurar ACL de tipo dominio.....	36
Tabla 10: HU Configurar ACL de tipo url.....	37
Tabla 11: HU Configurar conexión con Directorio Activo.....	38
Tabla 12: HU Configurar parámetros de la cache.....	40
Tabla 13: HU Configurar cuota de usuarios.....	41
Tabla 14: HU Configurar el formato de almacenamiento de los logs.....	42

## Índice de ilustraciones

Figura 1: Funcionamiento servidor proxy.....	7
Figura 2: Arquitectura del sistema.....	16
Figura 3: Arquitectura del módulo.....	44
Figura 4: Diagrama de paquetes.....	46
Figura 5: Enumeración del código.....	54
Figura 6: Grafo de flujo.....	55

## **Introducción**

El avance del desarrollo tecnológico ha contribuido a que las Tecnologías de Información y Comunicación (TIC) impulsen la vida cotidiana y profesional, las mismas están presentes en gran parte de las actividades humanas, aunque con mayor auge en el mundo de las empresas. Actualmente, es en el mundo de los negocios y las empresas donde mayor impacto están teniendo las TIC, asumiendo el liderazgo las grandes empresas transnacionales principalmente en países desarrollados, las cuales con su gran poder adquisitivo encabezan el mercado de producción y consumo; sin embargo, algunos países no tan desarrollados han potenciado la informatización de sus empresas logrando, aunque en menor medida, insertarse en el complejo mercado de los servicios y aplicaciones informáticas [1].

Cuba, en su estrategia de informatización de los Organismos de la Administración Central del Estado (OACE), ha tratado de apoyar tecnológicamente estas empresas buscando una mayor productividad y calidad en los productos y servicios que brindan. Uno de los pilares fundamentales en el cumplimiento de este objetivo son los centros de desarrollo, como la Empresa Nacional de Software (DESOFT S. A.) y los centros de altos estudios los cuales están vinculados a tareas de gran importancia en este proceso. Una de las instituciones con mayores resultados en este contexto es la Universidad de las Ciencias Informáticas (UCI), la cual ha tenido además excelentes resultados en la producción de software y prestación de servicios en países como la República Bolivariana de Venezuela.

La UCI centra su principal fuerza de producción en sus centros productivos, los cuales son un elemento importante en el proceso de desarrollo de la universidad. Estos cuentan con varias líneas de desarrollo enfocadas principalmente en el desarrollo informático del país, tales como las telecomunicaciones, biomedicina, realidad virtual, software libre, entre otros. Precisamente para el desarrollo del software libre es creado el Centro de Software Libre (CESOL), el cual está compuesto por los Departamentos Sistemas Operativos (SO) y Servicios Integrales en Migración, Asesoría y Soporte (SIMAYS). Este último especializado en brindar servicios relacionados con la migración a aplicaciones de código abierto como asesoría, consultoría, capacitación y soporte técnico; SIMAYS tiene como misiones fundamentales la migración a software libre y código abierto y el desarrollo de herramientas que apoyen este proceso. Una de las tareas de este departamento es la migración de los servicios telemáticos, los cuales son elemento de primer orden.

Los servicios telemáticos están presentes en la mayoría de las organizaciones y en muchos casos, son de importancia estratégica para las mismas. Actualmente la mayoría de los OACE brindan estos servicios

sobre sistemas operativos de la familia Microsoft Windows; sin embargo, la migración hacia plataformas GNU/Linux, es de suma importancia, debido a un amplio conjunto de características como lo son: la reducción de los costos de licencias, tanto por el sistema operativo instalado como por las aplicaciones que se ejecutan en el mismo, el poder alcanzar una alta disponibilidad y tiempo efectivo de funcionamiento, el aseguramiento de una mayor estabilidad y fiabilidad, entre otros.

Debido a las características que poseen las empresas cubanas de poca conectividad y hardware obsoleto con bajas prestaciones, el departamento SIMAYS inicia el desarrollo de la Herramienta para la Migración y Administración de Servicios Telemáticos (HMAST) desde las diferentes versiones de Windows Server hacia plataformas libres. La misma está en su primera versión y cuenta con los módulos para administrar el servicio DHCP y los que implementan el protocolo SSH; sin embargo, en estas empresas, una vez migradas, se hace necesario la administración del servicio proxy, el cual es indispensable para la navegación y descarga que realizan los usuarios de estas instituciones. Además, por las limitaciones de acceso a Internet, es necesario la gestión de cuotas de usuarios, regular el ancho de banda de acceso a algunas páginas, así como la configuración para la comunicación entre otros servidores proxy.

Para dar solución a la situación problemática descrita anteriormente se presenta como **problema científico**: ¿Cómo realizar la administración del servicio proxy desde HMAST?

El **objeto de estudio** de la investigación se centra en la administración del servicio proxy en entornos de código abierto, y como **campo de acción** la administración del servicio proxy en entornos de código abierto para HMAST. Por lo que se propone como **objetivo general** desarrollar un módulo para HMAST que permita la administración del servicio proxy en entornos de código abierto. Se pretende cumplir el objetivo general a partir de los siguientes **objetivos específicos**:

- Sistematizar en los software de código abierto que brindan el servicio proxy.
- Sistematizar en el estudio de herramientas para la administración del servicio proxy.
- Realizar el análisis y diseño de la aplicación a desarrollar.
- Implementar el software definido.
- Realizar pruebas al sistema desarrollado.

Para dar cumplimiento a los objetivos específicos se planifican las siguientes **tareas de investigación**:

- Revisión y análisis de bibliografía especializada del servicio proxy sobre plataformas GNU/Linux.
- Estudio de herramientas de código abierto que realizan la administración del servicio proxy.
- Análisis y diseño del módulo para HMAST para la administración del servicio proxy.
- Implementación del módulo definido.
- Realización de las pruebas correspondientes al módulo implementado.

En la presente investigación se plantea como **idea a defender** que el desarrollo de un módulo para HMAST facilitará el proceso de administración del servicio proxy en las instituciones cubanas.

Para el desarrollo de la investigación se tuvieron en cuenta los siguientes métodos de investigación científica:

Dentro de los métodos del nivel teórico, el **Analítico – Sintético** permitió realizar la investigación de las herramientas de administración del servicio proxy y un estudio del funcionamiento y las configuraciones necesarias de este servicio, facilitando la implementación del módulo propuesto. También se utilizó el **Análisis Histórico – Lógico** que posibilitó determinar los antecedentes históricos relacionados con las herramientas para la administración del servicio proxy y comprender lógicamente cuales son las tendencias actuales.

La investigación se ha estructurado para una mejor comprensión del contenido en 3 capítulos, conclusiones generales y bibliografía utilizada. Se cuenta además con los anexos que complementan el trabajo realizado. Los capítulos se estructuran de la siguiente forma:

**Capítulo 1: Servicios Proxy.** Se realiza un estudio acerca del servicio proxy, así como las herramientas que permiten su administración, sus principales características y funcionalidades; además, se abordan conceptos claves que serán usados durante el desarrollo de la investigación y se fundamentan las tecnologías, lenguajes de programación, metodología y herramientas utilizadas en el desarrollo del sistema.

**Capítulo 2: Características del módulo a implementar.** Se hace referencia a la solución que se propone, describiéndose el proceso ágil basado en las historias de usuario, correspondiente a las nuevas funcionalidades, prototipo de interfaz de usuario, plan de revisión, entre otros artefactos de interés.

**Capítulo 3: Implementación y pruebas del módulo.** Se implementa la solución propuesta, además, se explican las clases principales del sistema, se confecciona el plan de pruebas y se realizan las pruebas necesarias para verificar que las funcionalidades desarrolladas dan cumplimiento a los requisitos planteados.

## **Capítulo 1: Servicios Proxy**

En el presente capítulo se ofrece una visión general de los servicios proxy, abordando definiciones y conceptos fundamentales, así como herramientas que administran este servicio. Se describen además, las herramientas, tecnologías y metodología de desarrollo de software definidas en la implementación de HMAST. Para la realización de esta investigación es necesario conocer los conceptos relacionados con los servicios proxy.

### **1.1 Conceptos asociados al servicio proxy**

#### **Servidor Proxy HTTP**

Un servidor proxy es un sistema informático que está situado entre el cliente que solicita un documento web y el servidor de destino (otro sistema informático) que sirve el documento. En su forma más simple, un servidor proxy facilita la comunicación entre el cliente y el servidor de destino sin modificar solicitudes o respuestas [2].

#### **Proxy Cache**

Un servidor proxy-caché permite incrementar la velocidad de acceso a Internet al mantener localmente las páginas más consultadas por los usuarios de una organización, evitando las conexiones directas con los servidores remotos.

Los usuarios configuran su navegador web para dirigir sus accesos al servicio proxy-caché en vez de ir directamente al destino final. El servidor proxy-caché se encarga de proporcionarle la página pedida bien obteniéndola de su caché o accediendo al documento original; al dar servicio a muchos usuarios la caché contendrá muchos documentos beneficiándose toda la organización de ello. Se evitan transferencias innecesarias y con ello se aumenta la velocidad en la carga de las páginas, ya que no es necesario pedir una página cuando ya esté almacenada en la caché [3].

#### **Proxy Inverso**

Un proxy inverso es un servidor proxy-caché "al revés" que, en lugar de permitirles el acceso a Internet a usuarios internos, permite a usuarios de Internet acceder indirectamente a determinados servidores internos.

El servidor proxy inverso es utilizado como un intermediario por los usuarios de Internet que desean acceder a un sitio Web interno al enviar sus solicitudes indirectamente. Con un proxy inverso, el servidor Web está protegido de ataques externos directos, lo cual fortalece la red interna. Además, la función

caché de un proxy inverso puede disminuir la carga de trabajo del servidor asignado, razón por la cual se le denomina en ocasiones acelerador de servidor.

Finalmente, con algoritmos perfeccionados, el proxy inverso puede distribuir la carga de trabajo mediante la redirección de las solicitudes a otros servidores similares. Este proceso se denomina equilibrio de carga [4].

### **Proxy Transparente**

Un proxy transparente combina un servicio proxy con NAT<sup>1</sup> de manera que las conexiones son enrutadas dentro del proxy sin configuración por parte del cliente, y habitualmente sin que el propio cliente conozca de su existencia. Este es el tipo de proxy que utilizan los proveedores de servicios de Internet (ISP).

Un servidor proxy transparente es comúnmente instalado únicamente con la finalidad de compartir el Internet a varias computadoras pero normalmente es instalado por los propios usuarios o técnicos sin experiencia en redes, lo cual ocasiona una enorme vulnerabilidad de seguridad a las empresas al no establecer ninguna política de seguridad. Esto ocasiona que las computadoras queden expuestas al Internet y sean víctimas fáciles de cualquier atacante de la Internet [5].

### **1.2 Funcionamiento de un servidor proxy**

Básicamente funciona como caché de contenido de Red (principalmente HTTP), proporcionando en la proximidad de los clientes un caché de páginas y archivos disponibles a través de la red en servidores HTTP remotos, permitiendo a los clientes de la red local acceder hacia estos de forma más rápida.

Cuando se recibe una petición para un recurso de red especificado en una URL (*Uniform Resource Locator*), ver Figura 1, el servidor a través de las listas de control de acceso (ACL), verifica que el origen y el destino de la petición estén permitidos. Las ACL son reglas que el servidor proxy implementa y son las encargadas de permitir o denegar, respondiendo a las políticas que sean establecidas a través de ellas los contenidos solicitados. Una vez que se verifica que el recurso esté permitido, busca el resultado dentro de su caché. Si este es encontrado, responde al cliente proporcionado inmediatamente el contenido solicitado, de lo contrario, lo traerá desde el servidor remoto que contiene el recurso que el cliente solicitó, mostrándolo al mismo y guardando una copia en la caché. El contenido en la caché es posteriormente eliminado a través de un algoritmo de expiración de acuerdo con la antigüedad, tamaño e historial de respuestas a solicitudes (*hits*).

---

<sup>1</sup> **NAT:** (*Network Address Translation* - Traducción de Dirección de Red) es un mecanismo utilizado por routers IP para intercambiar paquetes entre dos redes que asignan mutuamente direcciones incompatibles.

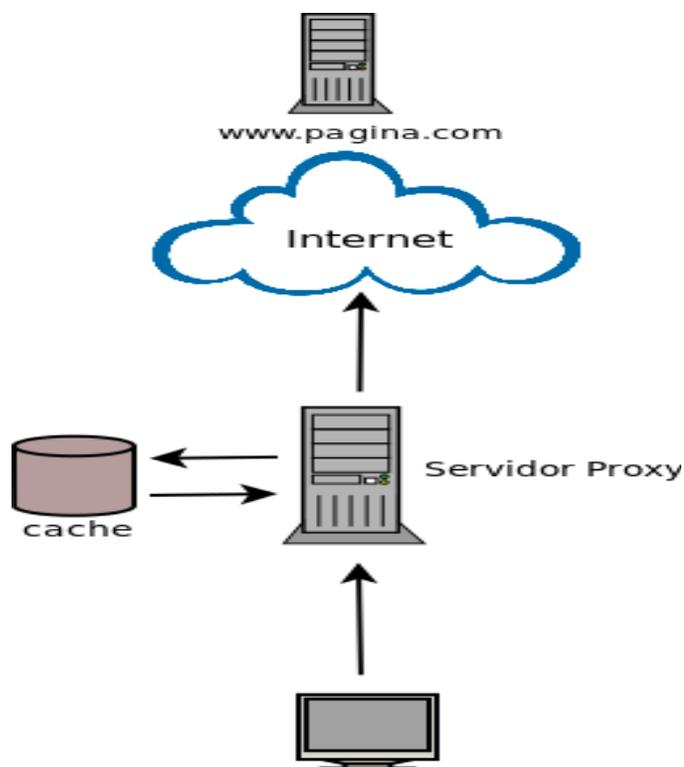


Figura 1: Funcionamiento servidor proxy

### 1.3 Servidores proxy de código abierto

En la actualidad existe una gran variedad de servicios dirigidos a funcionar de intermediarios en sistemas operativos GNU/Linux. A continuación se especifican las funcionalidades más importantes, así como ventajas y desventajas de cada uno de los servicios analizados, para definir cuál será utilizado en la administración del servicio proxy desde HMAST.

#### **Varnish Web Cache**

Varnish Web Cache es un acelerador Web, a veces referido como un acelerador de HTTP o un proxy HTTP inverso, el cual mejora significativamente el rendimiento de su Web.

Varnish acelera el cargado de una página Web mediante el almacenamiento de una copia de una página que un usuario visita por primera vez. La próxima vez que un usuario solicite la misma página, Varnish servirá a la copia en lugar de solicitar la página desde el servidor Web. Esto ayuda a que el servidor Web tenga que manejar menos tráfico y el rendimiento de su sitio Web y escalabilidad sea más óptimo. El

caché hecho por este servidor es a menudo la pieza más crítica de software en un negocio basado en la Web [6].

### Principales Características

Varnish almacena los datos en la memoria virtual y deja la tarea de decidir lo que se almacena en la memoria y lo que se guarda en el disco al sistema operativo. Esto ayuda a evitar la situación en la que el sistema operativo comienza a copiar datos al disco cuando la aplicación está haciendo lo mismo.

Una característica significativa de Varnish es que cada conexión cliente se asigna a un subproceso independiente. Cuando se alcanza el límite configurado en el número de subprocesos activos, las conexiones entrantes se colocan en cola y solo cuando esta cola alcanza su límite configurado serán rechazadas las conexiones entrantes.

Con el fin de reducir el número de llamadas al sistema a un mínimo, los datos de registro o logs se almacenan en la memoria compartida, y la tarea de escribir los registros en el disco se delega a una aplicación separada.

Se pueden destacar además, las siguientes características de Varnish Cache [7]:

- ✓ Diseño moderno.
- ✓ Utiliza *Visual Class Library* (VCL), el cual es un lenguaje de configuración que permite una gran flexibilidad y adaptación. Incluso es posible añadir código C con lo que es posible expandir Varnish sin modificar el código fuente.
- ✓ Equilibrio de carga.
- ✓ Web GUI (*Beta*), muy útil en caso de configuraciones en cluster.
- ✓ Posibilidad de cambiar configuraciones sobre la marcha.
- ✓ Reescritura y redireccionamiento de url (no es necesario un programa externo).
- ✓ Consulta de estadísticas y logs en tiempo real.

### Tinyproxy

Tinyproxy es un proxy ligero HTTP / HTTPS para sistemas operativos POSIX. Diseñado desde cero para ser rápido y pequeño, sin embargo, es una solución ideal para casos de uso, tales como

implementaciones integradas donde se requiere un proxy HTTP con todas las funciones, pero los recursos del sistema para una mayor representación no están disponibles. Tinyproxy es software libre y se distribuye con la licencia GNU GPL (versión 2 o superior) [8].

**Principales Características [8]:**

- ✓ Tinyproxy posee un tamaño pequeño y requiere de muy pocos recursos del sistema. La huella de memoria tiende a ser alrededor de dos MB con glibc<sup>1</sup>, y la carga del CPU aumenta linealmente con el número de conexiones simultáneas (dependiendo de la velocidad de la conexión). Por lo tanto, se puede ejecutar en una máquina de pocas prestaciones, o en un dispositivo de red, como un enrutador de banda ancha basado en Linux, sin ningún impacto notable en el rendimiento.
- ✓ Permite el reenvío de las conexiones HTTPS sin modificar el tráfico de cualquier manera a través del método CONNECT.
- ✓ Brinda funciones de proxy transparente, de modo que los usuarios puedan navegar sin necesidad de configuración del lado del cliente. También se puede usar como un proxy inverso para sitios Web.
- ✓ Posee características de privacidad las cuales permiten configurar qué encabezados HTTP se deben permitir, y cuáles deben ser bloqueados. Esto le permite restringir tanto los datos que vienen a su navegador Web desde el servidor HTTP, como los que van en sentido contrario.
- ✓ Facilita el monitoreo remoto, se puede acceder a las estadísticas del proxy desde cualquier lugar, lo que permite saber exactamente el rendimiento del servidor.

**Privoxy**

Privoxy es un proxy Web con capacidades avanzadas de filtrado para proteger la privacidad, filtrar contenido Web, gestionar cookies, controlar el acceso, y quitar anuncios, carteles, ventanas emergentes y otra basura de Internet. Privoxy tiene una configuración muy flexible y se puede personalizar a sus necesidades y gusto individual. Privoxy es adecuado tanto para sistemas aislados como para redes multiusuario [9].

**Principales Características:**

---

<sup>1</sup> Glibc es la biblioteca estándar de lenguaje C de GNU. Se distribuye bajo los términos de la licencia GNU LGPL.

Privoxy ofrece muchas características adicionales, algunas de ellas actualmente en fase de desarrollo, que dan el control del usuario final, más intimidad y más libertad:

- ✓ Soporta "Connection: keep-alive". Las conexiones salientes se pueden mantener con vida independiente del cliente.
- ✓ Compatible con IPv6, siempre que el sistema operativo lo haga también y el script de configuración que lo detecte.
- ✓ Soporta el etiquetado que permite cambiar el comportamiento basado en las cabeceras de cliente y servidor.
- ✓ Se puede ejecutar como un proxy "interceptor", lo que elimina la necesidad de configurar los navegadores de forma individual.
- ✓ Acciones y filtros sofisticados para manipular el servidor y el cliente.
- ✓ Puede ser encadenado con otros servidores proxy.
- ✓ Integrado basado en el navegador de configuración y utilidad de control en <http://config.privoxy.org/> (atajo: <http://url/>). Basado en el navegador seguimiento de reglas y efectos de filtro. Conmutación remota.
- ✓ Configuración modularizada que permite la configuración estándar y configuraciones de usuario a residir en archivos separados, por lo que la instalación de los archivos actualizados de las acciones no sobrescribirá la configuración individual del usuario.
- ✓ Soporte para PCRE (*Perl Compatible Regular Expressions*) en los archivos de configuración y una sintaxis más sofisticada y flexible.
- ✓ Filtrado de páginas Web (sustitución de texto, elimina banners en función del tamaño, invisibles "Web bugs" y molestias HTML, entre otros).
- ✓ Plantillas HTML personalizables por el usuario para la mayoría de proxy generadas páginas.
- ✓ Autodetección y relectura de cambios en los archivos de configuración.
- ✓ La mayoría de las funciones se pueden controlar en función de cada sitio o por la ubicación.

## **Squid**

Squid es un proxy caché para la Web con soporte HTTP, HTTPS, FTP entre otros. Reduce el ancho de banda y mejora los tiempos de respuesta en caché y reutilizando las páginas Web solicitadas con frecuencia. Squid tiene amplios controles de acceso y funciona como un gran servidor acelerador. Se ejecuta en los sistemas operativos más disponibles, incluyendo Windows y está disponible bajo la GNU GPL [10].

Entre otras cosas, Squid puede funcionar como servidor intermediario y caché de contenido de Red para los protocolos HTTP, FTP, GOPHER y WAIS, Proxy de SSL, caché transparente, WWCP, aceleración HTTP, caché de consultas DNS y otras como filtración de contenido y control de acceso por IP y por usuario.

Squid está compuesto de un programa principal como servidor, un programa para búsqueda en servidores DNS, programas opcionales para reescribir solicitudes y realizar autenticación y algunas herramientas para administración y herramientas para clientes. Al iniciar Squid da origen a un número configurable (5, de modo predeterminado a través de la opción `dns_children`) de procesos de búsqueda en servidores DNS, cada uno de los cuales realiza una búsqueda única en servidores DNS, reduciendo la cantidad de tiempo de espera para las búsquedas en servidores DNS.

Las principales funciones de Squid son las siguientes [11]:

- ✓ Permite el acceso Web a máquinas privadas (IP privada) que no están conectadas directamente a Internet.
- ✓ Controla el acceso Web aplicando reglas.
- ✓ Registra el tráfico Web desde la red local hacia el exterior.
- ✓ Controla el contenido Web visitado y descargado.
- ✓ Controla la seguridad de la red local ante posibles ataques e intrusiones en el sistema.
- ✓ Funciona como una caché de páginas Web. Es decir, almacena las páginas Web visitadas por los usuarios y de esta manera las puede enviar a otros usuarios sin tener que acceder a Internet de nuevo.
- ✓ Guarda en caché las peticiones DNS e implementa una caché para las conexiones fallidas.

- ✓ Registra logs de todas las peticiones cursadas.
- ✓ Soporta el protocolo ICP que permite integrar cachés que colaboran y permite crear jerarquías de cachés y el intercambio de datos.

Como consecuencia de estas funciones, la implantación de un servidor proxy-caché en una red proporciona las siguientes ventajas:

- ✓ Reduce los tiempos de respuesta: Si la página Web que se solicita está en la caché del servidor, esta se sirve sin necesidad de acceder de nuevo al servidor original, con lo cual se ahorra tiempo.
- ✓ Disminuye el tráfico en la red y el consumo de ancho de banda: Si la página Web está almacenada en la caché del servidor, la petición no sale de la red local y no será necesario hacer uso de la línea exterior consiguiendo así un ahorro en la utilización del ancho de banda.
- ✓ Cortafuegos: Cuando se utiliza un servidor proxy-caché, este comunica con el exterior, y puede funcionar como cortafuegos, lo cual aumentará la seguridad del usuario respecto a la información a la que se acceda.
- ✓ Filtrado de servicios: Es posible configurar el servidor proxy-caché dejando solo disponibles aquellos servicios (HTTP, FTP) que se consideren necesarios, impidiendo la utilización del resto.

### Selección del Servidor

A continuación se muestra una tabla comparativa donde quedan reflejados aspectos relevantes que caracterizan a cada uno de los servidores proxy analizados anteriormente, así como funcionalidades necesarias en las instituciones cubanas, con el fin de comparar y arribar a conclusiones acertadas que apoyen la selección del que será utilizado para dar solución al problema planteado.

Funcionalidades	Squid	Varnish Cache	Privoxy	Tiny Proxy
Gestión de Listas de Control de Acceso (ACL).	Alta	Bajo	Alto	Medio
Configuración para la comunicación con otros proxy-cache.	Si	No	Si	No
Gestión de Cuotas de usuarios.	Si	No	No	No
Gestión del ancho de banda de la red.	Si	No	No	No

Funcionalidades	Squid	Varnish Cache	Privoxy	Tiny Proxy
Almacenamiento de las peticiones DNS.	Si	No	No	No

Tabla 1: Comparación de servidores proxy en entornos de código abierto.

Una vez analizada la tabla comparativa de los servidores proxy, se selecciona Squid como software para administrar el servicio proxy desde HMAST, por resultar ser el más completo de los que fueron analizados. Este cuenta con un amplio número de ventajas asociadas con las funcionalidades y protocolos que soporta. Entre sus características distintivas se destacan que permite el acceso Web a máquinas privadas que no están conectadas directamente a Internet, guarda en caché las peticiones DNS e implementa una caché para las conexiones fallidas y soporta el protocolo ICP, el cual permite integrar cachés que colaboran entre sí, así como crear jerarquías de caché e intercambio de datos. Actualmente es el más usado en entornos libres y herramientas reconocidas para la administración de servicios telemáticos como Zentyal y Webmin, utilizan Squid para administrar el servicio proxy.

#### 1.4 Herramientas para la administración de servicios telemáticos

Las herramientas de administración de servidores facilitan el trabajo de los administradores de red permitiendo realizar cambios desde una interfaz sin necesidad de acceder directamente al fichero de configuración. En la actualidad se han desarrollado una gran variedad de estas herramientas en entornos libres, de las cuales se realizará un estudio describiéndose sus principales funcionalidades, enfocadas en la administración del servicio proxy.

##### Zentyal

Zentyal (anteriormente conocido como *eBox Platform*) es un servidor de red unificada de código abierto (o una plataforma de red unificada) para las PYMEs<sup>I</sup>. Zentyal puede actuar gestionando la infraestructura de red, como puerta de enlace a Internet (Gateway), gestionando las amenazas de seguridad (UTM), como servidor de oficina, como servidor de comunicaciones unificadas o una combinación de estas. Además, Zentyal incluye un marco de desarrollo (un framework) para facilitar el desarrollo de nuevos servicios basados en Unix [12].

Para la administración del servicio proxy este servidor utiliza Squid permitiendo las siguientes configuraciones:

I PYMEs: Acrónimo de Pequeñas y Medianas Empresas.

- ✓ Definir si el proxy funcionará en modo transparente.
- ✓ Filtrado de páginas Web según su contenido.
- ✓ Implementar un límite flexible para controlar el ancho de banda que consumen los usuarios.
- ✓ Configuración de los parámetros básicos de administración del servicio proxy como puerto escucha, cache, ACL, validación de usuario.

El módulo de Zentyal para la administración del servicio proxy presenta, como todos los demás módulos, una interfaz con un alto grado de usabilidad. Sin embargo, este servidor no reconoce y sobrescribe las configuraciones establecidas en el fichero principal de configuración de Squid, de manera que no permitirá realizar configuraciones que no estén establecidas en el módulo como las comunicaciones entre proxy.

### **Webmin**

Webmin es una interfaz Web escrita en Perl para administrar un servidor. Con esta herramienta se pueden configurar los permisos para Usuarios y Grupos o configurar el funcionamiento del servidor Mysql Server, cuotas de espacio, servicios, archivos de configuración, apagado del equipo, entre otras opciones.

Webmin tiene una estructura por módulos para administrar una amplia variedad de herramientas como Apache, PHP, MySQL, DNS, Samba, DHCP, Proxy, entre otros, además de ser completamente configurables y de poder crear nuevos módulos [13].

Este servidor administra el servicio proxy a través de Squid, permitiendo las siguientes configuraciones:

- ✓ Establecer las configuraciones básicas y avanzadas para el correcto funcionamiento del servicio como puerto de escucha, parámetros de la cache, reglas de acceso.
- ✓ Sincronización del servicio con el cortafuegos del sistema.
- ✓ Administrar opciones de registros del sistema.
- ✓ Reconoce los cambios realizados en el fichero principal de configuración.

El módulo de administración del servicio proxy de Webmin cuenta con múltiples opciones de configuración, sin embargo, está diseñado con una interfaz poco intuitiva por los numerosos campos de configuración que presenta en una misma página. Otro elemento a destacar es que no permite la

configuración del sistema para que funcione como proxy transparente ni proxy inverso, además de no permitir la configuración de la comunicación entre proxy.

## **HMAST**

HMAST es el sistema que permite administrar los servidores de forma remota, el cual tiene las funcionalidades necesarias para la gestión de usuarios, tareas programadas y servicios. Permite la administración de conexiones a través del protocolo SSH y así como la del servicio DHCP.

### **Arquitectura del sistema**

La arquitectura que presenta HMAST propone el diseño de una arquitectura N-Capas basada en el dominio compuesta por cinco capas, como se muestra en la Figura 2 las cuales son descritas a continuación.

#### **Capa Aplicación (*application*)**

Es una capa delgada, comúnmente solo realizará llamadas a servicios de capas inferiores (dominio), los servicios que publica (servicios de aplicación) tienen la responsabilidad, por ejemplo de adaptar la información que le llega, a los requerimientos de los servicios de dominio

#### **Capa Dominio (*domain*)**

Constituye el hilo conductor de la aplicación, sus componentes solo dependen de la Capa Infraestructura transversal (*infrastructureCrosscutting*) (están totalmente desacoplados). Implementa la lógica de dominio (reglas de negocio), es responsable de las validaciones. Define las interfaces de persistencia a datos (contratos de repositorio), pero no los implementa. Sus componentes no están ligados a tecnologías específicas.

#### **Capa Persistencia (*persistens*)**

La capa es responsable de contener el código necesario para persistir los datos. Los principales componentes que contendrá la capa son los repositorios: son clases que implementan los contratos de repositorios definidos en la capa de dominio.

#### **Capa Infraestructura transversal (*infrastructureCrosscutting*)**

Las responsabilidades de la presente capa están dadas a promover la reutilización de código, ella albergará las operaciones de seguridad, logging, monitoreo del sistema, mecanismos de persistencia reutilizables, validadores genéricos, en fin todas aquellas operaciones que se puedan llamar desde otras capas.



#### Funcionalidades que brinda

- ✓ Gestión de servidores lógicos: Permite la adición, edición, eliminación de los datos de un servidor lógico, además permite la conexión remota y desconexión a un servidor seleccionado.
- ✓ Gestión de servicios telemáticos asociados a un servidor lógico: Permite la adición, edición, eliminación de los datos de un módulo, así como activación y desactivación de los mismos.
- ✓ Gestión de los variables de configuración asociadas a un servidor lógico: Permite cargar y salvar las variables de configuración de los servicios telemáticos encontrado en un servidor lógico (ficheros de configuración, nombre de módulos, demonios, entre otros).

#### Consideraciones para implementar un módulo para HMAST.

- ✓ La lógica de Aplicación no deberá incluir ninguna lógica del Dominio, solo tareas de coordinación relativa a requerimientos técnicos de la aplicación, como conversiones de formatos de datos de entrada a entidades del Dominio, llamadas a componentes de Infraestructura para que realicen tareas complementarias.

- ✓ Se debe garantizar que no viajen hacia y desde la capa de presentación objetos de dominio, en su lugar deben viajar Objetos de Transferencia de Datos (DTO).
- ✓ Las entidades solo pueden tener dependencias de componentes de la capa de dominio.
- ✓ Las clases de servicios deben ser las únicas responsables (vías de acceso) de acceder a los repositorios, no se puede implementar código de persistencia a datos en la capa de dominio.
- ✓ Solo se puede acceder a la información almacenada en los servidores haciendo uso de los repositorios.
- ✓ Es importante que todo el código reutilizable por más de un repositorio se ponga a disposición de todos en la capa de infraestructura transversal.

### 1.5 Lenguajes de programación

Un lenguaje de programación es aquel elemento dentro de la informática que permite crear programas mediante un conjunto de instrucciones, operadores y reglas de sintaxis; que pone a disposición del programador para que este pueda comunicarse con los dispositivos hardware y software existentes [14].

En la concepción inicial del sistema base del módulo a desarrollar, quedó definido el uso del lenguaje de programación Java, apoyado en el framework de desarrollo Spring de esta misma plataforma. Se utiliza además la tecnología XML para una mayor estructuración de las configuraciones y la persistencia de datos en el software.

#### Java

Es un lenguaje de programación orientado a objetos desarrollado por *Sun Microsystems* en 1991. Dado que es un lenguaje orientado a objetos, implica que su concepción es muy próxima a la forma de pensar humana. También posee otras características muy importantes [15]:

- **Compilado:** genera ficheros de clases compiladas, pero estas clases compiladas, son en realidad interpretadas por la máquina virtual de java, siendo la máquina virtual de java la que mantiene el control sobre las clases que se estén ejecutando.
- **Multiplataforma:** El mismo código java que funciona en un sistema operativo, funcionará en cualquier otro sistema operativo que tenga instalada la máquina virtual java.

- **Seguro:** La máquina virtual, al ejecutar el código java, realiza comprobaciones de seguridad, además el propio lenguaje carece de características inseguras, como por ejemplo los punteros.

### **Spring Framework**

Spring es un framework para el desarrollo de aplicaciones y contenedor de inversión de control, de código abierto para la plataforma Java.

Spring Framework ofrece una programación completa y el modelo de configuración para las modernas aplicaciones empresariales basadas en Java en cualquier tipo de plataforma de despliegue. Un elemento clave de Spring es el apoyo de infraestructura a nivel de aplicación: este framework se centra en la "fontanería" de las aplicaciones empresariales a fin de que los equipos pueden centrarse en la lógica de negocio a nivel de aplicación, sin ataduras innecesarias a los entornos de despliegue específico.

### **XML**

XML proviene de *eXtensible Markup Language* (lenguaje marcado extensible). Se trata de un metalenguaje (un lenguaje que se utiliza para decir algo sobre otro lenguaje) extensible de etiquetas que fue desarrollado por el *World Wide Web Consortium* (W3C), un consorcio internacional que elabora recomendaciones para la World Wide Web<sup>1</sup>.

### **Lenguaje de Modelado.**

Es un conjunto estandarizado de símbolos y de modos de disponerlos para modelar parte de un diseño de software orientado a objetos. Generalmente se usa extensivamente en combinación con una metodología de desarrollo de software para avanzar de una especificación inicial a un plan de implementación y para comunicar dicho plan a todo un equipo de desarrolladores. El uso de un lenguaje de modelado es más sencillo que la auténtica programación, pues existen menos medios para verificar efectivamente el funcionamiento adecuado del modelo.

### **Lenguaje Unificado de Modelado (UML).**

Es un lenguaje gráfico para especificar, construir, visualizar, documentar las partes o artefactos que son información utilizada y originada mediante un proceso de software. Es un lenguaje estándar de modelado orientado a objetos [17].

---

<sup>1</sup> **World Wide Web:** (Mayormente conocido como WWW) es un sistema de distribución de información basado en hipertexto o hipermedios enlazados y accesibles a través de Internet.

UML no puede compararse con la programación estructurada, no es programación, solo se diagrama la realidad de una utilización en un requerimiento. Mientras que, programación estructurada, es una forma de programar como lo es la orientación a objetos, sin embargo, la orientación a objetos viene siendo un complemento perfecto de UML, pero no por eso se toma UML solo para lenguajes orientados a objetos.

### **1.6 Metodología de desarrollo**

Una metodología de desarrollo de software es un conjunto de procedimientos, técnicas, herramientas y un soporte documental que ayuda a los desarrolladores a realizar determinado software [18].

Principalmente se encarga de elaborar estrategias de desarrollo de software que promuevan prácticas adaptativas en vez de predictivas; centradas en las personas o los equipos, orientadas hacia la funcionalidad y la entrega, de comunicación intensiva y que requieren implicación directa del cliente.

#### **SXP**

SXP es una metodología de desarrollo de software compuesta por las metodologías SCRUM y XP que ofrece una estrategia tecnológica, a partir de la introducción de procedimientos ágiles que permitan actualizar los procesos de software para el mejoramiento de la actividad productiva fomentando el desarrollo de la creatividad, aumentando el nivel de preocupación y responsabilidad de los miembros del equipo, ayudando al líder del proyecto a tener un mejor control del mismo.

SXP está especialmente indicada para proyectos de pequeños equipos de trabajo, rápido cambio de requisitos o requisitos imprecisos, muy cambiantes, donde existe un alto riesgo técnico y se orienta a una entrega rápida de resultados y una alta flexibilidad. Ayuda a que trabajen todos juntos, en la misma dirección, con un objetivo claro, permitiendo además seguir de forma clara el avance de las tareas a realizar, de forma que los jefes pueden ver día a día cómo progresa el trabajo [19].

La utilización de la metodología de desarrollo de software SXP permitirá agilizar el análisis, diseño e implementación de la solución propuesta.

### **1.7 Herramientas utilizadas**

Una herramienta es una aplicación empleada para la construcción de otros programas o aplicaciones. Diseñadas y fabricadas para cumplir uno o más propósitos, teniendo una función técnica. A continuación se describen las herramientas a utilizar en el desarrollo de la presente investigación.

### **Netbeans IDE**

Netbeans IDE es un entorno de desarrollo; una herramienta para que los programadores puedan escribir, compilar, depurar y ejecutar programas. Está escrito en Java, pero puede servir para cualquier otro lenguaje de programación. Existe además un número importante de módulos para extender el NetBeans IDE. Es un producto libre y gratuito sin restricciones de uso [20].

### **Visual Paradigm**

Es una herramienta UML profesional que soporta el ciclo de vida completo del desarrollo de software. Es una herramienta *Computer Aided Software Engineering* (CASE) potente y fácil de utilizar, permite el modelado visual UML, propiciando la rápida construcción de aplicaciones de calidad. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación. Es una herramienta colaborativa, pues soporta múltiples usuarios trabajando sobre el mismo proyecto [21].

### **RapidSVN**

RapidSVN es un cliente de interfaz gráfica para la comunicación con servidores Subversion. Está escrito en C++ y distribuido bajo licencia GPL. Facilita el versionado de ficheros, desde una interfaz sencilla e intuitiva y se encuentra disponible para plataformas Windows, Linux, MAC OS X y Solaris. Es una herramienta rápida y eficiente [22].

## **1.8 Librería para el trabajo con ficheros de configuración**

### **Augeas**

Augeas es un editor que analiza los archivos en sus formatos nativos y los transforma en un tipo abstracto de datos llamado árbol, y viceversa. La manipulación de la configuración la realiza mediante este árbol, y los cambios que sean aplicados a él se guardan de forma nativa en dichos archivos de configuración. Para su realización utiliza unas estructuras llamadas Lenses (en español lentes), que permiten que Augeas pueda entender cualquier archivo de configuración.

Augeas cuenta con una API de no más de una docena de funcionalidades, que le brindan a los desarrolladores la posibilidad de interactuar con la herramienta. También brinda implementaciones en Python, Ocaml, Ruby, Perl, Haskell, Java, entre otros. Permite además la interacción mediante la consola de Linux, a través de la herramienta augtool [23].

### **Conclusiones parciales**

En el desarrollo del presente capítulo fueron analizados los principales servidores proxy existentes en entornos de código abierto, seleccionando el servidor Squid como más indicado para la administración del servicio proxy en las instituciones cubanas desde HMAST. Este software ofrece funcionalidades como la gestión de cuotas de navegación de usuarios, configuración del servidor como proxy padre o hijo, manejo del ancho de banda de la red además de un potente control de acceso a través de las ACL, elementos necesarios en la mayoría de las empresas nacionales que poseen conectividad. Además, el estudio realizado a las herramientas Zentyal y Webmin, las cuales ofrecen el servicio proxy a través del software seleccionado, permitió obtener aspectos positivos que contribuyeron en la concepción del módulo a desarrollar. Entre los más relevantes se encuentran la combinación de ACL en diferentes horarios de la semana, así como la posibilidad de establecer un ancho de banda a la regla creada.

## Capítulo 2: Características del módulo a implementar

En el presente capítulo se definirá el sistema propuesto donde se reflejarán los requisitos funcionales y no funcionales mediante la Lista de Reserva de Producto (LRP). Se recogen las funcionalidades de la solución propuesta reflejadas en las Historias de Usuarios con sus prototipos de interfaces de usuario no funcionales, además se define la arquitectura y los patrones de diseño que se utilizarán y la estructura que debe cumplir el módulo.

### 2.1 Propuesta del módulo a desarrollar.

Con el objetivo de darle solución al problema planteado, el presente trabajo de diploma propone desarrollar un módulo para HMAST con las funcionalidades necesarias para administrar el servicio proxy.

El módulo a desarrollar permitirá la administración del servicio proxy, estableciendo las principales configuraciones generales y avanzadas. Brinda la posibilidad de realizar configuraciones para la autenticación de los usuarios, permitiendo seleccionar el origen de los datos de usuarios a validar, como son Directorio Activo o ficheros locales. Permite además la creación de ACL asignándole un horario y un ancho de banda específico, así como la combinación de reglas, permitiendo una mayor flexibilidad y efectividad en la aplicación de las mismas. Otro elemento a destacar es la posibilidad de configurar el servidor como proxy padre, además de poder establecer conexiones con otros proxy hermanos o padres.

### 2.2 Lista de reserva del producto (LRP)

La lista de reserva del producto es una colección organizada y priorizada de los requisitos sobre el producto. El objetivo de la misma radica en cubrir las cualidades requeridas en el software y determinar el orden en que se le irá dando cumplimiento durante las iteraciones o sprint<sup>1</sup> a cada requerimiento recogido según la prioridad establecida en cada uno; también abarca las características que el producto debe tener.

---

<sup>1</sup> **Sprint:** Ciclo iterativo o período de tiempo en el que se implementa o mejora una funcionalidad del sistema. Durante un sprint el producto puede ser diseñado, implementado y probado para producir nuevos incrementos.

Prioridad	Ítem *	Descripción	Estimación	Estimado por
<b>Alta</b>				
	1	Instalar servicio	0.1	Alexander Pérez
	2	Desinstalar servicio	0.1	
	3	Iniciar servicio	0.1	
	4	Detener servicio	0.1	
	5	Reiniciar servicio	0.1	
	6	Agregar puerto de escucha	0.1	
	7	Modificar puerto de escucha	0.1	
	8	Eliminar puerto de escucha	0.1	
	9	Mostrar puerto de escucha	0.1	
	10	Establecer dirección de escucha	0.1	
	11	Modificar dirección de escucha	0.1	
	12	Eliminar dirección de escucha	0.1	
	13	Eliminar dirección de escucha	0.1	
	14	Establecer el visible hostname	0.1	
	15	Modificar el visible hostname	0.1	
	16	Establecer servidor dns en squid	0.1	
	17	Modificar servidor dns en squid	0.1	
	18	Mostrar servidor dns en squid	0.1	
	19	Establecer parámetros de autenticación con Directorio Activo.	0.1	
	20	Modificar parámetros de autenticación con Directorio Activo.	0.1	
	21	Eliminar parámetros de autenticación con Directorio Activo.	0.1	
	22	Mostrar parámetros de autenticación con Directorio Activo.	0.1	

	23	Establecer parámetros de autenticación con fichero local.	0.1	Alexander Pérez
	24	Modificar parámetros de autenticación con fichero local.	0.1	
	25	Eliminar parámetros de autenticación con fichero local.	0.1	
	26	Mostrar parámetros de autenticación con fichero local.	0.1	
	27	Modificar tamaño de la cache(RAM).	0.1	
	28	Mostrar tamaño de la cache(RAM).	0.1	
	29	Establecer parámetros de almacenamiento de la cache.	0.1	
	30	Modificar parámetros de almacenamiento de la cache.	0.1	
	31	Eliminar parámetros de almacenamiento de la cache.	0.1	
	32	Mostrar parámetros de almacenamiento de la cache.	0.1	
	33	Crear ACL para restringir acceso por Ip origen.	0.1	
	34	Crear ACL para restringir acceso por Ip destino.	0.1	
	35	Crear ACL para restringir acceso por puertos.	0.1	
	36	Crear ACL para restringir URLs.	0.1	
	37	Crear ACL para restringir acceso por dirección mac.	0.1	
	38	Crear ACL para restringir acceso por tipo mime.	0.1	

39	Crear ACL para restringir acceso por protocolos.	0.1
40	Crear ACL para restringir URLs por patrón.	0.1
41	Crear ACL para restringir la navegación por dominios.	0.1
42	Crear ACL para restringir el acceso al squid por subredes.	0.1
43	Crear ACL para restringir el acceso por días y horarios	0.1
44	Crear ACL para restringir el tamaño de ficheros al descargarse.	0.1
45	Crear ACL para restringir cantidad de usuarios por IP.	0.1
46	Crear ACL para restringir el acceso por horarios.	0.1
47	Mostrar ACL.	0.1
48	Modificar ACL.	0.1
49	Eliminar ACL.	0.1
50	Establecer política de acceso a las ACLs.	0.1
51	Modificar política de acceso a las ACLs.	0.1
52	Eliminar política de acceso a las ACLs.	0.1
53	Mostrar política de acceso aplicadas a las ACLs.	0.1
54	Establecer ancho de banda.	0.1
55	Modificar ancho de banda.	0.1
56	Eliminar ancho de banda.	0.1
57	Mostrar ancho de banda.	

<b>Media</b>				
	58	Establecer cuota de los usuarios.	0.1	Alexander Pérez
	59	Modificar cuota de usuarios.	0.1	
	60	Eliminar cuota de usuarios.	0.1	
	61	Mostrar cuota de usuarios.	0.1	
	62	Establecer puerto ICP para servir de proxy padre o hermano.	0.1	
	63	Modificar puerto ICP para servir de proxy padre o hermano.	0.1	
	64	Mostrar puerto ICP para servir de proxy padre o hermano.	0.1	
	65	Establecer dirección de servidores proxy hijos permitidos para realizar consultas ICP.	0.1	
	66	Modificar dirección de servidores proxy hijos permitidos para realizar consultas ICP.	0.1	
	67	Eliminar dirección de servidores proxy hijos permitidos para realizar consultas ICP.	0.1	
	68	Mostrar dirección de servidores proxy hijos permitidos para realizar consultas ICP.	0.1	
	69	Establecer comunicación con proxy padre o hermano.	0.1	
	70	Mostrar comunicación con proxy padre o hermano.	0.1	
	71	Eliminar comunicación con proxy padre o hermano.	0.1	

72	Mostrar comunicación con proxy padre o hermano.	0.1	
73	Establecer tamaño de objetos a almacenar.	0.1	
74	Modificar tamaño de objetos a almacenar.	0.1	
75	Mostrar tamaño de objetos a almacenar.	0.1	
76	Crear formato de los logs.	0.1	
77	Modificar formato de los logs.	0.1	
78	Mostrar formato de los logs.	0.1	
79	Establecer formato de logs a aplicar.	0.1	
80	Modificar formato de logs a aplicar.	0.1	
81	Eliminar formato de logs a aplicar.	0.1	
82	Mostrar formato de logs a aplicar.	0.1	
<b>RNF (Requisitos No Funcionales)</b>			
1	Servidor que tenga al menos 2 tarjetas de red, 512 MB de RAM y 100 GB de almacenamiento en disco duro.		Alexander Pérez
2	Usar como lenguaje de programación java con el framework de desarrollo Spring.		
3	Se utilizará la librería augeas-java, para el manejo de la información presente en los ficheros de configuración.		
4	Utilizar como entorno de desarrollo integrado Netbeans en su versión 7.2.1.		
5	Utilizar como herramienta de modelado la Visual Paradigm en su versión 8.0.		

	6	La aplicación se ejecutará en sistemas operativos libres.		
--	---	---	--	--

Tabla 2: Lista de Reserva del Producto.

### 2.3 Historias de Usuarios

Las historias de usuario son la técnica utilizada en SXP para especificar los requisitos del software, lo que equivaldría a los casos de uso en el proceso unificado. Las mismas son escritas por los clientes como las tareas que el sistema debe hacer y su construcción depende principalmente de la habilidad que tenga el cliente para definir las. Son escritas en lenguaje natural, sin un formato predeterminado, no excediendo su tamaño de unas pocas líneas de texto, guían la construcción de las pruebas de aceptación y son utilizadas para estimar tiempos de desarrollo.

#### Instalar servicio Squid

Historia de Usuario	
<b>Número:</b> HMAST_Proxy_1	<b>Nombre Historia de Usuario:</b> Instalar servicio Squid.
<b>Modificación de Historia de Usuario Número:</b> Ninguna.	
<b>Usuario:</b> Alexander Pérez Tasé	<b>Iteración Asignada:</b> 1.
<b>Prioridad en Negocio:</b> Muy Alta.	<b>Puntos Estimados:</b> 0,1 semana.
<b>Riesgo en Desarrollo:</b> Alta.	<b>Puntos Reales:</b> 0,1 semana.
<b>Descripción:</b> El módulo permitirá la instalación del servicio en el servidor.	
Observaciones: Verificará la existencia del servicio proxy en el servidor, de no existir brindará la opción de instalar el servicio.	
<p>✓<b>Instalar servicio:</b> Para la instalación del servicio se utilizará el comando apt-get install squid3. Una vez instalado el servicio, se le ofrecerá al usuario dos posibles plantillas para establecer configuraciones básicas. El primer caso será una o varias subredes locales detrás de un servidor proxy conectado directamente a Internet. Una vez seleccionada esta plantilla se podrá modificar el puerto de escucha establecido por defecto del squid (3128), el cual se mostrará en un campo de texto, en caso de querer agregar otro puerto de escucha se podrá realizar a través de un botón ubicado al lado del campo de texto. También se podrá establecer la dirección de la o las tarjetas de</p>	

red locales; una vez introducidas estas direcciones, automáticamente se le permitirá a las subredes de cada una de las direcciones el acceso al proxy.

El segundo caso estará caracterizado por la configuración de uno de los dos servidores proxy a administrar, uno ubicado entre una (o varias) subred local y otro servidor proxy de mayor jerarquía de red. Para la configuración de este servidor proxy se realizarán configuraciones similares a la primera plantilla. En el caso de la configuración del proxy de mayor jerarquía, se podrá establecer el puerto y las direcciones por las cuales escuchará las peticiones. También se podrá establecer como proxy padre a través de un checkbox, el cual habilitará un campo de texto para establecer el puerto por el cual el servidor escuchará las peticiones ICP, este puerto será 0 o un número mayor que 1025 y menor que 65635, además de ser diferente de los puertos de escucha establecidos anteriormente.

**Prototipo de interfaz: Anexo 1: Interfaz de instalación 1 Anexo 2: Interfaz de instalación 2 Anexo 3: Interfaz de instalación 3 Anexo 4: Interfaz de instalación 4**

Tabla 3: HU Instalar servicio Squid.

### Administrar Servicio

Historia de Usuario	
<b>Número:</b> HMAST_Proxy_2	<b>Nombre Historia de Usuario:</b> Administrar Servicio.
<b>Modificación de Historia de Usuario Número:</b> Ninguna.	
<b>Usuario:</b> Alexander Pérez Tasé	<b>Iteración Asignada:</b> 1.
<b>Prioridad en Negocio:</b> Muy Alta.	<b>Puntos Estimados:</b> 1 semana.
<b>Riesgo en Desarrollo:</b> Alta.	<b>Puntos Reales:</b> 1 semanas.
<b>Descripción:</b> El módulo permitirá iniciar, detener y reiniciar el sistema.	
<b>Observaciones:</b> Para administrar el servicio se selecciona la opción Administrar Servicio del menú principal y se mostrarán tres botones con las acciones a realizar.	
<b>✓Iniciar Servicio:</b> Para iniciar el servicio, se debe dar clic en la opción iniciar servicio ubicado en la parte superior derecha del sistema. Esta opción ejecutará el comando service squid3 start.	

<p>✓<b>Reiniciar Servicio:</b> Para reiniciar el servicio, se debe seleccionar la opción reiniciar servicio ubicado en la parte superior derecha del sistema. Esta opción ejecutará el comando service squid3 restart.</p> <p>✓<b>Detener Servicio:</b> Para detener el servicio, se selecciona la opción detener servicio ubicado en la parte superior derecha del sistema. Esta opción ejecutará el comando service squid3 stop.</p>
<p><b>Prototipo de interfaz: Anexo 5: Administración del servicio</b></p>

Tabla 4: HU Administrar Servicio

**Configurar parámetros de red**

Historia de Usuario	
<p><b>Número:</b> HMAST_Proxy_3</p>	<p><b>Nombre Historia de Usuario:</b> Configurar parámetros de red.</p>
<p><b>Modificación de Historia de Usuario Número:</b> Ninguna.</p>	
<p><b>Usuario:</b> Alexander Pérez Tasé</p>	<p><b>Iteración Asignada:</b> 1</p>
<p><b>Prioridad en Negocio:</b> Muy Alta.</p>	<p><b>Puntos Estimados:</b> 1 semana.</p>
<p><b>Riesgo en Desarrollo:</b> Alto.</p>	<p><b>Puntos Reales:</b> 1 semana.</p>
<p><b>Descripción:</b> El módulo mostrará los parámetros de configuración de las opciones de red del servidor. También permitirá adicionar, modificar y eliminar los mismos.</p>	
<p><b>Observaciones:</b> El puerto por el cual el servidor escuchará las peticiones estará definido por la directiva http_port, la cual tiene como valor predeterminado el 3128. Las direcciones de los servidores DNS está definida por la directiva dns_nameservers. Además, se podrá establecer el nombre con el cual el servidor proxy se identificará en la red el cual se puede definir con la directiva visible_hostname.</p> <p>Las configuraciones que se realizan en la interfaz de parámetros de red no serán guardados hasta que se sea seleccionada la opción de Aplicar Cambios.</p> <p>✓<b>Establecer puerto escucha:</b> El usuario establece el o los puertos por el cual el servidor Squid escuchará las peticiones de conexión. El puerto debe ser un número mayor que 1024 y menor que 65635. Por defecto está establecido el puerto 3128.</p>	

✓**Eliminar puerto escucha:** El usuario podrá eliminar los puertos escucha si hay más de dos puertos, en caso de solo haber uno esta opción será deshabilitada.

✓**Modificar puerto escucha:** El usuario podrá modificar el o los puertos por el cual el servidor squid escuchará las peticiones de conexión. El puerto debe ser un número mayor que 1024 y menor que 65635.

✓**Establecer dirección escucha:** La aplicación permitirá que se establezcan las direcciones por el cual el servidor squid escuchará las peticiones de conexión. La dirección escucha debe ser un IPv4 (Ejemplo: 192.16.1.5).

✓**Modificar dirección escucha:** La aplicación permitirá que se modifique la dirección por el cual el servidor squid escuchará las peticiones de conexión. La dirección escucha debe ser un IPv4 (Ejemplo: 192.16.1.200).

✓**Eliminar dirección escucha:** La aplicación permitirá que se elimine la dirección por el cual el servidor squid escuchará las peticiones de conexión.

✓**Establecer servidores dns:** El módulo debe permitir que se establezcan los servidores DNS a los cuales el servidor hará peticiones de resolución de nombres de dominios. Las direcciones de los servidores dns deben ser IPv4 (Ej: 255.255.255).

✓**Modificar servidores dns:** El módulo debe permitir que se modifique los servidores DNS a los cuales el servidor hará peticiones de resolución de nombres de dominios. Las direcciones de los servidores dns deben ser IPv4 (Ej: 255.255.255).

✓**Eliminar servidores dns:** El módulo debe permitir que se eliminen los servidores DNS a los cuales el servidor hará peticiones de resolución de nombres de dominios.

✓**Establecer el nombre del proxy:** El módulo debe permitir que se establezca el nombre con el que squid se anunciará, y también será el nombre que aparecerá en las páginas de error.

Modificar el nombre del proxy: El módulo debe permitir que se modifique el nombre con el que squid se anunciará, y también será el nombre que aparecerá en las páginas de error.

**Prototipo de Interfaz: Anexo 6: Parámetros de red**

Tabla 5: HU Configurar parámetros de red.

**Configurar Lista de Control de Acceso(ACL)**

Historia de Usuario	
<b>Número:</b> HMAST_Proxy_4	<b>Nombre Historia de Usuario:</b> Configurar Lista de Control de Acceso(ACL).
<b>Modificación de Historia de Usuario Número:</b> Ninguna.	
<b>Usuario:</b> Alexander Pérez Tasé	<b>Iteración Asignada:</b> 1.
<b>Prioridad en Negocio:</b> Alta.	<b>Puntos Estimados:</b> 1,2 semana.
<b>Riesgo en Desarrollo:</b> Alta.	<b>Puntos Reales:</b> 1,2 semana.
<b>Descripción:</b> El módulo mostrará los parámetros de configuración de las listas de control de acceso (ACL). También permitirá adicionar, modificar y eliminar las mismas.	
<p><b>Observaciones:</b> Las ACL definen el origen o destinos de las peticiones que realiza el cliente y están definidas por la directiva ACL y su sintaxis es la siguiente:</p> <p>ACL nombreACL tipoACL valor valor1 ...</p> <p>ACL nombreACL tipoACL "rutaArchivo"</p> <p>donde rutaArchivo es la dirección donde se encuentra el fichero con los valores.</p> <p>Para gestionar las ACL se selecciona la opción ACL del menú principal y se mostrará una lista con todas las ACL creadas.</p> <p>✓Para <b>Crear</b> una nueva <b>ACL</b>, se selecciona la opción Nueva ACL, la cual permite introducir el nombre de la ACL, el cual es un valor alfanumérico sin espacios. Se podrá habilitar además, el ancho de banda de la red para esa ACL, este valor serán un número mayor que 0, además se podrá seleccionar la unidad de la velocidad de transferencia (Bytes, Kb, Mb o Gb).</p> <p>✓Para <b>Modificar</b> una <b>ACL</b>, se selecciona la ACL que se desea modificar del listado con todas las ACL existentes y serán mostrados los valores que posee la ACL los cuales pueden ser modificados. Estos valores antes de ser aceptados para su modificación tendrán las mismas validaciones que el proceso de creación.</p> <p>Para <b>Eliminar</b> una <b>ACL</b>, se muestra un listado con todas las ACL existentes, se selecciona la o las</p>	

ACL que se quieren eliminar las cuales son eliminadas cuando se selecciona la opción de Eliminar ACL.
<b>Prototipo de interfaz: Anexo 7: Administrar ACL</b>

Tabla 6: HU Configurar Lista de Control de Acceso(ACL).

### Configurar ACL de tipo origen

Historia de Usuario	
<b>Número:</b> HMAST_Proxy_5	<b>Nombre Historia de Usuario:</b> Configurar ACL de tipo origen.
<b>Modificación de Historia de Usuario Número:</b> Ninguna.	
<b>Usuario:</b> Alexander Pérez Tasé	<b>Iteración Asignada:</b> 1.
<b>Prioridad en Negocio:</b> Alta.	<b>Puntos Estimados:</b> 0,1 semanas.
<b>Riesgo en Desarrollo:</b> Alta.	<b>Puntos Reales:</b> 0,1 semanas.
<b>Descripción:</b> El módulo mostrará los parámetros de configuración de las listas de control de acceso (ACL). También permitirá adicionar, modificar y eliminar los valores de la misma.	
<b>Observaciones:</b> Para configurar una ACL de tipo origen se muestran los mismo parámetros definidos en HU_Proxy_4, variando solamente la parte de los valores.  Los valores de la ACL podrán ser indicados de dos formas, introduciendo los valores manualmente o indicando la dirección del fichero que contendrá dichos valores. Los valores introducidos manualmente serán almacenados en una tabla, a la cual se le podrá editar y eliminar los mismos.  ✓Para <b>Adicionar</b> un nuevo valor para una <b>ACL de tipo origen</b> , se escoge la opción Adicionar, la cual permitirá introducir una dirección o rango IP. Estas direcciones deben ser IPv4 o IPv6, y en caso de ser un rango los ip estarán separados por un signo menos (-), además permitirá la opción de agregar una nueva dirección o rango. También permite seleccionar valores predefinidos (0, 4, 8, 16, 24, 32) para definir la máscara de red.  ✓Para <b>Modificar</b> un valor a una <b>ACL de tipo origen</b> , se selecciona el valor que se desea modificar de la tabla de valores de la ACL y se muestra un formulario con los valores cuando es	

<p>seleccionado el botón Modificar. Estos valores antes de ser aceptados para su modificación tendrán las mismas validaciones que el proceso de creación.</p> <p>✓Para <b>Eliminar</b> un <b>valor</b> a una <b>ACL</b> de <b>tipo origen</b>, de la tabla de valores de la ACL, se selecciona el o los valores que se quieren eliminar a través de checkbox y estos son eliminadas cuando se selecciona el botón de Eliminar.</p>
<p><b>Prototipo de interfaz: Anexo 8: Crear ACL de tipo origen Anexo 9: Adicionar valor a ACL de tipo origen</b></p>

Tabla 7: HU Configurar ACL de tipo origen.

### Configurar ACL de tipo destino

Historia de Usuario	
<b>Número:</b> HMAST_Proxy_6	<b>Nombre Historia de Usuario:</b> Configurar Lista de Control de Acceso(ACL) de tipo destino.
<b>Modificación de Historia de Usuario Número:</b> Ninguna.	
<b>Usuario:</b> Alexander Pérez Tasé	<b>Iteración Asignada:</b> 1.
<b>Prioridad en Negocio:</b> Alta.	<b>Puntos Estimados:</b> 0,1 semana.
<b>Riesgo en Desarrollo:</b> Alta.	<b>Puntos Reales:</b> 0,1 semanas.
<b>Descripción:</b> El módulo mostrará los parámetros de configuración de las listas de control de acceso (ACL). También permitirá adicionar, modificar y eliminar los valores de la misma.	
<b>Observaciones:</b> Para configurar una ACL de tipo destino se muestran los mismos parámetros definidos en HU_Proxy_4, variando solamente la parte de los valores.	
<p>Los valores de las ACL podrán ser indicados de dos formas, introduciendo los valores manualmente o indicando la dirección del fichero que contendrá dichos valores. Los valores introducidos manualmente serán almacenados en una tabla, a la cual se le podrá editar y eliminar los mismos.</p> <p>✓Para <b>Adicionar</b> un nuevo valor para una <b>ACL de tipo destino</b>, se selecciona la opción Adicionar, la cual permitirá introducir una dirección IP, estas deben ser IPv4 o IPv6, y se permitirá la posibilidad de agregar una nueva dirección en caso de solicitarlo. También cuenta con valores</p>	

<p>predefinidos (0, 4, 8, 16, 24, 32) para definir la máscara de red.</p> <p>✓Para <b>Modificar</b> un valor a una <b>ACL de tipo destino</b>, se selecciona el valor que se desea modificar mostrándose los datos pertenecientes a este cuando es seleccionado la opción Modificar. Estos valores antes de ser aceptados para su modificación tendrán las mismas validaciones que el proceso de creación.</p> <p>Para <b>Eliminar</b> un valor a una <b>ACL de tipo destino</b>, de la tabla de valores de la ACL, se selecciona el o los valores que se deseen eliminar, los cuales serán borrados al escoger la opción de Eliminar.</p>
<p><b>Prototipo de interfaz: Anexo 10: Crear ACL de tipo destino Anexo 11: Adicionar valor a ACL de tipo destino</b></p>

Tabla 8: HU Configurar ACL de tipo destino.

### Configurar ACL de tipo dominio

Historia de Usuario	
<b>Número:</b> HMAST_Proxy_8	<b>Nombre Historia de Usuario:</b> Configurar ACL de tipo dominio.
<b>Modificación de Historia de Usuario Número:</b> Ninguna.	
<b>Usuario:</b> Alexander Pérez Tasé	<b>Iteración Asignada:</b> 1.
<b>Prioridad en Negocio:</b> Alta.	<b>Puntos Estimados:</b> 0,1 semanas.
<b>Riesgo en Desarrollo:</b> Alta.	<b>Puntos Reales:</b> 0,1 semanas.
<b>Descripción:</b> El módulo mostrará los parámetros de configuración de las listas de control de acceso (ACL). También permitirá adicionar, modificar y eliminar los valores de la misma.	
Observaciones: Para la creación de la ACL de tipo dominio se debe especificar si el dominio es de origen o de destino el cual se podrá especificar a través de dos radiobuton.	
Para configurar una ACL de tipo dominio se muestran los mismos parámetros definidos en HU_Proxy_4, variando solamente la parte de los valores.	
Los valores de las ACL podrán ser indicados de dos formas, introduciendo los valores manualmente o indicando la dirección del fichero que contendrá dichos valores. Los valores	

introducidos manualmente serán almacenados en una tabla, a la cual se le podrá editar y eliminar los mismos.

✓Para **Adicionar** un nuevo valor para una **ACL de tipo dominio**, se selecciona la opción Adicionar, la cual muestra un formulario que permitirá especificar si será entrada solo una expresión del dominio y la posibilidad de introducir el valor de dominio o la expresión, el cual será una cadena de caracteres alfanuméricos con puntos (.) intermedios sin espacios y nunca terminará en punto, además se permitirá agregar un nuevo dominio o expresión de dominio en caso de ser requerido.

✓Para **Modificar** un valor a una **ACL de tipo dominio**, se selecciona el valor que se desea modificar mostrándose los datos pertenecientes a este cuando es seleccionado la opción Modificar. Estos valores antes de ser aceptados para su modificación tendrán las mismas validaciones que el proceso de creación.

Para **Eliminar** un valor a una **ACL de tipo dominio**, de la tabla de valores de la ACL, se selecciona el o los valores que se deseen eliminar, los cuales serán borrados al escoger la opción de Eliminar.

**Prototipo de interfaz: Anexo 12: Crear ACL de tipo dominio Anexo 13: Adicionar valor a ACL de tipo dominio**

Tabla 9: HU Configurar ACL de tipo dominio.

### Configurar ACL de tipo url

Historia de Usuario	
<b>Número:</b> HMAST_Proxy_9	<b>Nombre Historia de Usuario:</b> Configurar ACL de tipo url.
<b>Modificación de Historia de Usuario Número:</b> Ninguna.	
<b>Usuario:</b> Alexander Pérez Tasé	<b>Iteración Asignada:</b> 1.
<b>Prioridad en Negocio:</b> Alta.	<b>Puntos Estimados:</b> 0,1 semanas.
<b>Riesgo en Desarrollo:</b> Alta.	<b>Puntos Reales:</b> 0,1 semanas.
<b>Descripción:</b> El módulo mostrará los parámetros de configuración de las listas de control de	

<p>acceso (ACL). También permitirá adicionar, modificar y eliminar los valores de la misma.</p>
<p><b>Observaciones:</b> Para configurar una ACL de tipo url se muestran los mismos parámetros definidos en HU_Proxy_4, variando solamente la parte de los valores.</p> <p>Los valores de las ACL podrán ser indicados de dos formas, introduciendo los valores manualmente o indicando la dirección del fichero que contendrá dichos valores. Los valores introducidos manualmente serán almacenados en una tabla, a la cual se le podrá editar y eliminar los mismos.</p> <p>✓Para <b>Adicionar</b> un nuevo valor para una <b>ACL de tipo url</b>, se selecciona la opción Adicionar, la cual permitirá especificar si será entrada solo una expresión de la url y la posibilidad de introducir la url o la expresión, la cual será una cadena de caracteres alfanuméricos y puntos intermedios sin espacios, los cuales también pueden empezar con los caracteres \, ^, pero nunca terminará en punto, además se permitirá agregar un nuevo dominio o expresión de dominio en caso de ser requerido.</p> <p>✓Para <b>Modificar</b> un valor a una <b>ACL de tipo url</b>, se selecciona el valor que se desea modificar mostrándose los datos pertenecientes a este cuando es seleccionado la opción Modificar. Estos valores antes de ser aceptados para su modificación tendrán las mismas validaciones que el proceso de creación.</p> <p>✓Para <b>Eliminar</b> un valor a una <b>ACL de tipo url</b>, de la tabla de valores de la ACL, se selecciona el o los valores que se deseen eliminar, los cuales serán borrados al escoger la opción de Eliminar.</p>
<p><b>Prototipo de interfaz: Anexo 14: Crear ACL de tipo url Anexo 15: Adicionar valor a ACL de tipo url</b></p>

Tabla 10: HU Configurar ACL de tipo url.

### Configurar conexión con Directorio Activo

Historia de Usuario	
Número: HMAST_Proxy_14	Nombre Historia de Usuario: Configurar conexión con Directorio Activo.
Modificación de Historia de Usuario Número: Ninguna.	

<b>Usuario:</b> Alexander Pérez Tasé	<b>Iteración Asignada:</b> 1.
<b>Prioridad en Negocio:</b> Alta.	<b>Puntos Estimados:</b> 0,1 semana.
<b>Riesgo en Desarrollo:</b> Alta.	<b>Puntos Reales:</b> 0,1 semanas.
<p><b>Descripción:</b> El módulo mostrará los parámetros de configuración para la validación de los usuarios que se autenticarán teniendo como origen un Directorio Activo(AD). También permitirá adicionar y modificar los mismos.</p>	
<p><b>Observaciones:</b> La directiva auth_param define el programa auxiliar que usará squid para validar los usuarios dependiendo del origen, en el caso del AD de utiliza el programa squid_ldap_auth, al cual se le establecerá la dirección del AD, el dominio y un usuario y contraseña para realizar las consultas de las validaciones.</p> <p>✓<b>Mostrar datos de conexión con AD:</b> El módulo muestra los datos de la conexión con AD (Dirección, dominio, grupo organizacional y credenciales).</p> <p>✓Para <b>Establecer</b> los <b>parámetros de conexión con un AD</b>, el módulo permitirá introducir la dirección del mismo, la cual debe ser una dirección IPv4 o un nombre de dominio el cual será una cadena de caracteres alfanuméricos con puntos (.) intermedios. Además permitirá definir el dominio donde opera el AD, las credenciales para la conexión, donde el usuario tendrá el formato CN(Canonical Name), por defecto se establecerá el del administrador (CN=Administrator,OU=Users,DC=domio,DC=com) y la contraseña, que estará conformada por caracteres alfanuméricos y especiales.</p> <p>✓Para <b>Modificar los parámetros de conexión con un AD</b>, una vez mostrados los datos de configuración de la conexión existente, el módulo permitirá modificar los mismos, realizando las mismas validaciones que en el proceso de establecer.</p>	
<p><b>Prototipo de interfaz: Anexo 16: Configurar conexión con Directorio Activo</b></p>	

Tabla 11: HU Configurar conexión con Directorio Activo.

### Configurar parámetros de la cache

#### Historia de Usuario

<b>Número:</b> HMAST_Proxy_16	<b>Nombre Historia de Usuario:</b> Configurar parámetros de la cache.
<b>Modificación de Historia de Usuario Número:</b> Ninguna.	
<b>Usuario:</b> Alexander Pérez Tasé	<b>Iteración Asignada:</b> 1.
<b>Prioridad en Negocio:</b> Media.	<b>Puntos Estimados:</b> 0,1 semanas.
<b>Riesgo en Desarrollo:</b> Alta.	<b>Puntos Reales:</b> 0,1 semanas.
<b>Descripción:</b> El módulo mostrará los parámetros de configuración de la cache y almacenamiento del servidor proxy. También permitirá adicionar, modificar y eliminar los mismos.	
<p><b>Observaciones:</b> El tamaño de la cache especifica cuanta memoria usará el servidor para el cache de objetos de memoria está definido por la directiva <code>cache_men</code> y su sintaxis es <code>cache_men tamaño Mb</code>. La ruta de almacenamiento de los objetos de la cache en disco está definida por la directiva <code>cache_dir</code>, este parámetro define la ruta del directorio, el tipo de cache de disco y su sintaxis es la siguiente:</p> <pre>cache_dir TipoFichero Directorio Tamaño Nivel1 Nivel2</pre> <p>Para configurar los parámetros de la cache se selecciona la opción <code>Parámetros de</code> del menú principal.</p> <p>✓<b>Mostrar datos de configuración de la cache:</b> El módulo permitirá mostrar los datos de configuración de la cache.</p> <p>✓Para <b>Establecer tamaño de memoria de la cache</b>, el módulo permitirá introducir el tamaño de memoria RAM que será dedicada para almacenar los objetos de cache, este valor será un número mayor que 0 y menor que el tamaño de la memoria RAM del servidor.</p> <p>✓Para <b>Modificar el tamaño de memoria de la cache</b>, se modifica valor, al cual se le harán las mismas validaciones que en el caso de establecimiento.</p> <p>✓Para <b>Establecer dirección de almacenamiento</b>, se establece el valor en el campo de texto para la dirección del directorio de almacenamiento. Esta dirección deberá ser una dirección válida dentro del servidor.</p>	

<p>✓Para <b>Modificar dirección de almacenamiento</b>, se modifica el campo de texto Tamaño de la Cache (RAM), al cual se le harán las mismas validaciones que en el caso de establecimiento.</p> <p>✓Para <b>Establecer tamaño del directorio de almacenamiento</b>, se establece el valor en el campo de texto para el almacenamiento, este valor será un número mayor que 0 y menor que la capacidad de almacenamiento del disco.</p> <p>✓Para <b>Modificar tamaño del directorio para almacenare</b>, se modifica el campo de texto Tamaño Directorio Almacenamiento, al cual se le harán las mismas validaciones que en el caso de establecer.</p> <p>✓ Para <b>Establecer tamaño máximo de objetos a almacenar</b>, se establece el valor en el campo de texto para el tamaño máximo de los objetos a almacenar, este valor será un número mayor que 0. También se muestra un combobox para seleccionar la unidad de almacenamiento de información (Kb,Mb,Gb).</p> <p>✓Para <b>Modificar tamaño máximo de objetos a almacenar</b>, se modifica el campo de texto Tamaño Máximo Obj, al cual se le harán las mismas validaciones que en el caso de establecer.</p>
<p><b>Prototipo de interfaz: Anexo 17: Configurar parámetros de la cache</b></p>

Tabla 12: HU Configurar parámetros de la cache.

### Configurar cuota de usuarios

Historia de Usuario	
<b>Número:</b> HMAST_Proxy_19	<b>Nombre Historia de Usuario:</b> Configurar cuota de usuarios.
<b>Modificación de Historia de Usuario Número:</b> Ninguna.	
<b>Usuario:</b> Alexander Pérez Tasé	<b>Iteración Asignada:</b> 1.
<b>Prioridad en Negocio:</b> Media.	<b>Puntos Estimados:</b> 1 semana.
<b>Riesgo en Desarrollo:</b> Alta.	<b>Puntos Reales:</b> 1 semanas.
<b>Descripción:</b> El módulo mostrará un listado con los usuarios que tienen asignado una cuota. También permitirá adicionar, modificar y eliminar cuotas a los usuarios del servidor proxy.	

<p><b>Observaciones:</b> Para la asignación de cuotas de deberá establecer la siguiente estructura:</p> <p>usuario cuota / período</p> <p>Para configurar la cuota de los usuarios se selecciona la opción Usuarios del menú principal , el cual le mostrará una vista con tres pestañas, seleccionando la pestaña Cuotas.</p> <p>✓<b>Mostrar Cuotas por usuario:</b> El módulo muestra en una tabla la cuota por usuario que tienen permiso para navegar.</p> <p>✓Para <b>Adicionar una cuota</b> a un usuario, se selecciona la opción Adicionar la cual permitirá establecer el nombre de usuario a quien se le establecerá la cuota, este usuario deberá ser una cadena de caracteres alfanuméricos comenzando siempre con una letra y sin espacios. Además, se podrá definir la cuota asignada al usuario, el cual será un número mayor que 0, permitiendo indicar si será monitoreada por horas o por <i>MegaBytes</i> y el período en cual será usada la cuota (Diario, Semanal, Mensual, Anual).</p> <p>✓Para <b>Modificar una cuota</b> a un usuario, se selecciona la cuota de usuario que se desea modificar de la tabla y son mostrados los valores cuando es seleccionado la opción Modificar. Estos valores antes de ser aceptados para su modificación tendrán las mismas validaciones que el proceso de creación.</p> <p>✓<b>Visualizar el estado de la cuenta:</b> El módulo debe brindar la dirección donde el usuario podrá ver el estado de su cuenta.</p> <p><b>Prototipo de interfaz: Anexo 18: Configurar cuota de usuarios Anexo 19: Adicionar nueva cuota a un usuario</b></p>
---

Tabla 13: HU Configurar cuota de usuarios.

**Configurar el formato de almacenamiento de los logs**

Historia de Usuario	
<p><b>Número:</b> HMAST_Proxy_20</p>	<p><b>Nombre Historia de Usuario:</b> Configurar el formato de almacenamiento de los logs.</p>
<p><b>Modificación de Historia de Usuario Número:</b> Ninguna.</p>	

<b>Usuario:</b> Alexander Pérez Tasé	<b>Iteración Asignada:</b> 1.
<b>Prioridad en Negocio:</b> Muy Alta.	<b>Puntos Estimados:</b> 1 semana.
<b>Riesgo en Desarrollo:</b> Alta.	<b>Puntos Reales:</b> 1 semana.
<b>Descripción:</b> El módulo mostrará una tabla con los formatos definidos de almacenamiento de los logs, además de permitir adicionar, modificar y eliminar los mismos. También se podrá definir cuál formato será el establecido y el directorio donde serán almacenados.	
<b>Observaciones:</b> Para configurar el formato de almacenamiento de los logs se selecciona la opción Logs del menú principal.	
<ul style="list-style-type: none"> <li>•Para <b>Establecer formato de los logs</b>, se permite al usuario seleccionar el formato de logs a aplicar, el cual podrá ser seleccionado entre los formatos definidos anteriormente.</li> <li>•Para <b>Adicionar un formato de logs</b>, se selecciona la opción Adicionar, la cual permitirá establecer el nombre del formato, que debe ser una cadena de caracteres alfanuméricos sin espacios. Estos formatos son expresiones definidas por squid y podrán ser introducidas en forma de cadena.</li> <li>•Para <b>Modificar un formato de logs</b>, se selecciona el formato que se desea modificar de la tabla y se le muestran un los valores cuando es seleccionado la opción Modificar. Estos valores antes de ser aceptados para su modificación tendrán las mismas validaciones que el proceso de creación.</li> <li>•Para <b>Establecer la dirección de almacenamiento de los logs</b>, se le permite al usuario definir la dirección donde se desean almacenar. Esta dirección debe ser una dirección válida dentro del servidor.</li> </ul>	
<b>Prototipo de interfaz:</b>	

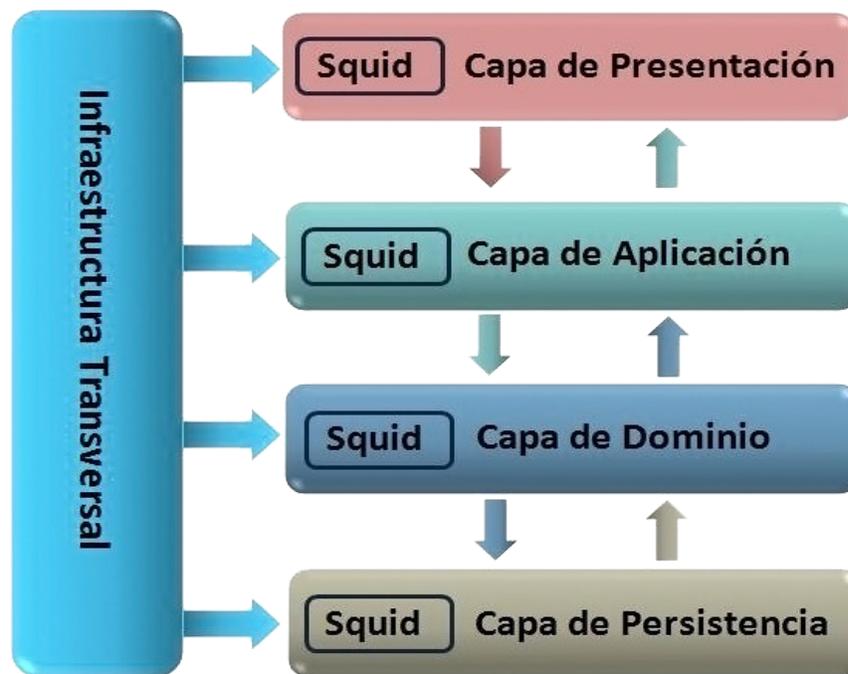
Tabla 14: HU Configurar el formato de almacenamiento de los logs.

## 2.4 Arquitectura del módulo

El proceso de diseño de la arquitectura tiene que decidir qué funcionalidad es la más importante a desarrollar. A partir de esta decisión se selecciona el tipo de aplicación y el estilo arquitectural, y tomar las decisiones importantes sobre seguridad y rendimiento que afectan al conjunto del sistema. El diseño de la

arquitectura decide cuales son los componentes más básicos del sistema y como se relacionan entre ellos para implementar la funcionalidad [24].

Con el objetivo de mantener una homogeneidad entre los componentes del sistema, se define la utilización de la arquitectura establecida para HMAST (ver Figura 2), incluyendo en esta la distribución del módulo para la administración del servicio proxy, entre las diferentes capas según las responsabilidades que estas establezcan. Además, es representado el flujo de información entre las diferentes capas del módulo, indicado por las flechas representadas. A continuación se muestra una representación de la arquitectura del módulo:



## 2.5 Diagrama de Paquetes

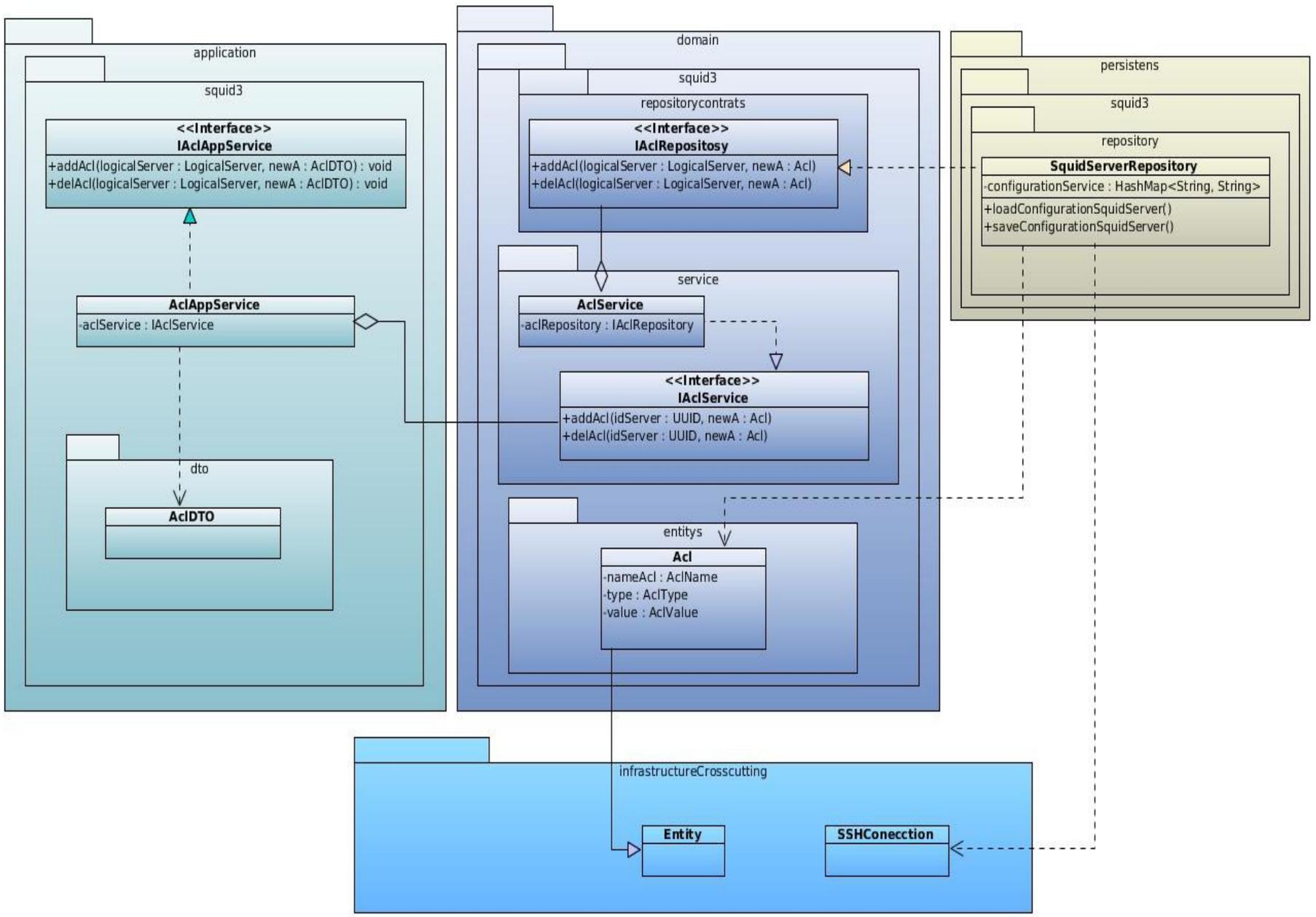
El diagrama de paquetes muestra la forma en que un sistema está dividido en agrupaciones lógicas mostrando las dependencias entre las mismas. Dado que normalmente un paquete está pensado como un directorio, los diagramas de paquetes suministran una descomposición de la jerarquía lógica de un sistema.

El diagrama de la Figura 5 representa la distribución de las clases para la implementación de la historia de usuario configurar ACL. Debido a que el alcance del módulo no requiere de la implementación de una interfaz visual, no será contemplada la capa de Presentación (*presentation*). Esta capa es la responsable de presentar al usuario los conceptos de negocio mediante una interfaz de usuario (IU), facilitar la explotación de dichos procesos, informar sobre la situación de los procesos de negocio e implementar las reglas de validación.

En la capa de aplicación (*application*) se encuentra la clase interfaz IACLAppService que contiene métodos que permiten el acceso desde la capa de presentación, los cuales son implementados por la clase ACLAppService. Esta última es la responsable de adaptar la información que le llega desde la interfaz de usuario, a los requerimientos de los servicios del dominio, estos servicios son accedidos a través del atributo ACLService contenido en esta clase. Es capa contiene además, el paquete DTO (*Data Object Transfer*), el mismo almacena clases que representan la información que llega desde la capa de presentación.

La capa de dominio (*domain*) contiene los subpaquetes Entidades (*entitys*), Servicio (*service*) y Contratos de repositorios (*repositoryscontrats*). En Entidades se encuentra la clase ACL, quien representa las reglas de control de acceso definidas en el servidor proxy. El subpaquete Servicio contiene la clase IACLService que define métodos que son accedidos desde la capa de aplicación, implementados por la clase ACLService, quien es la encargada de realizar las validaciones de los datos antes de realizar las operaciones en el repositorio. El acceso al repositorio se realiza a través del atributo ACLRepository encontrado en esta clase. En el subpaquete Contratos de repositorios se encuentra la clase IACLRepository, la cual tiene definidos los métodos que son accedidos desde el subpaquete Servicio y son implementados por la clase SquidServerRepository encontrada en la capa de persistencia de los datos (*persistens*). Esta clase es responsable de cargar y salvar la información en los repositorios.

Las demás clases que reflejan el resto de las historias de usuarios, independientemente de las funcionalidades que estas contengan, presentan el mismo flujo de comunicación dentro de cada uno de los paquetes como quedó descrito anteriormente.



## 2.6 Patrones de diseño

Los patrones de diseño son el soporte de las soluciones a problemas comunes que surgen en el desarrollo de software. Son catalogados además, como soluciones simples y elegantes a problemas específicos del diseño orientado a objetos. En el modelo de diseño del software se tuvieron en cuenta los patrones de asignación de responsabilidades GRASP y los patrones GOF, los cuales son descritos a continuación.

### Patrones GRASP

En el diseño orientado a objetos, GRASP, son Patrones Generales de Software para Asignación de Responsabilidades. Aunque se considera que más que patrones propiamente dichos, son una serie de buenas prácticas de aplicación recomendable en el diseño de software [25]. A continuación, son descritos los patrones de este tipo que son utilizados para el diseño de la solución propuesta.

**Experto:** Es un patrón que se usa más que cualquier otro al asignar responsabilidades; es un principio básico que suele ser útil en el diseño orientado a objetos. Con el uso de este patrón se conserva el encapsulamiento, ya que los objetos se valen de su propia información para hacer lo que se les pide. Esto soporta un bajo acoplamiento, lo que favorece al hecho de tener sistemas más robustos y de fácil mantenimiento. Con la utilización de este patrón, se hizo posible definir dónde colocar en cada clase las funcionalidades que necesitan de esa información, dicha clase sería el experto en información.

**Creador:** La creación de objetos es una de las actividades más frecuentes en un sistema orientado a objetos. El patrón Creador guía la asignación de responsabilidades relacionadas con la creación de objetos. El propósito fundamental del mismo, es encontrar un creador que se debe conectar con el objeto producido en cualquier evento. Este patrón se utilizó para identificar qué clase A debe crear elementos de una clase B, apoyándose en que la clase A debería: contener, agregar, registrar, utilizar y tener los datos de inicialización de la clase B.

En el módulo, la utilización de este patrón es evidenciado cuando las clases presentes en la capa Aplicación crean instancia de las entidades para la conversión de los objetos DTO a objetos del dominio. Un ejemplo es cuando la clase ACLAppService crean un nuevo objeto de tipo ACL para enviarlo a la capa de dominio para su posterior validación.

**Bajo acoplamiento:** El acoplamiento es una medida de la fuerza con que una clase está conectada a otras. Acoplamiento bajo significa que una clase no depende de muchas otras. El uso de este patrón

permite que las clases no se afecten por cambios de otros componentes, haciendo posible que sean fáciles de entender y de reutilizar.

**Alta cohesión:** La cohesión es una medida de cuán relacionadas y enfocadas están las responsabilidades de una clase. Una alta cohesión caracteriza a las clases con responsabilidades estrechamente relacionadas que no realicen un trabajo enorme. Se hizo necesario la utilización de este patrón en el software en cuestión con el fin de controlar la complejidad de cada clase utilizada para mantener un buen comportamiento de las mismas, por esta razón, las que se identificaron con una amplia cantidad de funcionalidades, se dividieron en otras clases siguiendo el propósito de distribuir de forma equitativa el peso de la complejidad, manteniendo además, la coherencia entre ellas.

### **Patrones GOF**

GoF, acrónimo de Gang o Four, Pandilla de los Cuatro, en su traducción al español. Los patrones de diseño, conocidos como GoF se clasifican en tres grandes categorías basadas en su propósito: creacionales, estructurales y de comportamiento. A continuación, es detallado el que ha sido utilizado en el diseño de la solución.

**Patrón Solitario (*Singleton*):** Garantiza la existencia de una única instancia para una clase y la creación de un mecanismo de acceso global a dicha instancia. Es usado debido a la necesidad de trabajar con el mismo objeto en distintos momentos.

En el módulo, para realizar la conexión con el servidor al cual se le administrará el servicio proxy, se realiza una única instancia de la clase SSHConection, la cual está contenida en el paquete Infraestructura Transversal (*infrastructureCrosscutting*).

### **Conclusiones parciales**

Una vez identificadas las principales funcionalidades que brindará el módulo a implementar, quedaron definidos 84 requisitos funcionales, los cuales fueron agrupados en 21 HU y 6 requerimientos no funcionales. Se logró una mayor comprensión entre el programador y el cliente a través de la descripción de las HU. Al concluir el capítulo queda definida una arquitectura en 5 capas basada en dominio la cual, con un correcto uso de los patrones de arquitectura y de diseño, es capaz de ser flexible ante cualquier cambio en la lógica del negocio.

### Capítulo 3: Implementación y pruebas del módulo

En el presente capítulo se describe el proceso de implementación, en el cual quedan definidas las iteraciones previstas para la implementación de los requisitos levantados. Son descritas las Tareas de Ingenierías propuestas para las Historias de Usuarios más relevantes, y los estándares de código a utilizar en la implementación del módulo. Además, se expondrán las pruebas realizadas al mismo para ganar en calidad, lo que garantiza que el módulo llegue a manos del cliente con el menor número de errores posibles.

#### 3.1 Plan de iteraciones

Una vez analizadas las Historias de Usuario (HU) por parte del cliente, quedando estimado el tiempo y esfuerzo dedicado para desarrollar cada una de ellas, se procede a realizar la planificación de las etapas de implementación del sistema. Este plan concentra las HU por iteraciones, definiendo cuáles serán desarrolladas en cada iteración del proceso de implementación. Teniendo en cuenta lo anteriormente planteado se decide implementar el sistema en dos iteraciones, las cuales son descritas a continuación:

- ✓ Iteración 1: La primera iteración tiene como objetivo dar cumplimiento a las HU relacionadas con la configuración básica del servidor proxy.
- ✓ Iteración 2: La segunda iteración tiene como finalidad dar cumplimiento a las HU relacionadas con las configuraciones avanzadas del servidor. Con la implementación de estas funcionalidades se completan las demás configuraciones básicas con las que debe contar el servidor.

#### 3.2 Tareas de Ingeniería

Tarea de Ingeniería para la HU Instalar servicio Squid:

Tarea de Ingeniería	
<b>Número Tarea:</b> 1	<b>Número Historia de Usuario:</b> HMAST_PROXY_1
<b>Nombre Tarea:</b> Estudiar comando para realizar la instalación del servidor.	
<b>Tipo de Tarea :</b> Desarrollo	<b>Puntos Estimados:</b> 0.1
<b>Fecha Inicio:</b>	<b>Fecha Fin:</b>
<b>Programador Responsable:</b> Alexander Pérez Tasé	
<b>Descripción:</b> Se ejecuta el comando <b>apt-get install squid3</b> , el cual tiene como finalidad la instalación del servidor squid.	

Posteriormente se comprueba su correcto funcionamiento.

**Tarea de Ingeniería para la HU Administrar servicio Squid:**

Tarea de Ingeniería	
<b>Número Tarea:</b> 2	<b>Número Historia de Usuario:</b> HMAST_PROXY_2
<b>Nombre Tarea:</b> Probar los comandos que permiten Iniciar, Detener, Recargar y Reiniciar el servicio.	
<b>Tipo de Tarea :</b> Desarrollo	<b>Puntos Estimados:</b> 0.1
<b>Fecha Inicio:</b>	<b>Fecha Fin:</b>
<b>Programador Responsable:</b> Alexander Pérez Tasé	
<p><b>Descripción:</b> Los comandos que permiten gestionar el servicio son los siguientes:</p> <ul style="list-style-type: none"> <li>- Iniciar: <b>service squid3 start</b></li> <li>- Detener: <b>service squid3 stop</b></li> <li>- Reiniciar: <b>service squid3 restart</b></li> </ul> <p>Se ejecutan cada uno de estos comandos y se comprueban los resultados.</p>	

**Tarea de Ingeniería para la HU Configurar parámetros de red:**

Tarea de Ingeniería	
<b>Número Tarea:</b> 3	<b>Número Historia de Usuario:</b> HMAST_PROXY_3
<b>Nombre Tarea:</b> Verificar la directiva que permite la configuración del puerto y dirección escucha.	
<b>Tipo de Tarea :</b> Desarrollo	<b>Puntos Estimados:</b> 0.1
<b>Fecha Inicio:</b>	<b>Fecha Fin:</b>
<b>Programador Responsable:</b> Alexander Pérez Tasé	
<p><b>Descripción:</b> El fichero de configuración del servidor Squid, ubicado en /etc/squid3/squid.conf, cuenta con la directiva http_port, la cual establece el o los puertos por los cuales el servidor escuchará las peticiones de conexión, también se puede especificar la dirección de la tarjeta de red por la cual se desea escuchar las peticiones en caso de tener más de una, esta estará</p>	

separada del puerto por dos puntos (:). El puerto debe ser un número mayor que 0 y menores que 65535, en caso que se especifique la dirección de la tarjeta de red, esta podrá ser un nombre de dominio o una dirección IPv4. La directiva establece por defecto el puerto 3128.

Ejemplo:

http\_port 3128

http\_port proxy.com:8080

http\_port 10.0.0.1:8080

**Tarea de Ingeniería para la HU Configurar Lista de Control de Acceso(ACL):**

Tarea de Ingeniería	
<b>Número Tarea:</b> 4	<b>Número Historia de Usuario:</b> HMAST_PROXY_4
<b>Nombre Tarea:</b> Probar la directiva que permite establecer una ACL.	
<b>Tipo de Tarea :</b> Desarrollo	<b>Puntos Estimados:</b> 0.1
<b>Fecha Inicio:</b>	<b>Fecha Fin:</b>
<b>Programador Responsable:</b> Alexander Pérez Tasé	
<p><b>Descripción:</b> El fichero de configuración squid.conf, cuenta con la directiva ACL, la cual define el origen o destino de la petición realizada por el cliente. La directiva de configuración ACL define los elementos de ACL, su estructura es:</p> <p>ACL nombreACL tipoACL valor valor1 ...</p> <p>ACL nombreACL tipoACL "archivo"</p>	

**Tarea de Ingeniería para la HU Establecer política de acceso para las ACLs:**

Tarea de Ingeniería	
<b>Número Tarea:</b> 5	<b>Número Historia de Usuario:</b> HMAST_PROXY_5
<b>Nombre Tarea:</b> Probar directiva que permite establecer la política de acceso de una ACL.	
<b>Tipo de Tarea :</b> Desarrollo	<b>Puntos Estimados:</b> 0.1
<b>Fecha Inicio:</b>	<b>Fecha Fin:</b>

<b>Programador Responsable:</b> Alexander Pérez Tasé
<b>Descripción:</b> El fichero de configuración squid.conf, cuenta con la directiva http_acces, define el tipo de acceso a uno o más elementos de ACL y su estructura es:  http_access allow   deny [!]ACLname ACLname2 ...

**Tarea de Ingeniería para la HU Configurar parámetros para la comunicación y uso de la cache de otro servidor proxy:**

Tarea de Ingeniería	
<b>Número Tarea:</b> 6	<b>Número Historia de Usuario:</b> HMAST_PROXY_18
<b>Nombre Tarea:</b> Probar la directiva relacionada con la comunicación son servidores proxy padres o hermanos.	
<b>Tipo de Tarea :</b> Desarrollo	<b>Puntos Estimados:</b> 0.1
<b>Fecha Inicio:</b>	<b>Fecha Fin:</b>
<b>Programador Responsable:</b> Alexander Pérez Tasé:	
<b>Descripción:</b> En el fichero de configuración squid.conf, la directiva icp_port especifica el puerto por el cual el servidor proxy escuchará las peticiones ICP, la directiva cache_peer se utiliza para especificar otros proxy-cache en una jerarquía como padres o como hermanos, es decir, definir si hay un proxy adelante o en paralelo. La sintaxis básica es la siguiente:  cache_peer servidor tipo http_port icp_port opciones	

### 3.3 Estándares de codificación

Un estándar de codificación completo comprende todos los aspectos de la generación de código. Si bien los programadores deben implementar un estándar de forma prudente, este debe tender siempre a lo práctico. Un código fuente completo debe reflejar un estilo armonioso, como si un único programador hubiera escrito todo el código de una sola vez [26].

#### **Estándar para nombrar las clases:**

Los nombres de las clases deben comenzar con la primera letra en mayúscula y el resto se pondrá con minúscula, cuando sea un nombre compuesto se utilizará la notación Pascal.

Ejemplo: *SquidServerRepository*

**Estándar para nombrar las funciones:**

El nombre a emplear para las funciones se escribe con la primera palabra en minúscula, en caso de que sea un nombre compuesto se utiliza la notación Camel, y con sólo leerlo se reconoce el propósito de la misma.

Ejemplo: *loadConfigurationSquidServer*

**Estándar para nombrar las variables:**

El nombre de estas variables se escribe la primera palabra con minúscula, si es un nombre compuesto se utilizará notación Camel.

Ejemplo: *ACLName*

**Estándar para nombrar los componentes:**

Todos los paquetes comienzan con *cu.uci.hmast.xxx.yyy.zzz.kkk*

*xxx* →presentation, application, domain, persistence.

*yyy* →nombre del módulo (squid3, dhcp3, bind9).

*zzz* →elementos que pueden contener los componentes verticales (entitys, repositorys).

*kkk* → clases o subpaquetes.

Ejemplo: *cu.uci.hmast.domain.squid3.entitys.ACL*

### 3.4 Pruebas de software

El único instrumento adecuado para determinar el estado de la calidad de un producto de software es el proceso de pruebas. En este proceso se ejecutan pruebas dirigidas a componentes del software o al sistema en su totalidad, con el objetivo de medir el grado en que la aplicación desarrollada cumple con los requerimientos [27]. Entre los métodos más efectivos para la realización de las pruebas de software se destacan el método de Caja Blanca y el de Caja Negra.

Debido a que el alcance del módulo a implementar no abarca el desarrollo de una interfaz gráfica, se realizarán pruebas unitarias con el uso del método de Caja Blanca, teniendo en cuenta las características que este ofrece. A continuación se describe el proceso de pruebas realizado.

### Pruebas de Caja Blanca

Las pruebas de Caja Blanca es un método utilizado para el diseño de casos de pruebas que se basan en un análisis detallado de la estructura del código, se comprueban los caminos lógicos, bucles y condiciones examinando así el estado del programa para informar de la situación real de la calidad del software.

En este tipo de pruebas se utilizan diferentes métodos con el objetivo de analizar el código por diferentes criterios como son:

- ✓ Pruebas de camino básico.
- ✓ Pruebas de condición.
- ✓ Pruebas de bucles.

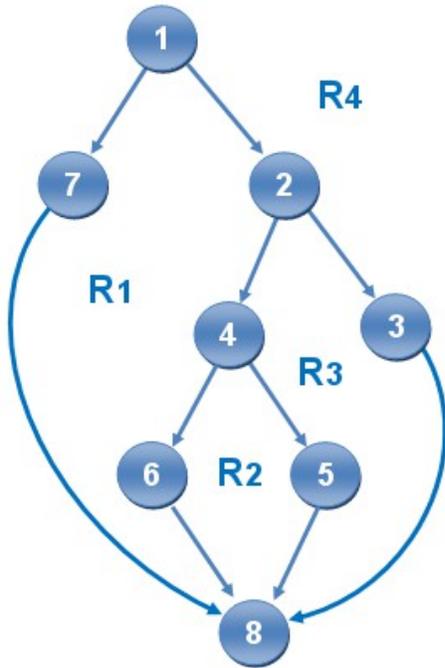
Para el desarrollo de las pruebas mediante el método de Caja Blanca, se utilizará la técnica del camino básico. Para aplicar esta técnica, primeramente se procede a enumerar el código que se desea probar como se muestra en la Figura 5.

```

1 public void addACL(UUID idServer, ACL newACL) throws ELogicalServerNotExist, EInvalidACL {
2     LogicalServer logicalServer = serverAdmin.getLogicalServer(idServer);
3     if (logicalServer != null) {
4         int pos = aclRepository.existACL(logicalServer, newACL);
5         if (pos == Integer.MAX_VALUE) {
6             aclRepository.addACL(logicalServer, newACL);
7         } else if (pos != Integer.MAX_VALUE && (String.valueOf(pos).charAt(0) != '-')) {
8             aclRepository.addValueToACL(logicalServer, aclRepository.getAllACLs(logicalServer), newACL, pos);
9         } else {
10            throw new EInvalidACL("ACL is already defined");
11        }
12    } else {
13        throw new ELogicalServerNotExist("Logical Server not Exist");
14    }
15 }

```

Una vez realizada la enumeración, se construye el grafo de flujo (ver Figura 6), teniendo en cuenta la notación para cada una de las instrucciones del código.



Luego se calcula la complejidad ciclomática mediante alguna de las siguientes fórmulas:

$$V(G) = \text{Aristas} - \text{Nodos} + 2 = 10 - 8 + 2 = 4$$

$$V(G) = \text{Nodos predicados} + 1 = 3 + 1 = 4$$

$$V(G) = \text{Número de regiones} = 4$$

El cálculo efectuado mediante las tres fórmulas ha dado como resultado el mismo valor, por lo que se puede afirmar que la complejidad ciclomática del código es 4 lo cual acota el límite superior para el número de caminos independientes que componen el conjunto básico y, consecuentemente, un valor límite superior para el número de pruebas que se deben diseñar y ejecutar con el propósito de garantizar que se cubran todas las sentencias de los procedimientos.

A continuación se describen los casos de pruebas para cada uno de los caminos identificados.

Caminos	Secuencia
Camino básico 1	1 - 7 - 8
Camino básico 2	1 - 2 - 3 - 8
Camino básico 3	1 - 2 - 4 - 5 - 8
Camino básico 4	1 - 2 - 4 - 6 - 8

#### Caso de Prueba para el Camino básico 1

**Nombre de quien realiza la prueba:** Alexander Pérez Tasé

**Estado:** Satisfactoria

<b>Descripción:</b> Para la ejecución de esta prueba se debe introducir el id de un LogicalServer y un objeto de tipo ACL, en caso que no exista un Objeto de tipo LogicalServer con el id pasado por parámetro, se lanza una excepción de tipo ELogicalServerNorExit.	
<b>Entradas:</b>	idServer, newACL
<b>Condición de Ejecución:</b> El valor de la variable idServer no se corresponde con ningún id de ninguno de los objetos de tipo LogicalServer.	
<b>Resultado:</b> La excepción de tipo ELogicalServerNorExit es lanzada y no se adiciona la nueva ACL.	

### Caso de Prueba para el Camino básico 2

<b>Nombre de quien realiza la prueba:</b> Alexander Pérez Tasé	<b>Estado:</b> Satisfactoria
<b>Descripción:</b> Para la ejecución de esta prueba se debe introducir el id de un LogicalServer y un objeto de tipo ACL, una vez comprobado que el id sea válido, se verifica la existencia de una ACL con el mismo nombre. En caso de no existir, se adiciona la nueva ACL al listado de ACL del servidor.	
<b>Entradas:</b>	idServer, newACL
<b>Condición de Ejecución:</b> El valor de la variable idServer se corresponde con el id de alguno de los objetos de tipo LogicalServer y el nombre de la nueva ACL a adicionar no se corresponde con ninguna de las ya existentes.	
<b>Resultado:</b> Se adiciona la nueva ACL.	

### Caso de Prueba para el Camino básico 3

<b>Nombre de quien realiza la prueba:</b> Alexander Pérez Tasé	<b>Estado:</b> Satisfactoria
<b>Descripción:</b> Para la ejecución de esta prueba se debe introducir el id de un LogicalServer y un objeto de tipo ACL, una vez comprobado que el id sea válido, se verifica la existencia de una ACL con el mismo nombre. En caso de existir alguna y esta posea el tipo de ACL igual que la nueva, los valores de la nueva ACL son adicionados a la ya existente.	

<b>Entradas:</b>	idServer, newACL
<b>Condición de Ejecución:</b> El valor de la variable idServer se corresponde con el id de alguno de los objetos de tipo LogicalServer y el nombre y el tipo de la nueva ACL a adicionar se corresponde con alguna de las ya existentes.	
<b>Resultado:</b> Los valores de la nueva ACL son adicionados a la ACL existente con el mismo nombre y tipo.	

#### Caso de Prueba para el Camino básico 4

<b>Nombre de quien realiza la prueba:</b> Alexander Pérez Tasé	<b>Estado:</b> Satisfactoria
<b>Descripción:</b> Para la ejecución de esta prueba se debe introducir el id de un LogicalServer y un objeto de tipo ACL, una vez comprobado que el id sea válido, se verifica la existencia de una ACL con el mismo nombre. En caso de existir, y posea un tipo diferente a la nueva, se lanza la excepción de tipo EInvalidACL.	
<b>Entradas:</b>	idServer, newACL
<b>Condición de Ejecución:</b> El valor de la variable idServer se corresponde con el id de alguno de los objetos de tipo LogicalServer, además el nombre se corresponda con alguna de las ya existentes, pero el tipo sea diferente.	
<b>Resultado:</b> La excepción de tipo EInvalidACL es lanzada y no se adiciona la nueva ACL.	

#### Pruebas realizadas por el cliente

Con el fin de demostrar el cumplimiento del objetivo planteado y la satisfacción del cliente con el resultado obtenido, se realizaron pruebas para validar la calidad del módulo implementado. Las mismas fueron realizadas a través de un Main, el cual es proyecto de pruebas en java que posee solo una clase que contiene un método principal donde se realizan llamadas a cada una de las funcionalidades implementadas.

#### Resultados de las pruebas

Luego de aplicar las pruebas al finalizar la primera iteración de desarrollo, fueron detectadas 4 no conformidades relacionadas con errores de validación de entidades, las cuales fueron erradicadas en su

totalidad para dar paso así a la segunda etapa de desarrollo. En esta etapa, se detectaron 2 no conformidades asociadas con errores de sintaxis. Un factor importante a tener en cuenta, es que durante el proceso de desarrollo fueron aplicadas pruebas a nivel de programador, lo cual permitió que en las pruebas finales de cada iteración de desarrollo, se obtuvieran un menor número de errores. Además, en las pruebas realizadas por el cliente se comprobó que el módulo cuenta con los requisitos establecidos por el mismo y por tanto, se encuentra listo para integrarse a HMAST.

### **Conclusiones parciales**

En el presente capítulo fueron definidas las iteraciones en las cuales fueron implementadas las HU en dependencia de su importancia, lo que permitió establecer un orden de prioridad para dicho proceso. Además, se describieron las Tareas de Ingeniería propuestas para las HU más significativas del módulo implementado, permitiendo una mejor organización y agilidad en el proceso de desarrollo. Una vez desarrollado el módulo, quedó detallado el procedimiento en que fueron aplicadas las pruebas unitarias a cada una de las funcionalidades del software, permitiendo detectar errores de validación existentes en el código, los cuales fueron erradicados en su totalidad. Culminado el proceso de implementación y pruebas del módulo, el mismo se encuentra en condiciones de ser integrado a HMAST.

## **Conclusiones**

Del estudio realizado en el presente trabajo se puede constatar que:

- ✓ El estudio realizado de los servidores en entornos de código abierto dirigidos a la implementación del servicio proxy, permitió seleccionar el servidor proxy Squid como alternativa más completa para administrar dicho servicio desde HMAST.
- ✓ El análisis y diseño del módulo definido permitió generar Tareas de Ingeniería correspondientes a cada Historia de Usuario que guiaran el proceso de implementación del módulo.
- ✓ El módulo desarrollado posee funcionalidades generales y avanzadas para la administración del servicio proxy, permitiendo la creación de ACL y la aplicación de las políticas de acceso a las mismas, además de poderlas combinar, la configuración de los parámetros de almacenamiento en la cache, así como la selección del origen de los datos de usuarios que se autentican para utilizar el servicio.
- ✓ Al concluir las pruebas, se obtiene un producto con el menor número de errores posibles, el cual permite la administración del servicio proxy integrado desde HMAST. Contribuyendo de esta forma, a la automatización de una de las principales tareas llevadas a cabo por el departamento SIMAYS en la migración a aplicaciones de código abierto en el país.

### **Recomendaciones**

Con la culminación del presente trabajo quedan algunas recomendaciones que pueden servir de punto de partida para mejorar

- ✓ Realizar una interfaz gráfica que permita administrar el servicio proxy en HMAST, basada en el diseño y la implementación descritas en el presente trabajo de diploma.
- ✓ Continuar con las investigaciones para añadir nuevas funcionalidades al módulo y obtener mejoras en futuras versiones.

### Referencias bibliográficas

- [1] EL DEBATE: ¿Las TIC reducen la distancia entre las empresas? [En Línea]. [Accedido 5 April 2013]. Disponible en: [http://www.revistacts.net/index.php?option=com\\_content&view=article&id=484:el-debate-ilas-tic-reducen-la-distancia-entre-las-empresas&catid=19:debates&Itemid=38](http://www.revistacts.net/index.php?option=com_content&view=article&id=484:el-debate-ilas-tic-reducen-la-distancia-entre-las-empresas&catid=19:debates&Itemid=38).
- [2] *Squid Proxy Server 3.1. Beginner's Guide* [online]. SAINI, KULBIR. 1ra. 2011. [Accedido 2 May 2013]. ISBN 978-1-849513-90-6. Disponible en: [http://www.k-max.name/wp-content/uploads/2012/02/Saini\\_K.\\_-\\_Squid\\_Proxy\\_Server\\_3.1\\_Beginners\\_Guide\\_2011.pdf](http://www.k-max.name/wp-content/uploads/2012/02/Saini_K._-_Squid_Proxy_Server_3.1_Beginners_Guide_2011.pdf).
- [3] Servicio Proxy-Cache. [En Línea]. [Accedido 18 May 2013]. Disponible en: <http://www.ujaen.es/sci/redes/proxy/>.
- [4] Servidores proxy y servidores de proxy inversos. [En Línea]. [Accedido 7 February 2013]. Disponible en: <http://es.kioskea.net/contents/lan/proxy.php3>.
- [5] Servidores Proxy Transparentes - Servicio de Web Hosting - Taller Digital VW. [En Línea]. [Accedido 7 February 2013]. Disponible en: <http://www.tallerdigitalvw.com/servidores/servidores-proxy-transparentes>.
- [6] What is Varnish? | Varnish Software. [En Línea]. [Accedido 6 February 2013]. Disponible en: <https://www.varnish-software.com/products-services/varnish-cache>.
- [7] Varnish Cache Reverse Proxy. [En Línea]. [Accedido 6 February 2013]. Disponible en: <http://marketingagil.com/2009/12/varnish-reverse-proxy/>.
- [8] Tinyproxy - A light-weight HTTP/HTTPS proxy. [En Línea]. [Accedido 6 February 2013]. Disponible en: <https://banu.com/tinyproxy/>.
- [9] Debian -- Details of package privoxy in sid. [En Línea]. [Accedido 11 February 2013]. Disponible en: <http://packages.debian.org/es/sid/privoxy>.
- [10] squid: Optimising Web Delivery. [En Línea]. [Accedido 6 February 2013]. Disponible en: <http://www.squid-cache.org/>.
- [11] Squid: servidor proxy-caché | Observatorio Tecnológico. [En Línea]. [Accedido 7 February 2013]. Disponible en: <http://recursostic.educacion.es/observatorio/web/ca/software/servidores/589-elvira-mifsud>.
- [12] Presentación — Documentación de Zentyal 3.0. [En Línea]. [Accedido 6 February 2013]. Disponible en: <http://doc.zentyal.org/es/presentation.html>.
- [13] Webmin, administra tu servidor y más cosas con esta interfaz web. [En Línea]. [Accedido 23 February 2013]. Disponible en: <http://linuxzone.es/2012/02/28/webmin-administra-tu-servidor-y-mas-cosas-con-esta-interfaz-web/>.
- [14] Definición de lenguaje de programación. [En Línea]. [Accedido 12 February 2013]. Disponible en: <http://www.definicion.org/lenguaje-de-programacion>.

- [15] Exes. Capítulo 2: Características del lenguaje java. *emagister* [online]. 6 February 2013. [Accedido 6 February 2013]. Disponible en: <http://www.emagister.com/curso-java/caracteristicas-lenguaje-java-1>.
- [16] Definición de XML - Qué es, Significado y Concepto. [En Línea]. [Accedido 13 February 2013]. Disponible en: <http://definicion.de/xml/>.
- [17] *El Lenguaje Unificado de Modelado*. BOOCH, G., RUMBAUGH, J. and JACOBSON. [no date].
- [18] Apuntes Informática Aplicada a la Gestión Pública. Capítulo 2, Ingeniería del software, metodologías de desarrollo. 2011/12. Universidad de Murcia (España). Rafael Barzanallana. [En Línea]. [Accedido 5 April 2013]. Disponible en: <http://www.um.es/docencia/barzana/IAGP/IAGP2-Metodologias-de-desarrollo.html>.
- [19] PEÑALVER ROMERO, GLADYS MARSÍ. *Metodología ágil para proyectos de software libre* [online]. UCI. S.I., 2008.
- [20] Welcome to NetBeans. [En Línea]. [Accedido 9 February 2013]. Disponible en: <http://netbeans.org/>.
- [21] UML, BPMN and Enterprise Architecture Tool for Software Development. [En Línea]. [Accedido 12 February 2013]. Disponible en: <http://www.visual-paradigm.com/>.
- [22] RapidSVN. [En Línea]. [Accedido 12 February 2013]. Disponible en: <http://www.rapidsvn.org/>.
- [23] BAPTISTE, JUAN LUIOS. Kde4 Se Libre. [En Línea]. 25 May 2013. [Accedido 25 May 2013]. Disponible en: <http://www.slideshare.net/campuspartycolombia/kde4-se-libre>.
- [24] *Guía de Arquitectura N-Capas orientada al Dominio con .NET 4.0*. CÉSAR, Zorrilla Castro, Unai de la Torre Llorente, and RAMOS BARROSO, MIGUEL ANGEL, Calvarro Nelso, Javier. 2010. 422 p. ISBN 978-84-936696-3-8.
- [25] GRASP. [En Línea]. [Accedido 30 May 2013]. Disponible en: <http://www.esacademic.com/dic.nsf/eswiki/1288764>.
- [26] Revisiones de código y estándares de codificación. [En Línea]. [Accedido 18 May 2013]. Disponible en: [http://msdn.microsoft.com/es-es/library/aa291591\(v=vs.71\).aspx](http://msdn.microsoft.com/es-es/library/aa291591(v=vs.71).aspx).
- [27] Las pruebas de software forman el único instrumento adecuado para determinar la calidad de software. [En Línea]. [Accedido 19 May 2013]. Disponible en: <http://pruebasdesoftware.com/laspruebasdesoftware.htm>.

## Bibliografía

- ✓ About | Varnish Community. [En línea]. [Accedido el 6 February 2013]. Disponible en: <https://www.varnish-cache.org/about>.
- ✓ Apuntes Informática Aplicada a la Gestión Pública. Capítulo 2, Ingeniería del software, metodologías de desarrollo. 2011/12. Universidad de Murcia (España). Rafael Barzanallana. [En Línea]. [Accedido 5 April 2013]. Disponible en: <http://www.um.es/docencia/barzana/IAGP/IAGP2-Metodologias-de-desarrollo.html>.
- ✓ BOOCH, G., RUMBAUGH, J. and JACOBSON. *El Lenguaje Unificado de Modelado*. S.I.: Addison-Wesley., [no date].
- ✓ Características de XML. [En línea]. [Accedido el 13 February 2013]. Disponible en: <http://trevinca.ei.uvigo.es/~txapi/espanol/proyecto/superior/memoria/node156.html>.
- ✓ Características del lenguaje java. [En Línea]. [Accedido 12 February 2013]. Disponible en: <http://www.emagister.com/curso-java/caracteristicas-lenguaje-java-1>.
- ✓ Debian -- Details of package privoxy in sid. [En Línea]. [Accedido 11 February 2013]. Disponible en: <http://packages.debian.org/es/sid/privoxy>.
- ✓ Definición de Java - Qué es, Significado y Concepto. [En línea]. [Accedido el 12 February 2013]. Disponible en: <http://definicion.de/java/>.
- ✓ Definición de lenguaje de programación. [En línea]. [Accedido el 12 February 2013]. Disponible en: <http://www.definicion.org/lenguaje-de-programacion>.
- ✓ Definición de XML - Qué es, Significado y Concepto. [En línea]. [Accedido el 13 February 2013]. Disponible en: <http://definicion.de/xml/>.
- ✓ Diccionario de Informática e Internet de Microsoft, Cuarta edición. [s.], McGraw-Hill/Interamericana.
- ✓ EL DEBATE: ¿Las TIC reducen la distancia entre las empresas? [En Línea]. [Accedido 5 April 2013]. Disponible en: [http://www.revistacts.net/index.php?option=com\\_content&view=article&id=484:el-debate-ilas-tic-reducen-la-distancia-entre-las-empresas&catid=19:debates&Itemid=38](http://www.revistacts.net/index.php?option=com_content&view=article&id=484:el-debate-ilas-tic-reducen-la-distancia-entre-las-empresas&catid=19:debates&Itemid=38).
- ✓ *El Lenguaje Unificado de Modelado*. BOOCH, G., RUMBAUGH, J. and JACOBSON. [no date].
- ✓ Exes. Capítulo 2: Características del lenguaje java. *emagister* [En línea]. 6 February 2013. [Accedido el 6 February 2013]. Disponible en: <http://www.emagister.com/curso-java/caracteristicas-lenguaje-java-1>.
- ✓ jQuery. [En línea]. [Accedido el 12 February 2013]. Disponible en: <http://jquery.com/>.

- ✓ Learning JSP: Primeros Pasos. [En línea]. [Accedido el 13 February 2013]. Disponible en: <http://learningjsp.blogspot.com/p/primeros-pasos.html>.
- ✓ MCGRAW and HILL. *Servidor Proxy*. Cuarta edición. S.l.: s.n., 2001. ISBN 84-481-2893.
- ✓ Mns spring framework - Core. [En línea]. [Accedido el 8 February 2013]. Disponible en: <http://www.slideshare.net/CarlosDiazValverde/mns-spring-framework-core>.
- ✓ PEÑALVER ROMERO, GLADYS MARSÍ. *Metodología ágil para proyectos de software libre* [En Línea]. UCI. S.l., 2008.
- ✓ Presentación — Documentación de Zentyal 3.0. [En Línea]. [Accedido 6 February 2013]. Disponible en: <http://doc.zentyal.org/es/presentation.html>.
- ✓ Privoxy 3.0.14 beta -. [En línea]. [Accedido el 11 February 2013]. Disponible en: <http://mscerts.programming4.us/es/18976.aspx>.
- ✓ Proxy Transparente. [En línea]. [Accedido el 6 February 2013]. Disponible en: [http://dns.bdat.net/blog/index.php?option=com\\_content&view=article&id=397:proxy-transparente&catid=21:squid](http://dns.bdat.net/blog/index.php?option=com_content&view=article&id=397:proxy-transparente&catid=21:squid).
- ✓ RapidSVN. [En Línea]. [Accedido 12 February 2013]. Disponible en: <http://www.rapidsvn.org/>.
- ✓ Revisiones de código y estándares de codificación. [En Línea]. [Accedido 18 May 2013]. Disponible en: [http://msdn.microsoft.com/es-es/library/aa291591\(v=vs.71\).aspx](http://msdn.microsoft.com/es-es/library/aa291591(v=vs.71).aspx).
- ✓ Servicio de Proxy HTTP — Documentación de Zentyal 3.0. [En línea]. [Accedido el 23 February 2013]. Disponible en: <http://doc.zentyal.org/es/proxy.html>.
- ✓ Servicio Proxy-Cache. [En Línea]. [Accedido 18 May 2013]. Disponible en: <http://www.ujaen.es/sci/redes/proxy/>.
- ✓ Servidores Proxy Cache - Servicio de Web Hosting - Taller Digital VW. [En línea]. [Accedido el 7 February 2013]. Disponible en: <http://www.tallerdigitalvw.com/servidores/servidores-proxy-cache>.
- ✓ Servidores Proxy Transparentes - Servicio de Web Hosting - Taller Digital VW. [En Línea]. [Accedido 7 February 2013]. Disponible en: <http://www.tallerdigitalvw.com/servidores/servidores-proxy-transparentes>.
- ✓ Servidores proxy y servidores de proxy inversos. [En línea]. [Accedido el 7 February 2013]. Disponible en: <http://es.kioskea.net/contents/lan/proxy.php3>.
- ✓ Spring Framework | SpringSource.org. [En Línea]. [Accedido 12 February 2013]. Disponible en: <http://www.springsource.org/spring-framework>.
- ✓ squid: Optimising Web Delivery. [En Línea]. [Accedido 6 February 2013]. Disponible en:

<http://www.squid-cache.org/>.

- ✓ *Squid Proxy Server 3.1. Beginner's Guide* [En Línea]. SAINI, KULBIR. 1ra. 2011. [Accedido 2 May 2013]. ISBN 978-1-849513-90-6. Disponible en: [http://www.k-max.name/wp-content/uploads/2012/02/Saini\\_K.\\_-\\_Squid\\_Proxy\\_Server\\_3.1\\_Beginners\\_Guide\\_2011.pdf](http://www.k-max.name/wp-content/uploads/2012/02/Saini_K._-_Squid_Proxy_Server_3.1_Beginners_Guide_2011.pdf).
- ✓ Squid: servidor proxy-caché | Observatorio Tecnológico. [En Línea]. [Accedido el 7 February 2013]. Disponible en: <http://recursostic.educacion.es/observatorio/Web/ca/software/servidores/589-elvira-mifsud>.
- ✓ *Squid: The Definitive Guide* [En Línea]. WESSELS, DUANE. 2004. 496 p. [Accedido 2 May 2013]. ISBN 0-596-00162-2. Disponible en: [http://repo.meh.or.id/Network n Security/Squid The Definitive Guide.pdf](http://repo.meh.or.id/Network%20Security/Squid%20The%20Definitive%20Guide.pdf).
- ✓ Tinyproxy - A light-weight HTTP/HTTPS proxy. [En Línea]. [Accedido 6 February 2013]. Disponible en: <https://banu.com/tinyproxy/>.
- ✓ tinyproxy 1.7.0 -. [En Línea]. [Accedido el 6 February 2013]. Disponible en: <http://mscerts.programming4.us/es/310350.aspx>.
- ✓ UML, BPMN and Enterprise Architecture Tool for Software Development. [En Línea]. [Accedido el 12 February 2013]. Disponible en: <http://www.visual-paradigm.com/>.
- ✓ Users List. Visual Paradigm for UML. *Visual Paradigm* [En Línea]. 3 February 2013. [Accedido el 3 February 2013]. Disponible en: <http://www.visual-paradigm.com/>.
- ✓ Varnish Cache Reverse Proxy. [En Línea]. [Accedido el 6 February 2013]. Disponible en: <http://marketingagil.com/2009/12/varnish-reverse-proxy/>.
- ✓ Varnish y WordPress | Ayuda WordPress. [En Línea]. [Accedido el 6 February 2013]. Disponible en: <http://ayudawordpress.com/varnish-y-wordpress/>.
- ✓ Webmin, administra tu servidor y más cosas con esta interfaz Web. [En Línea]. [Accedido el 23 February 2013]. Disponible en: <http://linuxzone.es/2012/02/28/Webmin-administra-tu-servidor-y-mas-cosas-con-esta-interfaz-Web/>.
- ✓ Welcome to NetBeans. [En Línea]. [Accedido el 9 February 2013]. Disponible en: <http://netbeans.org/>.
- ✓ What is Varnish? | Varnish Software. [En Línea]. [Accedido el 6 February 2013]. Disponible en: <https://www.varnish-software.com/products-services/varnish-cache>.

Anexos



Anexo 1: Interfaz de instalación 1



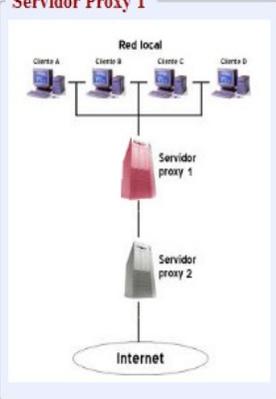
Anexo 2: Interfaz de instalación 2

DHCP 162.124.201.255	DNS 162.124.201.255	CORREO correo.uci.cu	FTP 162.124.201.255	WEB 162.124.201.255	PROXY correo.uci.cu
-------------------------	------------------------	-------------------------	------------------------	------------------------	------------------------

Instalar Servicio

Administración del Servicio proxy en : 192.168.2.1

Servidor Proxy 1



Configuraciones por defecto

Dirección de tarjeta de red Local 1:

Dirección de tarjeta de red Local 2:

Puerto de escucha:

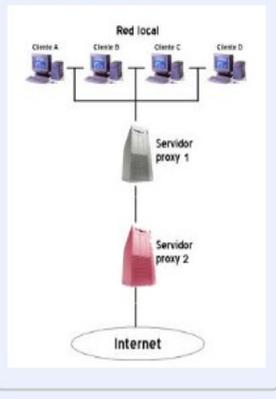
Anexo 3: Interfaz de instalación 3

DHCP 162.124.201.255	DNS 162.124.201.255	CORREO correo.uci.cu	FTP 162.124.201.255	WEB 162.124.201.255	PROXY correo.uci.cu
-------------------------	------------------------	-------------------------	------------------------	------------------------	------------------------

Instalar Servicio

Administración del Servicio proxy en : 192.168.2.1

Servidor Proxy 2



Configuraciones por defecto

Puerto de escucha:

Dirección de tarjeta de red escucha:

Establecer como Proxy Padre

\* Puerto ICP:

Anexo 4: Interfaz de instalación 4

66



Anexo 5: Administración del servicio

DHCP 162.124.201.255	DNS 162.124.201.255	CORREO correo.uci.cu	FTP 162.124.201.255	WEB 162.124.201.255	PROXY correo.uci.cu
-------------------------	------------------------	-------------------------	------------------------	------------------------	------------------------

ACL

Cache

Usuarios

Parámetros de Red

Logs

Administración del Servicio proxy en : 192.168.2.1

**Parámetros de Red**

\* Dirección de Escucha:  +

\* Puerto escucha:  +

\* Puerto escucha:  + X

Servidor DNS:  + X

Servidor DNS:  + X

Nombre Proxy:

Anexo 6: Parámetros de red

DHCP 162.124.201.255	DNS 162.124.201.255	CORREO correo.uci.cu	FTP 162.124.201.255	WEB 162.124.201.255	PROXY correo.uci.cu
-------------------------	------------------------	-------------------------	------------------------	------------------------	------------------------

ACL

Cache

Usuarios

Parámetros de Red

Logs

Administración del Servicio proxy en : 192.168.2.1

**ACL**

Mostrar: 5

Nombre ACL	Tipo	Política	Subir/Bajar
<input checked="" type="checkbox"/> local_network	src	Permitir	
<input type="checkbox"/> safe_port	port	Permitir	
<input type="checkbox"/> safe_domian	dstdomain	Permitir	
<input type="checkbox"/> bad_url	url_reg	Denegar	

<< 1 2 >>

Anexo 7: Administrar ACL

68

DHCP 162.124.201.255	DNS 162.124.201.255	CORREO correo.uci.cu	FTP 162.124.201.255	WEB 162.124.201.255	PROXY correo.uci.cu
-------------------------	------------------------	-------------------------	------------------------	------------------------	------------------------

ACL

Cache

Usuarios

Parámetros de Red

Logs

Administración del Servicio proxy en : 192.168.2.1
▶
▶
▶

**Nueva ACL**

\* Nombre:

\* Tipo: origen

\* Política: Permitir

**Días y Horario**

L  M  M  J  V  S  D

Hora Inicio:  - Hora Fin:

**Limitar ancho de banda**

Ancho de Banda:  Kb/s

**Valores**

**Insertar Valores**

+ Adicionar
✎ Editar
✖ Eliminar

Buscar
Mostrar: 5

IP / Rango IP	Máscara
192.168.1.0	/24
192.168.2.12	/32
192.168.2.15 - 192.168.2.25	/32

<< 1 2 >>

**Archivo**

...

✔ Guardar
✖ Cancelar

Anexo 8: Crear ACL de tipo origen

Administración del Servicio proxy en : 192.168.2.1

**Nueva ACL**

\* Nombre:

\* Tipo: **origen**

\* Política: **Permitir**

**Días y Horario**

L  M  M  J  V  S  D

Hora Inicio:  - Hora Fin:

**Limitar ancho de banda**

Ancho de Banda:  Kb/s

**Adicionar IP origen**

IP:

Rango de IP:  -

Máscara: **24**

Anexo 9: Adicionar valor a ACL de tipo origen

Administración del Servicio proxy en : 192.168.2.1

**Nueva ACL**

\* Nombre:

\* Tipo: **destino**

\* Política: **Permitir**

**Días y Horario**

L  M  M  J  V  S  D

Hora Inicio:  - Hora Fin:

**Limitar ancho de banda**

Ancho de Banda:  Kb/s

**Valores**

**Insertar Valores**

**Mostrar:** 5

IP	Máscara
192.168.1.0	/24
192.168.2.12	/32

<< 1 2 >>

**Archivo**

Anexo 10: Crear ACL de tipo destino

DHCP 162.124.201.255	DNS 162.124.201.255	CORREO correo.uci.cu	FTP 162.124.201.255	WEB 162.124.201.255	PROXY correo.uci.cu
-------------------------	------------------------	-------------------------	------------------------	------------------------	------------------------

Administración del Servicio proxy en : 192.168.2.1

**Nueva ACL**

\* Nombre:

\* Tipo: destino

\* Política: Permitir

**Días y Horario**

L  M  M  J  V  S  D

Hora Inicio:  Hora Fin:

**Limitar ancho de banda**

Ancho de Banda:  Kb/s

**Valores**

**Adicionar IP destino**

IP destino:

Máscara: 24

Anexo 11: Adicionar valor a ACL de tipo destino

DHCP 162.124.201.255	DNS 162.124.201.255	CORREO correo.uci.cu	FTP 162.124.201.255	WEB 162.124.201.255	PROXY correo.uci.cu
-------------------------	------------------------	-------------------------	------------------------	------------------------	------------------------

ACL

Cache

Usuarios

Parámetros de Red

Logs

Administración del Servicio proxy en : 192.168.2.1

**Nueva ACL**

\* Nombre:

\* Tipo: domino

**Dominios**

Origen  Destino

\* Política: Permitir

**Días y Horario**

L  M  M  J  V  S  D

Hora Inicio:  - Hora Fin:

**Limitar ancho de banda**

Ancho de Banda:  Kb/s

**Valores**

**Insertar Valores**

+ Adicionar
↔ Editar
✖ Eliminar

Buscar Mostrar: 5

Dominio / Expresión de dominio
gmail.com
yahoo.es

<< 1 2 >>

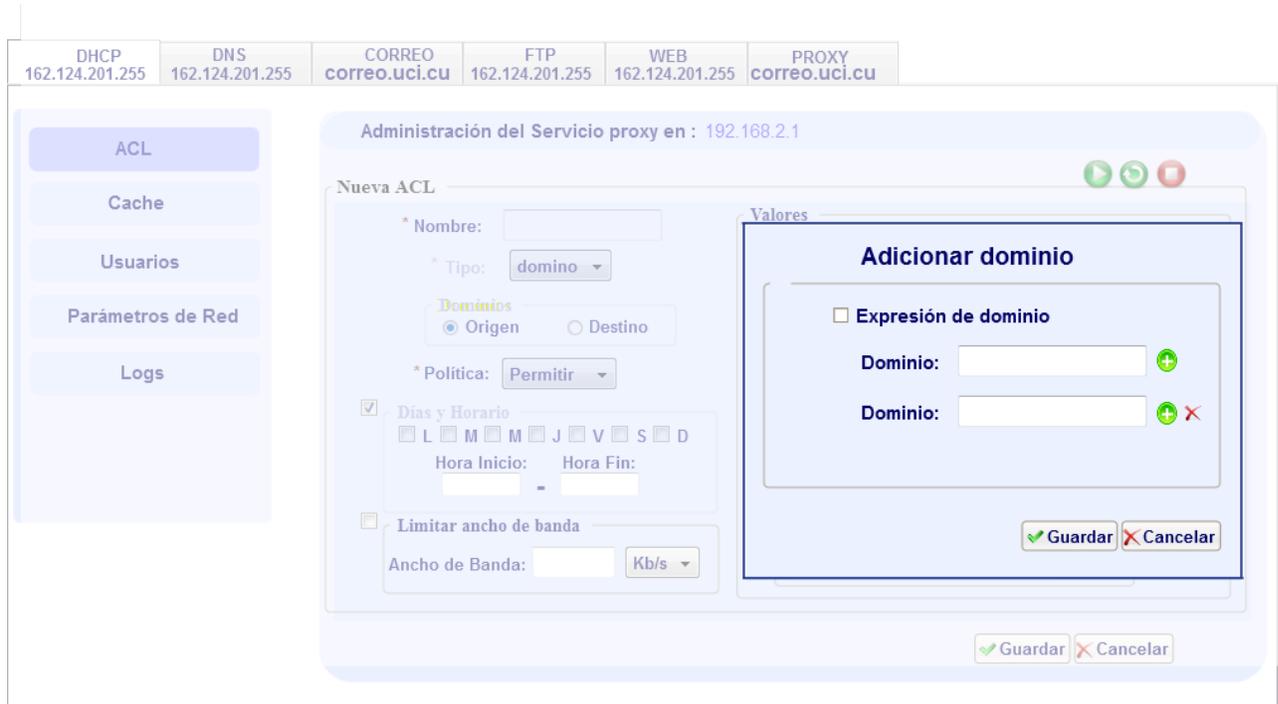
**Archivo**

...

✔ Guardar
✖ Cancelar

Anexo 12: Crear ACL de tipo dominio

72



Anexo 13: Adicionar valor a ACL de tipo dominio

DHCP 162.124.201.255	DNS 162.124.201.255	CORREO correo.uci.cu	FTP 162.124.201.255	WEB 162.124.201.255	PROXY correo.uci.cu
-------------------------	------------------------	-------------------------	------------------------	------------------------	------------------------

Administración del Servicio proxy en : 192.168.2.1

ACL

Cache

Usuarios

Parámetros de Red

Logs

**Nueva ACL**

\* Nombre:

\* Tipo:

\* Política:

**Días y Horario**

L  M  M  J  V  S  D

Hora Inicio:  - Hora Fin:

**Limitar ancho de banda**

Ancho de Banda:

**Valores**

**Insertar Valores**

Mostrar:

**Dominio / Expresión de dominio**

gmail.com

yahoo.es

**Archivo**

Anexo 14: Crear ACL de tipo url



Anexo 15: Adicionar valor a ACL de tipo url



Anexo 16: Configurar conexión con Directorio Activo

DHCP 162.124.201.255    DNS 162.124.201.255    CORREO correo.uci.cu    FTP 162.124.201.255    WEB 162.124.201.255    PROXY correo.uci.cu

ACL  
Cache  
Usuarios  
Parámetros de Red  
Logs

Administración del Servicio proxy en : 192.168.2.1

General Proxy Padre Otras Cache

\* Tamaño cache (RAM): 256 MB

Almacenamiento

\* Directorio:

\* Tipo de Fichero: UFS

\* Tamaño Directorio:  MB

\* Nivel 1:  \* Nivel 2:

\* Tamaño Máximo de Objeto:  KB

Guardar Cancelar

Anexo 17: Configurar parámetros de la cache

DHCP 162.124.201.255    DNS 162.124.201.255    CORREO correo.uci.cu    FTP 162.124.201.255    WEB 162.124.201.255    PROXY correo.uci.cu

ACL  
Cache  
Usuarios  
Parámetros de Red  
Logs

Administración del Servicio proxy en : 192.168.2.1

LDAP Local Cuotas

Nuevo usuario

usuario      Mostrar: 5   

Usuario	Cuota(MB)	Cuota(Horas)	Período
<input checked="" type="checkbox"/> atase	250	-	Mensual
<input type="checkbox"/> yasielpv	-	20	Mensual
<input type="checkbox"/> ypvillazon	100	-	Semanal
<input checked="" type="checkbox"/> avhernandez	200	-	Mensual
<input type="checkbox"/>			

<< 1 2 >>

Anexo 18: Configurar cuota de usuarios



Anexo 19: Adicionar nueva cuota a un usuario

## Glosario de términos

**DHCP:** Sigla en inglés de *Dynamic Host Configuration Protocol*, en español protocolo de configuración dinámica de host, es un protocolo de red que permite a los clientes de una red IP obtener sus parámetros de configuración automáticamente.

**SSH:** *Secure SHell*, en español intérprete de órdenes segura, es el nombre de un protocolo y del programa que lo implementa, y sirve para acceder a máquinas remotas a través de una red.

**FTP:** Siglas en inglés de File Transfer Protocol, en español Protocolo de Transferencia de Archivos, es un protocolo de red para la transferencia de archivos entre sistemas conectados a una red TCP (*Transmission Control Protocol*), basado en la arquitectura cliente-servidor.

**Framework:** Es una estructura conceptual y tecnológica de soporte definido, normalmente con artefactos o módulos de software concretos, que puede servir de base para la organización y desarrollo de software.

**SCRUM:** Es un marco de trabajo para la gestión y desarrollo de software basada en un proceso iterativo e incremental utilizado comúnmente en entornos basados en el desarrollo ágil de software.

**IDE:** Entorno de Desarrollo Integrado, llamado también IDE por sus sigla en inglés de *Integrated Development Environment*, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica.

**Licencia GPL:** Licencia Pública General de GNU pretende garantizarle la libertad de compartir y modificar software libre, para asegurar que el software es libre para todos sus usuarios. Esta Licencia Pública General se aplica a la mayor parte del software de la Free Software Foundation y a cualquier otro programa si sus autores se comprometen a utilizarla.

**IPv6:** Protocolo de Internet versión 6, es una versión del protocolo Internet Protocol (IP), diseñada para reemplazar a Internet Protocol version 4 (IPv4).

**Notación Camel:** Consiste en escribir los identificadores con la primera letra de cada palabra en mayúsculas y el resto en minúscula. Se llama notación "Camel" porque los identificadores recuerdan las jorobas de un camello.