



Universidad de las Ciencias Informáticas
Facultad 1

Título: Módulo de administración del servicio FTP para HMAST.

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

Autora:

Adrianet Cobo Fonseca

Tutores:

Ing. Gladys Marsi Peñalver Romero

Ing. Pablo Soria Acosta

Consultante:

Reidiel Castillo Arbelo

Ciudad de La Habana, Cuba

Junio 2013

*La tarea que los graduados
de la UCI tienen por
delante es grandiosa.
Espero que la cumplan,
y la cumplirán.*

Ordehiza



Módulo de administración del servicio FTP
Declaración de Autoría

Declaración de Autoría

Declaro ser la única autora 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 _____.

Adrianet Cobo Fonseca (Autora)

Ing. Gladys Marsi Peñalver Romero (Tutora)

Ing. Pablo Soria Acosta (Tutor)

Agradecimientos

*Primero que todo le quiero agradecer **a mis padres**, porque sin su apoyo no estaría donde estoy ni sería la persona que soy hoy.*

***A ti mami** por ser mi ejemplo a seguir, por darme fuerzas para salir adelante y brindarme todo tu amor y cariño, gracias por confiar en mí, te quiero mucho.*

***A ti papá** por consentirme y quererme tanto, por brindarme todo tu apoyo siempre, te quiero con la vida.*

***A mi hermana** que aunque ahora esté lejos y no la tenga aquí conmigo, ella sabe que la quiero y la extraño mucho.*

***A mi abuela** por todos sus sabios consejos, te quiero mucho y **a mi abuelo** que aunque ya no lo tengo conmigo siempre me apoyó y la última vez que pudimos hablar me dijo que estudiara mucho para que fuera alguien importante en la vida, donde quiera que estés mil gracias.*

***A mi sobrino** precioso que adoro, no hay un momento en que no piense en él y me de fuerzas para seguir adelante, siempre vas a ser mi niño lindo.*

***A toda mi familia** que de una forma u otra siempre me dio ánimos en todos los sentidos, a mis primos Liset, Ketsia, Angela María, Lilian, Jorgito, Luis Carlos, Carlos Alberto, los quiero a todos.*

*A mis tías que adoro por todos sus consejos **tía Angelia, tía Odalis, a mi tía Martha y tía María Elena.***

***A Roberto Carlos** por demostrarme que el amor es capaz de cualquier cosa, sin él no hubiese tenido fuerzas para alcanzar muchas cosas en estos cinco años, gracias por todos los momentos compartidos, gracias por soportar todas mis malcriadeces, siempre te voy a querer y nunca me voy a olvidar de ti mi tilin.*

A **Olivia** por ser mi segunda madre en estos cinco años, a **Devia y Julia** por quererme como a su nieta, yo también las adoro, y nunca me voy a olvidar de ustedes.

A mi eterna amiguis **Yinet** por todos tus consejos, eres como una hermana para mí, gracias por apoyarme siempre.

A mi compañera de cuarto y amiga **Yenny**, por soportarme todo este tiempo, regañarme, aconsejarme y apoyarme siempre.

A mi amiga de toda la vida **Laritzza**, sabes que te quiero mucho, a mis amistades del **IPVCE Enier, Orisel, Leisa y Arianna**.

A todos mis compañeros de aula que más que compañeros fueron mis amigos **Elizabeth, Carlos, Celso, Marisol, Yaima, Lisandra, Marbelis** y todos aquellos que aunque no pudieron llegar hasta aquí los llevo siempre en mi corazón.

A **Reinier**, el mejor profe y amigo que tuve en estos cinco años, gracias por todo tu apoyo y por ser como eres.

A mis tutores **Gladys y Pablo** por apoyarme y haber depositado su confianza en mí.

A **Reidiel** por ayudarme incondicionalmente y coger dolores de cabeza tratando de convencerme de mis equivocaciones.

Al equipo de **HMAST Yasiel, Yadiel, José Carlos, Maren y Nurisel** por apoyarnos en todo momento.

A todas esas personas que de una forma u otra, en el transcurso de estos cinco años, han vivido momentos tristes y alegres a mi lado. A los que me ayudaron en la realización de este trabajo. A todos gracias por haber contribuido en la formación de la persona que soy hoy.

Dedicatoria

A mi abuelo que aunque la vida lo arrancó de mi lado siempre lo llevo en mi mente y en mi corazón y estoy segura que en cualquier lugar que esté, se siente orgulloso de su nieta más chiquita.

A mis padres por brindarme todo su amor incondicional y por ser las personas más importantes de mi vida.

A mi hermana y mi sobrinito por darme todo su amor y cariño.

Resumen

Los Organismos de Administración Central del Estado en Cuba, con el fin de lograr su desarrollo y perfeccionamiento han estado realizando un proceso paulatino de migración hacia plataformas libres. Sin embargo, no contaban con una herramienta que se ajustara a sus necesidades a la hora de migrar los diferentes servicios telemáticos desde algunas versiones de Windows Server hacia plataformas libres, motivo por el cual el Departamento de Servicios Integrales en Migración Asesoría y Soporte de la Universidad de las Ciencias Informáticas, se dio a la tarea de desarrollar una Herramienta para la Migración y Administración de Servicios Telemáticos (HMAST). Esta herramienta posee funcionalidades para la administración de los usuarios, las tareas programadas, así como los servicios DHCP y SSH, sin embargo, no permite administrar el servicio FTP.

La solución que propone el presente trabajo de diploma, surge con el propósito de realizar la administración del servicio de transferencia de archivos mediante el protocolo FTP desde HMAST. La investigación estuvo enmarcada fundamentalmente en el estudio de los diferentes software dirigidos a realizar la transferencia de archivos en entornos libres, lo que dio paso a la selección de VsFTPd para el desarrollo de la aplicación. Como resultado más importante se destaca la creación de un software capaz de llevar a cabo una serie de procesos encaminados a lograr un correcto funcionamiento del servicio FTP, tales como realizar configuraciones básicas y avanzadas, tanto por tipo de usuario como para el servicio en general.

Palabras claves: administración, migración, servicios telemáticos, transferencia de archivos, VsFTPd.

Índice

Introducción.....	1
Capítulo 1: Marco teórico referencial para el módulo.....	4
1.1 Funcionamiento del Protocolo de Transferencia de Archivos (FTP).....	4
1.2 Análisis de servicios de código abierto para la transferencia de archivos	8
1.2.1 Démonio FTP de la Universidad de Washington (Wu-FTPd).....	8
1.2.2 Démonio FTP Puro (PureFTPd).....	9
1.2.3 Démonio FTP Profesional (ProFTPd).....	10
1.2.4 Démonio FTP de Línea Gris (GIFTPd).....	11
1.2.5 El Démonio FTP Muy Seguro (VsFTPd).....	12
1.3 Estudio de las herramientas que administran el servicio FTP.....	15
1.3.1 Webmin.....	15
1.3.2 Zentyal.....	16
1.4 Descripción de HMAST.....	18
1.5 Lenguajes y framework usados.....	22
1.5.1 Lenguaje de programación.....	22
1.5.2 Lenguaje de marcado.....	23
1.5.3 Lenguaje de modelado	23
1.5.4 Framework de desarrollo.....	24
1.6 Metodología y Herramientas utilizadas.....	25
1.6.1 Metodología de desarrollo.....	25
1.6.2 Herramienta de programación.....	26
1.6.3 Herramienta CASE.....	27
1.6.4 Herramienta auxiliar.....	28
1.7 Tecnología utilizada para el manejo de configuraciones en los ficheros	28
1.7.1 Augeas.....	28
1.8 Conclusiones parciales.....	29
Capítulo 2: Características del módulo.....	30
2.1 Propuesta del módulo a desarrollar.....	30
2.2 Requisitos funcionales y no funcionales.....	30
2.2.1 LRP.....	30
2.3 Historias de Usuarios (HU).....	37
2.4 Arquitectura del software.....	41
2.4.1 Estilos arquitectónicos.....	41
2.5 Diagrama de Paquetes.....	42
2.6 Patrones de diseño.....	45
2.6.1 Patrones GRASP.....	45
2.6.2 Patrones GOF.....	46
2.7 Conclusiones parciales.....	47
Capítulo 3: Desarrollo de la solución propuesta.....	48
3.1 Plan de Iteraciones.....	48
3.2 Tareas de ingeniería.....	48
3.3 Estándares de Codificación.....	50
3.4 Pruebas de software.....	51
3.4.1 Diseño de casos de prueba.....	52

3.4.2 Pruebas realizadas por el cliente.....	55
3.4.3 Resultados de las pruebas.....	55
3.5 Conclusiones parciales.....	56
Conclusiones.....	57
Recomendaciones.....	58
Referencias bibliográficas.....	59
Bibliografía.....	62
Anexo 1: Historias de Usuario.....	66
Anexo 2: Tareas de Ingeniería.....	90
Glosario de términos.....	101

Índice de Figuras

Figura 1: Funcionamiento del protocolo FTP.....	5
Figura 2: Arquitectura del sistema.....	19
Figura 3: Arquitectura del módulo para la administración del servicio FTP.....	42
Figura 4: Diagrama de paquetes para la HU Realizar configuraciones para usuarios locales.....	44
Figura 5: Enumeración del código.....	52
Figura 6: Grafo de flujo.....	53

Índice de Tablas

Tabla 1: Comparación entre servidores FTP.....	14
Tabla 2: Lista de Reserva del Producto (LRP).....	37
Tabla 3: Listado de las HU.....	40
Tabla 4: Listado de Tareas de Ingeniería.....	50
Tabla 5: Conjunto de caminos básicos.....	53
Tabla 6: Caso de prueba para el camino básico 1.....	54
Tabla 7: Caso de prueba para el camino básico 2.....	55

Introducción

La introducción y uso de las Tecnologías de la Información y las Comunicaciones (TIC), ha sido un salto vertiginoso en el desarrollo científico técnico a escala mundial. Cuba ha incorporado el uso de estas tecnologías, y es gracias a la voluntad política de la Revolución Cubana y a la clara visión del Comandante Fidel Castro Ruz, promotor incansable del uso de las TIC, que en el país cobra gran importancia el uso social y colectivo de estas tecnologías [1].

Con el fin de asegurar la soberanía tecnológica, impulsar la innovación nacional y facilitar la inclusión digital, actualmente Cuba se encuentra inmerso en un proceso paulatino de migración total y masiva hacia Software Libre y aplicaciones de código abierto, pues es real el hecho de que constituye una necesidad, debido a lo que representa para la sociedad cubana en los aspectos políticos, económicos, sociales y tecnológicos. Persiguiendo el propósito de apoyar la migración en el país, en la Universidad de las Ciencias Informáticas (UCI), fue creado el Centro de Software Libre (CESOL), el cual cuenta con el Departamento de Servicios Integrales en Migración, Asesoría y Soporte (SIMAYS). La misión del mismo va dirigida a tres líneas fundamentales: el desarrollo de estrategias, guías y planes de migración; las herramientas de soporte al proceso de migración y la migración de los servicios telemáticos y aplicaciones de escritorio.

Como parte de la estrategia de migración, desde el año 2011 SIMAYS se dio a la tarea de desarrollar la Herramienta de Migración y Administración de Servicios Telemáticos (HMAST), la cual surge debido a la necesidad que tenían algunos de los Órganos de la Administración Central del Estado (OACE) de contar con una herramienta que se ajustara a sus necesidades a la hora de migrar desde diferentes versiones de Windows Server hacia plataformas libres, los servicios telemáticos que estos brindan, así como la administración de los mismos. Esta herramienta está compuesta por un sistema base que permite administrar los servidores, el cual posee las funcionalidades necesarias para la administración de usuarios, tareas programadas y servicios.

A pesar de la existencia de HMAST, todavía resulta un reto llevar a cabo el proceso de migración y administración de los servicios telemáticos de forma exitosa haciendo uso de la misma, debido a que está en su primera versión y solo tiene desarrollados dos módulos: uno para administrar las conexiones mediante el protocolo SSH y otro para realizar la administración del servicio DHCP. Un factor importante a tener en cuenta es que entre los servicios telemáticos, está enmarcado el que implementa el Protocolo de

Transferencia de Archivos (FTP), a través del cual se puede realizar el intercambio de archivos entre equipos remotos de forma ágil y segura. Además brinda la posibilidad de mantener la información digitalizada centralizada en un espacio único. En correspondencia con las ventajas que ofrece el servicio FTP para las instituciones que lo utilizan, se hace necesario que HMAST permita la administración del mismo.

Para dar solución a la problemática anteriormente planteada se define el siguiente **problema científico**:
¿Cómo realizar la administración del servicio FTP desde la herramienta HMAST?

Se establece como **objetivo general** desarrollar un módulo para HMAST que permita la administración del servicio FTP. Con el propósito de lograr el objetivo propuesto se han definido los siguientes **objetivos específicos**:

- Sistematizar sobre los software de código abierto que brindan el servicio FTP.
- Sistematizar sobre las herramientas de código abierto para la administración del servicio FTP.
- Analizar y diseñar el módulo para la administración del servicio FTP.
- Implementar las funcionalidades del módulo definido.
- Probar las funcionalidades del módulo obtenido.

Para lograr los objetivos propuestos se precisan las siguientes **tareas investigativas**:

- Revisión de la literatura especializada sobre los software de código abierto que brindan el servicio FTP.
- Estudio de herramientas de código abierto que realizan la administración del servicio FTP.
- Especificación de los requisitos funcionales y no funcionales para el módulo a desarrollar.
- Definición de la arquitectura del módulo y patrones de diseño a utilizar.
- Desarrollo de las Historias de Usuario.
- Diseño y ejecución de los casos de pruebas.

Persiguiendo el objetivo de establecer los límites de la investigación, el **objeto de estudio** se define como la administración del servicio FTP en entornos libres. Quedando enmarcado el **campo de acción** en la

administración del servicio FTP en entornos libres para HMAST.

El presente trabajo de diploma plantea la siguiente **idea a defender**: el desarrollo de un módulo para la herramienta HMAST facilitará el proceso de administración del servicio FTP.

Para el desarrollo de la investigación se hace uso del **método Analítico - Sintético** el cual se aplica para investigar sobre los diferentes servidores de FTP existentes, teniendo en cuenta las características de cada uno de ellos. De esta forma, permite arribar a conclusiones que apoyen la selección de aquel servidor que más se ajuste al objetivo principal de la investigación.

También se utilizó el **Análisis Histórico – Lógico** el cual permitió determinar los antecedentes históricos relacionados con las herramientas para la administración del servicio FTP.

El presente trabajo tiene estructurado su contenido en tres capítulos, los cuales se definen a continuación:

Capítulo 1- Marco teórico referencial para el módulo: Se realiza una descripción del funcionamiento del protocolo FTP para luego centrarse en el estado del arte del objeto de estudio, abordando sobre los servicios que permiten realizar la transferencia de archivos en entornos libres, de los cuales se resaltan sus principales características, funcionalidades y deficiencias. Además, se estudia la forma en que las herramientas para la administración de servidores como Zentyal y Webmin, administran el servicio FTP. Una vez realizado este análisis se describen brevemente las principales características de HMAST, sistema base del módulo a desarrollar. Por último se fundamentan las tecnologías, lenguajes de programación, metodología y herramientas utilizadas en el desarrollo de la aplicación.

Capítulo 2- Características del módulo: Se presenta la propuesta de solución al problema planteado, quedando definidos los requisitos funcionales y no funcionales del módulo. También se describen las Historias de Usuario (HU) que quedaron definidas. Por último, en este capítulo se detalla la arquitectura del módulo y elementos del diseño, como son los patrones utilizados y la descripción del Diagrama de paquetes.

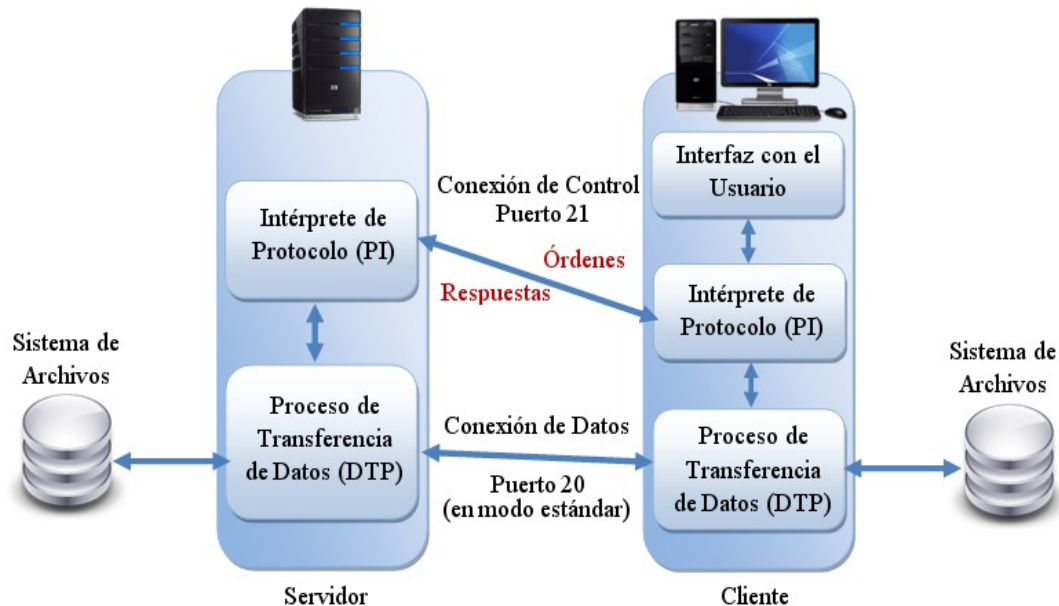
Capítulo 3- Desarrollo de la solución propuesta: Se elabora el Plan de Iteraciones, el cual define cuáles funcionalidades serán desarrolladas en cada etapa del proceso de implementación. Además, se detallan las tareas de ingeniería más importantes realizadas, las cuales contribuyeron al desarrollo del módulo. Por último, se aborda sobre las pruebas realizadas a las funcionalidades, con el fin de comprobar el correcto funcionamiento de las mismas y su correspondencia con los requisitos planteados.

Capítulo 1: Marco teórico referencial para el módulo

El presente capítulo se centra principalmente en realizar un estudio del protocolo FTP, con el propósito de entender el funcionamiento del mismo. Además, se realiza un análisis de los software o demonios que brindan el servicio FTP en sistemas operativos GNU/Linux, quedando detalladas sus características, resaltando las ventajas y desventajas de cada uno de ellos, con el objetivo de seleccionar el idóneo para realizar la administración de dicho servicio. Se analizan también, cómo algunas herramientas para administrar servidores realizan la administración del servicio FTP, con el fin de enriquecer la solución. Un factor importante a tener en cuenta en la concepción del módulo, es el funcionamiento y las características del sistema al cual va a ser integrado, por esta razón en el presente capítulo se describen aspectos relevantes del mismo. Por último, quedan detallados qué lenguajes de programación, metodología, herramientas y tecnologías serán utilizadas para desarrollar el software en cuestión, las cuales se corresponden con las que quedaron definidas en estudios iniciales referentes al desarrollo de HMAST, aspectos que se encuentran especificados en el documento Plantilla de Concepción del Sistema HMAST, recogido en el expediente de proyecto de dicho sistema.

1.1 Funcionamiento del Protocolo de Transferencia de Archivos (FTP)

El protocolo FTP permite definir la manera en que los datos deben ser transferidos a través de una red TCP/IP. Funciona según el modelo Cliente/Servidor, donde el componente Servidor se denomina demonio FTP. El objetivo principal de este servicio es permitir que equipos remotos puedan compartir información, posibilitando de esta forma, la independencia entre los sistemas de archivo del equipo cliente y los del servidor. Durante una conexión FTP, se encuentran abiertos dos canales de transmisión: un canal de comandos (canal de control) y el de datos como se muestra en la Figura 1 [2].



Por lo tanto, el cliente y el servidor cuentan con dos procesos que permiten la administración de estos dos tipos de información:

- El Proceso de Transferencia de Datos (DTP), es el proceso encargado de establecer la conexión y de administrar el canal de datos. El DTP del lado del servidor se denomina Servidor de DTP y el DTP del lado del cliente se denomina Usuario de DTP.
- El Intérprete de Protocolo (PI) interpreta el protocolo y permite que el DTP pueda ser controlado mediante los comandos recibidos a través del canal de control.

Estos procesos son diferentes en el cliente y en el servidor:

- El Servidor PI es responsable de escuchar los comandos que provienen de un Usuario PI a través del canal de control en un puerto de datos. Además, se encarga de establecer la conexión para el canal de control, de recibir los comandos FTP del Usuario PI a través de este, de emitir una respuesta y de ejecutar el Servidor de DTP.
- El Usuario PI es responsable de establecer la conexión con el servidor FTP, de enviar los

comandos FTP, de recibir respuestas del Servidor PI y de controlar al Usuario de DTP, si fuera necesario.

De forma general, cuando un cliente FTP se conecta con un servidor FTP, el Usuario PI inicia la conexión con el servidor de acuerdo con el protocolo Telnet. El cliente envía comandos FTP al servidor, este los interpreta, ejecuta su DTP y después envía una respuesta. Una vez que se establece la conexión, el Servidor PI proporciona el puerto por el cual se enviarán los datos al Usuario DTP y este escucha el puerto especificado para los datos provenientes del servidor.

Tipos de acceso al servicio FTP

➤ Acceso anónimo:

Los servidores FTP anónimos ofrecen sus servicios libremente a todos los usuarios, permiten acceder a sus archivos sin necesidad de tener una cuenta que los identifique. Es la manera más cómoda fuera del servicio web de permitir que todo el mundo tenga acceso a cierta información sin que para ello el administrador de un sistema tenga que crear una cuenta para cada usuario.

Sin embargo, al acceder de esta forma se tendrán menos privilegios que accediendo como un usuario normal. Lo recomendable es que al acceder de forma anónima al servidor solo se puedan leer y copiar los archivos que sean públicos.

➤ Acceso de usuario:

Si se desea tener privilegios de acceso a cualquier parte del sistema de archivos del servidor FTP, de modificación de datos existentes, y de posibilidad de subir archivos propios, generalmente se suele realizar mediante una cuenta de usuario. En el servidor se guarda la información de las distintas cuentas que pueden acceder a él, de manera que para iniciar una sesión FTP se debe introducir un nombre de usuario y una contraseña que identifique al usuario unívocamente.

➤ Acceso de invitado:

El acceso sin restricciones al servidor que proporcionan las cuentas de usuario implica problemas de seguridad, lo que ha dado lugar a un tercer tipo de acceso FTP denominado invitado (*guest*), que se puede contemplar como una mezcla de los dos anteriores. La idea de este mecanismo es la siguiente: se trata de permitir que cada usuario se conecte a la máquina mediante su usuario y su contraseña, pero evitando que tenga acceso a partes del sistema de archivos que no necesita para realizar su trabajo, de

esta forma accederá a un entorno restringido, algo muy similar a lo que sucede en los accesos anónimos, pero con más privilegios.

Modos de conexión del cliente

FTP permite dos modos de conexión por parte del cliente, estos se denominan activo y pasivo, en los cuales el cliente establece una conexión con el servidor a través del canal de control mediante el puerto 21.

➤ **Modo activo:**

En modo activo, el servidor siempre crea el canal de datos en su puerto 20, mientras que en el lado del cliente el canal de datos se asocia a un puerto aleatorio mayor que el 1024. Para ello, el cliente manda un comando PORT¹ al servidor por el canal de control indicándole ese número de puerto, de manera que el servidor pueda abrirle una conexión de datos por donde se transferirán los archivos, en el puerto especificado. Esto trae como consecuencia un grave problema de seguridad, y es que la máquina cliente debe estar dispuesta a aceptar cualquier conexión de entrada en un puerto superior al 1024, con los problemas que ello implica si se tiene el equipo conectado a una red insegura como Internet. De hecho, los cortafuegos que se instalen en el equipo para evitar ataques seguramente rechazarán esas conexiones aleatorias. Para solucionar esto se desarrolló el modo pasivo.

➤ **Modo pasivo:**

Cuando el cliente envía un comando PASV² sobre el canal de control, el servidor FTP le indica por dicho canal, el puerto (mayor a 1023) al que debe conectarse el cliente. El cliente inicia una conexión desde el puerto siguiente al puerto de control hacia el puerto del servidor especificado anteriormente.

Antes de cada nueva transferencia tanto en el modo activo como en el pasivo, el cliente debe enviar otra vez un comando de control (PORT o PASV, según el modo en el que haya conectado), y el servidor recibirá esa conexión de datos en un nuevo puerto aleatorio (si está en modo pasivo) o por el puerto 20 (si está en modo activo).

Funcionalidades estándares del servicio

- Permitir o no el acceso de usuarios locales al servidor.

1 **PORT** cadena de caracteres que permite especificar el número de puerto utilizado.

2 **PASV** Comando que permite indicar al servidor de FTP que permanezca a la espera de una conexión en un puerto específico elegido aleatoriamente entre los puertos disponibles. La respuesta a este comando es la dirección IP del equipo y el puerto.

- Permitir o no el acceso de usuarios anónimos al servidor.
- Establecer permisos por tipos de usuarios.
- Restringir el ancho de banda para la transferencia de datos.
- Restringir la concurrencia en el servidor teniendo en cuenta la cantidad de clientes e IP conectados de forma simultánea.
- Restringir la cantidad de intentos fallidos por sesión.
- Brindar soporte opcional para una capa de cifrado SSL/TLS.
- Gestionar registros en el servidor.

1.2 Análisis de servicios de código abierto para la transferencia de archivos

En la actualidad existe una gran variedad de servicios dirigidos a realizar la transferencia de archivos en sistemas operativos GNU/Linux. A continuación se especifican las funcionalidades, así como ventajas y desventajas de cada uno de los que son analizados, con el propósito de seleccionar el que será utilizado para administrar el servicio FTP en la herramienta HMAST.

1.2.1 Demonio FTP de la Universidad de Washington (Wu-FTPd)

Es uno de los servicios para la transferencia de ficheros de dominio público y resulta, en cierta medida, fácil de configurar. Se encuentra bajo la licencia GPL y soporta la conexión como usuario real, si está definido en el fichero de claves del sistema, o como usuario anónimo. Pueden definirse además, varias clases de usuarios, dependiendo de la dirección IP o del nombre de la máquina cliente y tipo de usuario conectado; los tipos de usuarios definidos son: unix (usuarios normales del sistema), invitado (usuarios unix que ha sido clasificados como invitados) y por último los anónimos. Wu-FTPd es capaz de establecer límites de conexión y restricciones de acceso para cada clase de usuarios. Permite el uso de ficheros históricos para registrar los accesos al servidor. Utiliza varios tipos de mandatos como son borrar, modificar y cambiar permisos. Incluye también, mensajes de bienvenida e índices para cada directorio [3].

Cuenta con las siguientes características operativas para los usuarios anónimos:

- Directorio público utilizado como almacén de ficheros.
- El usuario público solamente puede ver el árbol público.

- Puede crearse un directorio para añadir archivos, pero es poco recomendable.

El fichero principal para configurar el servidor está ubicado en `/etc/ftpaccess`, el cual consta de cinco grupos de directivas, las cuales son: control de acceso, información automática para clientes, configuración de entrada, permisos y controles de tipo de archivo y directorio. Wu-FTPd cuenta con cuatro ficheros más de configuración dentro del directorio `/etc`, como son, `ftpusers`, el cual contiene la lista de usuarios que no podrán usar el servicio FTP, `ftpgroups`, con el formato del archivo para acceso a grupos, `ftphosts` que permite o deniega el acceso a ciertas cuentas desde las máquinas especificadas, y por último `ftpconversions`, el cual contiene la base de datos para la conversión de tipos de archivos.

Las versiones Wu-FTPd entre la 2.5.0 y la 2.6.2, incluyendo a ambas, contienen un desbordamiento de *buffer*³, lo que le permite a un atacante remoto ejecutar código arbitrario en el servidor y obtener privilegios de administrador [4]. Esto trae como consecuencia que dicho software no sea confiable en términos de seguridad.

La autora considera que no es efectivo utilizar Wu-FTPd para la administración del servicio FTP en la herramienta HMAST, ya que a pesar de proporcionar un rendimiento aceptable y ser generalmente un buen producto, presenta un historial de seguridad pobre, y debido a los constantes ataques a servidores de este tipo, dio paso a la creación de otros productos que fueran capaces de erradicar las deficiencias de WuFTPd. Otro aspecto importante a tener en cuenta es que contiene cinco archivos para ser configurados, lo que hace que en comparación con otros servidores, su configuración se haga más tediosa. Además, carece de funcionalidades como establecer limitaciones de ancho de banda, soporte para conexiones SSL y conexiones pasivas, así como autenticación de usuarios con MySQL, PostgreSQL y LDAP.

1.2.2 Demonio FTP Puro (PureFTPd)

Es un equipamiento lógico para servidor FTP licenciado bajo GPL. A pesar de su escasa popularidad, se distingue de otros proyectos porque ha tenido como objetivo mantener el servicio con poco consumo de recursos. Es fácil de instalar y configurar y además, cumple con los estándares para el protocolo FTP (permitir autenticación de usuarios locales y anónimos, generación de registros y configuración de permisos por tipos de usuarios) [5].

3 **desbordamiento de buffer** es un error de software que se produce cuando un programa no controla adecuadamente la cantidad de datos que se copian sobre un área de memoria reservada a tal efecto (buffer).

Sus características más notables son:

- Permitir usuarios virtuales.
- Gestión del ancho de banda y de espacio en disco por usuario.
- Cada usuario puede ser restringido de forma individual a su directorio de inicio o no.
- Las descargas pueden ser denegadas si la carga del sistema es demasiada.
- Estadísticas en tiempo real en formato HTML o XML.
- Autenticación de usuarios con MySQL, PostgreSQL, LDAP.
- Compatible con IPv6.
- La transmisión de contraseñas en texto plano y los comandos se pueden evitar, ya que tiene soporte opcional para una capa de cifrado SSL/TLS mediante la biblioteca OpenSSL.

Cuenta con dos ficheros de configuración ubicados en el directorio /etc: pure-ftpd.conf para la configuración del servidor y pureftpd.passwd para la gestión de usuarios. Una de las ventajas de PureFTPd es que permite definir a través de qué rango de puertos se tramitarán las conexiones pasivas, de ese modo se conectan la mayoría de los clientes FTP. El servidor puede estar “escuchando” un puerto a la espera de recibir conexiones (por ejemplo el 21) y luego, deriva las transferencias hacia otros puertos. Este hecho, en otros servidores FTP puede provocar muchos problemas al administrador del sistema, ya que es muy complicado saber a priori a través de qué puertos se realizarán las transferencias de datos y por lo tanto, a la hora de establecer las políticas de acceso al servidor web en el cortafuego, este hecho se revela como un problema [6].

En la versión 1.0.36 de PureFTPd, disponible en los repositorios de Ubuntu, es necesario crear un archivo por cada parámetro de configuración, donde el contenido se corresponda con el valor del mismo. Este proceso resulta incómodo, al ser comparado con otros servidores alternativos. La autora considera que a pesar de contar con esta deficiencia, brinda una serie de funcionalidades que lo hacen un potente candidato para realizar la administración del servicio FTP.

1.2.3 Démonio FTP Profesional (ProFTPd)

Es flexible, con cierto nivel de seguridad y fácil de configurar. Cuando el proyecto comenzó, el servidor

más utilizado era Wu-FTPd, pero los ataques a servidores de este tipo cada vez se hacían más frecuentes, por lo que pronto quedó claro que un rediseño completo era necesario para aplicar la configuración y las características deseadas, surgiendo de esta forma ProFTPd.

Entre todas las características que posee las más relevantes son que cuenta con un solo fichero de configuración ubicado en `/etc/proftpd.conf`, con una sintaxis parecida a la del servidor web Apache, posibilita configurar múltiples servidores virtuales, y se pueden tener varios servidores brindando servicio de FTP anónimo. Es modular, o sea, permite extender su funcionalidad ampliamente, está licenciado bajo GPL y es compatible con IPv6. Este servidor puede ser utilizado en varios sistemas operativos. Está diseñado para funcionar tanto como un servidor independiente o desde `inetd/xinetd`, dependiendo de la carga del sistema. Algunos de sus principales módulos son para bases de datos SQL, servidores LDAP y cifrado SSL/TLS [7].

Sin embargo, ProFTPd posee una serie de vulnerabilidades registradas en CVE⁴, directorio donde son publicadas las vulnerabilidades de seguridad en diferentes sistemas GNU/Linux. Una de las más significativas por el impacto que produce es una conocida como *Use-After-Free*, que podría permitir a usuarios maliciosos ejecutar código arbitrario en el servidor [8].

Aunque este hecho solo ha tenido grandes repercusiones en las versiones anteriores a la 1.3.3g ó 1.3.4, la autora considera que este tipo de servidor no es el idóneo para ser utilizado en la administración del servicio FTP desde HMAST. Además, es considerado uno de los servidores FTP con más vulnerabilidades listadas en CVE y sus propios desarrolladores han dado parte de puertas traseras descubiertas en este software.

1.2.4 Demonio FTP de Línea Gris (GIFTPd)

El nombre de este software surge debido a que su primer desarrollador lo nombró *GreyLine* del español línea gris, el desarrollo de GIFTPd comenzó aproximadamente a principios de 1998. Es un equipamiento lógico para servidor FTP en sistemas operativos derivados de Unix. Es de código abierto, como la mayoría de los servidores FTP disponibles para dichos sistemas operativos. Actualmente, el código fuente está disponible para GNU/Linux, FreeBSD, OpenBSD y Mac OS. Funciona sobre un ambiente de *chroot*⁵ y cuenta con su propia base de datos de usuarios [9].

4 **CVE** acrónimo de *Common Vulnerabilities and Exposures*, y es un diccionario para el conocimiento público de las vulnerabilidades de seguridad de información y exposiciones de los software.

5 **chroot** en los sistemas operativos derivados de Unix, es una operación que invoca un proceso, cambiando para este y sus hijos el directorio raíz del sistema.

GIFTPd soporta las siguientes funciones:

- Las limitaciones de ancho de banda.
- Las relaciones de descargar/subir.
- La gestión de usuarios por medio de un cliente FTP.
- Varios atributos de usuarios, como siteop, operario del sitio FTP y gadmin, administrador de grupos.
- La facultad de comprobar si un archivo ya ha sido subido.
- Mostrar estadísticas de varios tipos.
- Cuenta con gl_spy, una herramienta para observar el comportamiento de los usuarios.
- La compatibilidad con *bouncers*, servidores que esconden la dirección IP del servidor FTP.
- Ayuda de la encriptación con la integración de TLS/SSL.

GIFTPd sigue en su mayor parte el modelo de la "seguridad a través de la oscuridad", término que se asocia a distintos aspectos como son mantener en secreto el código fuente del software, así como algoritmos y protocolos utilizados, además se pueden adoptar políticas de no revelación pública de la información sobre vulnerabilidades. Un sistema que se apoya en la seguridad por ocultación, como también se conoce, puede tener vulnerabilidades teóricas o prácticas, pero sus propietarios o diseñadores creen que sus puntos débiles, debido al secreto que se mantiene sobre las debilidades del sistema, son muy difíciles de encontrar, y por tanto los atacantes tienen muy pocas probabilidades de descubrirlos. Sin embargo, esta teoría no es totalmente confiable.

La autora considera que GIFTPd no se corresponde completamente con el objetivo principal de la investigación, ya que al usarse en entornos más privados no se ajusta al marco empresarial, y este es precisamente uno de los contextos para los cuales está destinada la herramienta a la cual se le va a agregar el módulo de administración de FTP. Además no posee un alto nivel de seguridad, teniendo en cuenta los argumentos anteriormente planteados.

1.2.5 El Demonio FTP Muy Seguro (VsFTPd)

Es un servicio dirigido a realizar la transferencia de archivos para Linux y otros sistemas operativos Unix.

Se distingue principalmente porque sus valores predeterminados son seguros, y por su sencillez en la configuración, comparado con otras alternativas como ProFTPd, y Wu-FTPd. Es rápido y distribuido bajo la licencia GPL.

Posee un sistema de encriptación de archivos basado en SSL y permite configurar direcciones de IP virtuales con gran facilidad. VsFTPd cuenta con soporte para IPv6 y capacidad de crear usuarios virtuales. Instalarlo es sumamente sencillo y contiene solamente un fichero de configuración localizado en `/etc/vsftpd/vsftpd.conf`. Brinda soporte para el control de ancho de banda y para la autenticación con usuarios almacenados en un servidor LDAP. Se ejecuta de manera independiente o a través de `inetd`. Provee además, una poderosa configuración para cada tipo de usuario. Permite activar conexiones pasivas, en caso de que se activen, brinda la posibilidad de establecer el rango de puertos de escucha. Es capaz de gestionar la cantidad máxima de conexiones simultáneas tanto por IP como por clientes, así como la cantidad máxima permitida de intentos de acceso fallidos por sesión. Permite la generación de registros por cada operación que se realice en servidor, ya sea subir o descargar algún archivo.

A continuación son citadas algunas críticas positivas sobre VsFTPd [10]:

- El equipo SAC de SANS recomienda VsFTPd como el servidor FTP seguro preferido: *“For those of you looking for a secure FTP daemon alternative, the SAC team recommends vsftpd”*.
- IBM recomienda VsFTPd en sus artículos *“Securing Linux Servers for Service Providers”*. Esto es la parte superior de una sección que se titula *“Recommended FTP servers”*.
- Red Hat elogia el funcionamiento y la dimensión de VsFTPd : *“Individual servers handled more than 2,500 concurrent downloads”...“The other change was to use a very lightweight FTP daemon, vsftpd, designed for the demands place on a Server under this level of load”*.

Según el sitio oficial de este servicio, muchas organizaciones importantes del software libre lo usan, ejemplos de estos sitios son `ftp.debian.org`, `ftp.suse.com`, `ftp.redhat.com`, `ftp.openbsd.org`, `ftp.freebsd.org`, `ftp.gnu.org`, `ftp.gnome.org`, `ftp.kde.org`, `ftp.kernel.org` y `ftp.gimp.org`. Además, VsFTPd es el demonio FTP por defecto en Ubuntu, CentOS, NimblerX y RHEL, distribuciones de Linux reconocidas a nivel mundial.

Un servidor FTP con las características de VsFTPd, es sin lugar a dudas una opción apropiada para la administración del servicio FTP. Reconocido además, por empresas como las anteriormente mencionadas, las cuales se han ganado un prestigio incalculable mundialmente en el ámbito del desarrollo de software,

destacándose por su incansable trabajo, guiado a impulsar el avance del software libre y aplicaciones de código abierto.

A continuación se muestra una tabla donde quedan reflejados aspectos relevantes que caracterizan a los servicios de este tipo que han sido descritos, con el fin de comparar y arribar a conclusiones acertadas que apoyen la selección del que será utilizado para dar solución al problema planteado.

Características	VsFTPd	GIFTPd	PureFTPd	WuFTPd	ProFTPd
Licencia.	GPL	GPL	GPL	GPL	GPL
Seguridad.	Alta	Media	Alta	Baja	Baja
Facilidad de configuración.	Alta	Alta	Media	Baja	Alta
Autenticación y configuración de permisos para usuarios locales y anónimos.	x	x	x	x	x
Rango de puertos para conexiones pasivas.	x		x		x
Generación de registros.	x	x	x	x	x
Limitaciones de ancho de banda.	x	x	x		x
Enjaulado de usuarios.	x	x	x	x	x

Tabla 1: Comparación entre servidores FTP.

Después de haber analizado cada una de las características y deficiencias de los diferentes servidores FTP para entornos libres, centradas principalmente en la seguridad, complejidad de configuración, opciones de configuración avanzadas por tipo de usuarios, licencia bajo la cual están patentados, generación de registros, así como vulnerabilidades que presentan, la autora considera que los servidores que más se ajustan a las necesidades de la investigación son PureFTPd y VsFTPd. Sin embargo, al analizar las opiniones de empresas tan prestigiosas como IBM, Red Hat, el equipo SAC de SANS y teniendo en cuenta las ventajas y funcionalidades que este servidor ofrece, se llega a la conclusión que el que más se ajusta al objetivo general de la investigación es el VsFTPd, debido a que ofrece seguridad, facilidad de instalación y configuración. Es válido destacar que aunque PureFTPd también posee la mayoría de estas características, VsFTPd proporciona mayor facilidad de configuración en parámetros como permisos para usuarios anónimos, máscara de archivos o directorios cuando son creados, además, a diferencia de PureFTPd, no es necesario crear un directorio para cada parámetro que se desee configurar. Al analizar los argumentos anteriormente expuestos se concreta la selección de VSFTPd como

el servidor que se utilizará para la administración del servicio FTP en la herramienta HMAST.

1.3 Estudio de las herramientas que administran el servicio FTP

Actualmente existe una gran variedad de aplicaciones que permiten la administración de servicios telemáticos. Siguiendo el hilo de la investigación, es necesario centrar el estudio en cómo estas herramientas administran el servicio FTP, con el fin de acatar aquellas características que puedan ser útiles para el desarrollo del módulo en cuestión y desechar los aspectos negativos, dos factores que dan paso al logro de un producto con la calidad requerida.

Para la concepción de HMAST se realizó un análisis de las principales herramientas de este tipo como son Zentyal y Webmin, partiendo de dicho estudio, a continuación son descritas las herramientas mencionadas anteriormente, haciendo énfasis principalmente en el proceso de administración del servicio FTP.

1.3.1 Webmin

Webmin es una herramienta de configuración de sistemas, accesible vía web para OpenSolaris, GNU/Linux y otros sistemas Unix, la cual permite configurar aspectos internos de muchos sistemas operativos, como gestión de usuarios, cuotas de espacio, servicios, archivos de configuración y apagado del equipo. Está escrita en Perl y por defecto se comunica mediante el protocolo TCP a través del puerto 10000 [11].

Esta herramienta permite la configuración de la mayoría de los servicios telemáticos mediante módulos entre los cuales se encuentran: Apache WebServer, BIND DNS Server, CVS Server, DHCP Server, FTP, Proxy, Jabber, entre otros. En el caso de la administración del servicio FTP, Webmin, en su versión 1.6 cuenta entre sus módulos estándar con los servidores ProFTPd y WuFTPd, realizando la administración de los mismos mediante interfaces poco amigables, ya que el orden y los enunciados de las configuraciones carecen de organización y no poseen un formato adecuado. Además, no realiza una correcta validación de los parámetros que se introducen en algunos de los campos y en casi todas las configuraciones aparece una opción que permite establecer el valor de la directiva, sin embargo, no se especifica qué valor es.

En el caso de la configuración mediante Webmin de VsFTPd, servidor seleccionado en correspondencia con la investigación realizada para la administración de FTP en HMAST, es necesario incorporarle un módulo que define el Webmin para este propósito, ya que no lo contiene de forma predeterminada, a

diferencia de los anteriormente mencionados. Una vez instalado el módulo, Webmin maneja la configuración de forma organizada en comparación a cómo lo hace para los servidores FTP que contiene. También muestra el estado del fichero de configuración y pone a disposición un manual donde se especifican cada uno de los parámetros a configurar, detallando su funcionalidad y los valores que pueden tomar. Es importante destacar que esta herramienta permite realizar la mayoría de las funcionalidades que ofrece este servidor FTP, como por ejemplo la gestión de permisos por tipos de usuarios, incluso para usuarios virtuales, configuraciones generales como la máscara de los archivos creados, tiempo máximo de conexiones inactivas, máxima cantidad de IP y usuarios conectados de forma simultánea, permite también gestionar los registros del sistema, así como los certificados SSL y brinda soporte para este tipo de conexiones. Además, en algunos de los campos de entrada recomienda qué valores puede tomar, sin embargo, estas entradas no son validadas de forma correcta.

En general, Webmin administra el servicio FTP mediante VsFTPD, a través de una interfaz intuitiva, no obstante, posee algunas deficiencias que hacen que no se exploten todas las funcionalidades que ofrece este servidor, además al no validar los datos de forma correcta, puede provocar fallos al brindar el servicio.

1.3.2 Zentyal

Zentyal es una alternativa en código abierto a los productos de Microsoft como Windows Small Business Server, Windows Server y Microsoft Exchange, para infraestructura TIC en las Pequeñas y Medianas Empresas (PYMES). El desarrollo de Zentyal se inició en el año 2004 con el nombre de *eBox Platform* y actualmente es una solución consolidada de reconocido prestigio que integra más de 30 herramientas de código abierto para la administración de sistemas y redes en una sola tecnología. Zentyal está incluido en *Ubuntu* desde el año 2007, y a partir del año 2012 las ediciones comerciales están oficialmente respaldadas por Canonical, empresa encargada del desarrollo y comercialización de Ubuntu. En la actualidad Zentyal tiene más de 1.000 descargas diarias y dispone de una comunidad activa de miles de miembros [12].

Esta herramienta realiza la administración del servicio FTP mediante el servidor VsFTPD, a través de una interfaz intuitiva. Sin embargo, realiza este proceso de forma muy simple, ya que solamente contempla las opciones de otorgar acceso remoto a un directorio público y/o a los directorios personales de los usuarios del sistema. En el caso del acceso anónimo, brinda tres configuraciones posibles para el directorio público:

- Desactivado: No se permite el acceso a usuarios anónimos.
- Solo lectura: Se puede acceder al directorio con un cliente de FTP, pero únicamente se puede listar los ficheros y descargarlos. Esta configuración es adecuada para poner a disposición de cualquiera el contenido para su descarga.
- Lectura y escritura: Se puede acceder al directorio con un cliente de FTP y cualquiera puede añadir, modificar, descargar y borrar ficheros en este directorio. No se recomienda esta configuración a menos de estar muy seguro de lo que se hace.

Otro parámetro de configuración es para los directorios personales el cual permite que los usuarios de Zentyal accedan directamente a su directorio personal. Además se contempla la opción de brindar soporte SSL, donde se puede forzar el acceso seguro utilizando SSL, hacerlo opcional o deshabilitarlo. Se debe tener en cuenta que antes de poner en funcionamiento el servidor FTP, es necesario comprobar que las reglas del cortafuegos abran los puertos necesarios para este servicio.

De forma general, Zentyal realiza la administración del servicio FTP de forma rígida, ya que en su interfaz contempla solamente configuraciones básicas, realizando otras predeterminadamente, sin posibilidad de ser modificadas, a no ser, mediante el acceso directo al fichero ubicado en `/etc/vsftpd.conf`. Algunos de estos parámetros son la activación de registros, la máscara de los archivos cuando son creados, tanto para usuarios locales como anónimos, el texto de bienvenida y la activación de los mensajes de directorio. Sin embargo, no tiene en cuenta directivas como la cantidad máxima de usuarios e IP conectados de forma simultánea, el tiempo máximo de conexiones inactivas, configuración de usuarios virtuales y máxima velocidad de transferencia de datos para usuarios locales y anónimos. Por otra parte, no ofrece un manual de ayuda detallado para realizar las configuraciones, solamente cuenta con una breve explicación sobre el acceso y los permisos de los usuarios anónimos.

El resultado del estudio realizado aportó aspectos importantes a tener en cuenta en la concepción del módulo, entre los que se destaca la necesidad de explotar todas las ventajas y funciones que brinda VsFTPd de forma semejante a como lo maneja Webmin, debido a que Zentyal realiza la administración de FTP de forma muy básica, obviando elementos fundamentales de este servicio como son las configuraciones por tipo de usuarios, el límite de concurrencia, el ancho de banda de las transferencias de datos, así como otras configuraciones generales, que a pesar de realizarlas de forma predeterminada, no le brinda la posibilidad al usuario de personalizarlas. Otro factor importante a considerar es la correcta

validación de los parámetros de configuración, ya que Webmin no contempla este aspecto, el cual puede tener un impacto negativo en el funcionamiento del servicio.

1.4 Descripción de HMAST

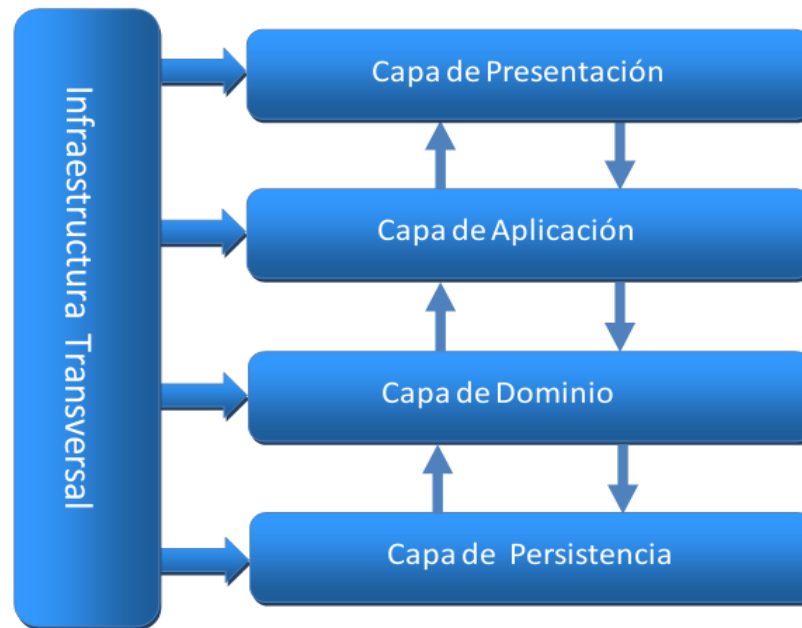
HMAST es el sistema al cual será integrado la solución a desarrollar. Esta herramienta permite administrar los servidores de forma remota, contemplando las funcionalidades necesarias para administrar los usuarios, las tareas programadas y los servicios. A continuación se describen aspectos relevantes de HMAST tales como la arquitectura que utiliza, sus principales funcionalidades, los requisitos no funcionales con los que debe cumplir y por último, se exponen las consideraciones a tener en cuenta al implementar un módulo para dicha herramienta.

Arquitectura del sistema

La arquitectura que presenta esta herramienta propone el diseño de una arquitectura N-Capas basada en el dominio (ver Figura 2), compuesta por cinco capas las cuales son descritas a continuación.

- Capa de la Presentación: Esta capa tiene como responsabilidades presentar al usuario los conceptos de negocio mediante una interfaz de usuario, facilitar la explotación de dichos procesos, informar sobre la situación de los procesos de negocio e implementación de las reglas de validación de dicha interfaz.
- Capa de la Aplicación: Esta capa es la encargada de transmitir los datos que llegan desde la Presentación, realizando las funcionalidades necesarias para que la capa de Dominio reciba la información que le permite satisfacer las peticiones.
- Capa del Dominio: Esta capa constituye el hilo conductor de la aplicación, ya que es la encargada de implementar la lógica de dominio, o sea, las reglas de negocio. En ella se encuentran los servicios, quienes son los responsables de realizar las validaciones necesarias para que la información sea confiable. El dominio también se compone por las entidades, que son la clave para gestionar los datos. Por último, se encuentran los contratos de repositorio que son los encargados de manejar la información presente en la capa de Persistencia.
- Capa de Persistencia: Esta capa es responsable de contener el código necesario para persistir los datos. Los principales componentes que contendrá la misma son los repositorios, que no son más que clases que implementan los contratos de repositorios definidos en la capa de Dominio.

- **Capa de Infraestructura Transversal:** Las responsabilidades de esta capa se basan en promover la reutilización de código, ella albergará las operaciones de seguridad, 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 y eliminación de los datos de un servidor lógico, además posibilita la conexión remota y desconexión a un servidor seleccionado.
- **Gestión de servicios telemáticos asociados a un servidor lógico:** Permite la adición, edición y 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 en un servidor lógico determinado (ficheros de configuración, nombre de módulos, demonios).

Requisitos no funcionales

A continuación quedan descritos y clasificados los requisitos no funcionales de HMAST.

De seguridad:

- Se realizará el registro de todas las operaciones efectuadas sobre los servicios.
- El sistema concederá acceso al mismo a partir de un nombre de usuario y una contraseña.
- El sistema implementará validaciones necesarias para garantizar la integridad de la información que se introduce por el usuario.
- La información que viaja por la red deberá ser encriptada.

De disponibilidad:

- Los usuarios del sistema deben tener acceso (según sus permisos) en todo momento a la información solicitada.
- El sistema estará disponible durante 24 horas, los 7 días de la semana, todos los días del año.

De rendimiento:

- El sistema con el uso de las tecnologías Web debe tener un tiempo de respuesta rápido y eficiente, inferior a 5 segundos.
- Después de instalado el software, el mismo debe ser capaz de soportar más de un usuario conectado simultáneamente.

De apariencia o interfaz externa:

- Debe contener un diseño sencillo, con pocas imágenes y gráficos para acelerar la velocidad de respuesta del sistema.
- El sistema estará configurado para trabajar con una resolución de 1024 x 786.

De hardware:

- El servidor Web donde está instalada la aplicación debe tener 512 MB de RAM y 60 GB de disco duro como mínimo.

- Se debe contar con una conexión superior a los 100 Mb/s de velocidad.

De software:

- Sistema operativo: Nova, Ubuntu o Debian.

De diseño:

- Framework de desarrollo: Spring.
- Lenguaje de programación: Java

De usabilidad:

- La aplicación contará con un menú que le facilite al usuario el acceso a los diferentes módulos para la administración de los servicios.
- Los módulos contarán con un menú que facilite el acceso a las diferentes opciones de configuración de los servicios que se administran.

De soporte:

- Se deben ofrecer servicios de adiestramiento al personal que va a trabajar con el software.
- El sistema contará con un manual de ayuda.

Legales:

- El software así como su documentación, son propios de la UCI.

Consideraciones al implementar un módulo para HMAST

- La lógica de Aplicación no deberá incluir ninguna lógica del Dominio, solo tareas de coordinación, relativas a requerimientos técnicos de la aplicación, como conversiones de formatos de datos de entrada a entidades del Dominio y 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 para la Transferencia de Datos (DTO).
- 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.

Una vez seleccionado el software que será utilizado para administrar el servicio FTP desde HMAST y haber detallado las principales características del sistema en general, se procede a describir los lenguajes herramientas y tecnologías utilizadas en el desarrollo de la solución.

1.5 Lenguajes y framework usados

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 el Lenguaje de Etiquetado Extensible (XML) para una mayor estructuración de las configuraciones de los módulos. A continuación, son descritas cada una de estas tecnologías debido a que se corresponden con las que serán utilizadas en el proceso de implementación del software en cuestión.

1.5.1 Lenguaje de programación

Los lenguajes de programación fueron creados por el hombre con el fin de modelar y hacer en las computadoras programas que simulen la realidad en la que se desarrolla la sociedad. Actualmente existen diferentes lenguajes de programación para desarrollar tanto aplicaciones de escritorio como aplicaciones para la web, los cuales han ido surgiendo debido a las necesidades y tendencias de las plataformas. A continuación se describe el lenguaje Java, el cual fue definido para desarrollar HMAST, herramienta base del módulo en cuestión y por tanto será utilizado para la implementación del mismo.

Java

Java es un lenguaje orientado a objetos, lo que implica que su concepción sea muy próxima a la forma de pensar humana, razón por la cual es calificado como un lenguaje de alto nivel. Fue diseñado para crear software fiables, para ello proporciona numerosas comprobaciones en compilación y en tiempo de ejecución.

Posee características que lo diferencian de otros lenguajes de programación como son [13]:

- Es un lenguaje compilado, generando ficheros de clases compiladas, las cuales son en realidad,

interpretadas por la máquina virtual de Java. Siendo esta máquina virtual la que mantiene el control sobre las clases que se estén ejecutando.

- **Multiplataforma:** El mismo código escrito en Java que funciona en un sistema operativo, funcionará en cualquier otro que tenga instalada la máquina virtual de Java.
- **Seguro:** La máquina virtual, al ejecutar el código en Java, realiza comprobaciones de seguridad, además el propio lenguaje carece de características inseguras, como por ejemplo los punteros.
- **Gracias al API de Java se puede ampliar el lenguaje para que sea capaz de, por ejemplo, comunicarse con equipos mediante red, acceder a bases de datos y crear páginas HTML dinámicas.**

1.5.2 Lenguaje de marcado

Un lenguaje de marcas es una forma de codificar un documento que, junto con el texto, incorpora etiquetas o marcas que contienen información adicional acerca de la estructura del mismo [14]. A menudo suelen ser confundidos con lenguajes de programación. Sin embargo, no son lo mismo, ya que el lenguaje de marcado no tiene funciones aritméticas o variables, como poseen los lenguajes de programación. A continuación queda descrito el XML, el cual es utilizado para almacenar y estructurar las configuraciones del módulo.

XML

Es un lenguaje que se caracteriza por no limitar el número ni el tipo de etiquetas a utilizar, ya que permite crear aquellas que sean necesarias. Ha alcanzado cierta popularidad entre los lenguajes de marcado, la cual se ha acrecentado a lo largo de los últimos años, y se debe al hecho de ser un estándar abierto y además libre, creado por W3C, el consorcio World Wide Web. Fue creado en 1996 y desde ese momento su utilización tuvo un crecimiento sostenido. Es importante destacar algunas de las ventajas que ofrece este lenguaje, un ejemplo de ellas es que si la información se transfiere en XML, cualquier aplicación podría escribir un documento de texto plano con los datos que estaba manejando en formato XML y otra aplicación recibir esta información, pudiendo así, trabajar con ella [15].

1.5.3 Lenguaje de modelado

El lenguaje de modelado de objetos es un conjunto estandarizado de símbolos y de modos de disponerlos para modelar un diseño de software orientado a objetos. El uso de un lenguaje de este tipo, es más

sencillo que la propia programación, pues existen menos medios para verificar de forma efectiva el correcto funcionamiento del modelo [16]. A continuación se describen las características del lenguaje de modelado utilizado para el diseño de la solución.

Lenguaje de Modelado Unificado (UML)

Este lenguaje se centra en la representación gráfica de un sistema, por tanto, UML es un lenguaje para describir modelos. Básicamente, un modelo es una simplificación de la realidad que se construye para comprender mejor el sistema que se quiere desarrollar [17].

Sus funciones son:

- **Especificación:** Permite expresar de una forma gráfica un sistema de forma que otro lo puede entender.
- **Visualización:** Posibilita especificar cuáles son las características de un sistema antes de su construcción.
- **Construcción:** A partir de los modelos especificados se pueden construir los sistemas diseñados.
- **Documentación:** Los propios elementos gráficos sirven como documentación del sistema desarrollado que pueden servir para su futura revisión.

Es importante destacar que aunque UML es orientado a objetos preferentemente, es útil en cualquier modelo tecnológico ya que es independiente de lenguajes de programación o tecnologías determinadas.

1.5.4 Framework de desarrollo

Un *framework* es un esquema o patrón, el cual provee funcionalidades genéricas que pueden ser utilizadas con el fin de desarrollar aplicaciones de manera rápida, fácil, modular y sencilla, ahorrando tiempo y esfuerzo. También es posible que esta tecnología defina una estructura para una aplicación completa, o solo se centre en un aspecto de la misma [18]. A continuación se describe el *framework* de desarrollo Spring, utilizado en la implementación del módulo, para lograr un bajo acoplamiento entre las clases. Quedó definido el uso del mismo debido a que se corresponde con el utilizado la concepción inicial de HMAST.

Spring Framework

Spring Framework, también conocido simplemente como Spring, es un *framework* de código abierto para el desarrollo de aplicaciones en la plataforma Java. Por su diseño, ofrece mucha libertad a los

desarrolladores en Java, así como soluciones bien documentadas y fáciles de usar para las prácticas comunes en la industria.

Sus características principales son:

- La inyección de dependencia: El objetivo es lograr un bajo acoplamiento entre los objetos de la aplicación. Con este patrón de diseño, los objetos no crean o buscan sus dependencias con aquellos objetos con los que colabora, sino que estas son dadas a los mismos. El contenedor (la entidad que coordina cada objeto en el sistema) es el encargado de realizar este trabajo al momento de instanciar el objeto [19].
- La programación orientada a aspectos: La Programación Orientada a Aspectos (POA) es un paradigma de programación relativamente reciente que ha logrado un nivel de desarrollo interesante. Su propósito es permitir una adecuada modularización de las aplicaciones y posibilitar una mejor separación de conceptos, buscando construir aplicaciones más fáciles de implementar, mantener y extender [20].

1.6 Metodología y Herramientas utilizadas

En el desarrollo de un software es imprescindible contar con una metodología para guiar el proceso y que se ajuste a las necesidades del mismo. Otro aspecto importante a tener en cuenta, es la correcta selección de las herramientas y tecnologías que serán utilizadas, garantizando que se correspondan con las exigencias del equipo y el entorno de desarrollo. A continuación se detallan las características de la metodología de desarrollo a utilizar, así como las herramientas y tecnologías definidas en la concepción de HMAST y que lógicamente se ajustan a las que serán utilizadas en el desarrollo del módulo en cuestión.

1.6.1 Metodología de desarrollo

En un proyecto de desarrollo de software la metodología define Quién debe hacer Qué, Cuándo y Cómo debe hacerlo. Por la importancia que tiene el uso de una metodología adecuada que guíe el proceso de desarrollo de un software, la selección de la misma debe realizarse cuidadosamente. El Departamento SIMAYS, en correspondencia con las ventajas que proporciona la metodología ágil SXP, orienta el uso de la misma en los trabajos que pertenezcan a él.

Metodología SXP

Es una metodología ágil desarrollada en la UCI y se caracteriza por tener las buenas prácticas de las metodologías SCRUM y XP. Con el uso de SCRUM para la planificación de los proyectos, SXP logra una eficiente gestión de proyectos de software y con la toma de las mejores prácticas de XP, proporciona una mejor calidad en el producto a entregar. Esta metodología ofrece una estrategia tecnológica, pues utiliza métodos ágiles con el fin de dar solución a muchos de los problemas que afectan el desarrollo y adecuada ejecución de los proyectos, la motivación y entrega de los mismos, aumentando el nivel de preocupación y responsabilidad de los miembros del equipo, promoviendo el compañerismo y solidaridad. Involucra a los miembros del equipo de desarrollo en las decisiones sobre el proyecto y sus vías de desarrollo. Además, el líder de proyecto puede llevar un mejor control de las tareas y la planificación de las mismas. Con el uso de esta metodología se logra una mayor interacción con el cliente, debido a que este forma parte del equipo de desarrollo [21].

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.

SXP consta de 4 fases:

- Planificación-Definición: se establece la visión, se fijan las expectativas y se realiza el aseguramiento del financiamiento del proyecto.
- Desarrollo: se realiza la implementación del sistema.
- Entrega, puesta en marcha.
- Mantenimiento: se realiza el soporte para el cliente.

Cada una de estas fases está compuesta por una serie de actividades que son las que generan los artefactos que quedan incluidos en el nuevo expediente de proyecto, estas actividades están recogidas en el guión de la metodología.

1.6.2 Herramienta de programación

Actualmente existen una serie de herramientas de programación que brindan al programador cierta comodidad para desarrollar software. Uno de los factores a tener en cuenta en el desarrollo de dichos programas es la selección de un entorno de desarrollo correcto, que además sea capaz de apoyar al

desarrollador en términos como el completamiento de código y el detector de errores. A continuación se detallan las características de NetBeans, entorno definido por los desarrolladores de HMAST, la cual se corresponde con la utilizada para desarrollar el módulo con el fin de lograr la compatibilidad necesaria.

NetBeans

La plataforma NetBeans permite que las aplicaciones sean desarrolladas a partir de un conjunto de componentes de software llamados módulos. Debido a que los mismos pueden ser desarrollados independientemente, las aplicaciones basadas en la plataforma NetBeans pueden ser extendidas fácilmente por otros desarrolladores de software [22].

Es el principal competidor en el campo de código abierto frente a Eclipse. Tiene una interfaz clara e intuitiva. Está escrito en Java, pero puede servir para cualquier otro lenguaje de programación. Permite identificar errores, el *debugger* o depurador y cuenta con un aceptable completamiento de código. Es un producto libre y gratuito sin restricciones de uso, fundado por Sun Microsystems en junio de 2000 y actualmente continúa siendo el patrocinador principal de los proyectos.

Para el desarrollo de la solución se utilizará NetBeans en su versión 7.2.1, debido a que ofrece un rendimiento significativamente mejorado con respecto a versiones anteriores y brinda una amplia documentación. Incluye características como el detector errores y posibles soluciones para los mismos, auto completado de código y ordenamiento de código, legible y fácil de entender. Así como mejoras en Java EE, Maven, C/C ++ y PHP. Además, se integra con la herramienta de modelado como Visual Paradigm, brindado de esta forma una gran ventaja al programador.

1.6.3 Herramienta CASE

Las Herramientas de Ingeniería de Software Asistida por Computadora (CASE), son un conjunto de métodos, utilidades y técnicas que facilitan la automatización del ciclo de vida del desarrollo de software. Las Herramientas CASE se iniciaron con un procesador de palabras que fue usado para crear y manipular documentación [23]. A continuación quedan descritas las características de Visual Paradigm, herramienta definida para el modelado en HMAST y por tanto se hace uso de la misma para el diseño los diagramas en el presente módulo.

Visual Paradigm

Visual Paradigm es una herramienta UML profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. El software de

modelado UML ayuda a la construcción de aplicaciones de calidad y a un menor coste. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación. Esta herramienta CASE también proporciona abundantes tutoriales, demostraciones interactivas y proyectos UML [24].

Visual Paradigm soporta un conjunto de lenguajes, tanto en la generación de código como en la ingeniería inversa. Puede generar código a partir de los modelos y viceversa. Cualquiera de los cambios que se realicen en el código existente puede reflejarse en el modelo. Esta herramienta visual permite construir la aplicación con mayor rapidez, mayor exactitud, mejor trabajo en equipo y es fácil de utilizar.

1.6.4 Herramienta auxiliar

Además de las herramientas anteriormente mencionadas se utiliza RapidSVN, establecida para mantener el control de versiones de ficheros, la cual permite realizar cambios sobre los elementos almacenados y manejar el registro histórico de las acciones realizadas con cada elemento o conjunto de elementos. A continuación son expuestas sus características.

RapidSVN

RapidSVN es un cliente de interfaz gráfica para la comunicación con servidores *Subversion*⁶, se distribuye bajo la Licencia Pública General de GNU, conocida como licencia GPL. Está completamente escrito en C++. Facilita el mantenimiento de las diferentes versiones de ficheros, mediante una interfaz sencilla e intuitiva. Es una herramienta rápida y eficiente que se puede ejecutar en plataformas Windows, Linux, MAC OS X y Solaris.

1.7 Tecnología utilizada para el manejo de configuraciones en los ficheros

Con el objetivo de contar con un mecanismo flexible para el manejo de los ficheros de configuración, en el desarrollo de HMAST se define el uso de la librería Augeas. En correspondencia con esta decisión, será utilizada dicha librería para facilitar el trabajo con el fichero de configuración de VsFTPd. A continuación quedan descritas sus características.

1.7.1 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

⁶ **Subversion** es un sistema de control de versiones.

su realización utiliza unas estructuras llamadas Lenses (en español lentes), que permiten que Augeas pueda entender cualquier archivo de configuración. Es una herramienta modular, construida con el lenguaje de programación C, brindando rapidez y eficiencia desde el punto de vista del uso de recursos computacionales. Además 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 [25].

1.8 Conclusiones parciales

En el desarrollo del capítulo fueron analizados los servidores FTP de código abierto, con el propósito de seleccionar el que más se ajustara al objetivo de la investigación. Según el estudio realizado la autora llegó a la conclusión de que el más idóneo para establecer la administración del servicio FTP en la herramienta HMAST es VsFTPd, teniendo en cuenta la seguridad que ofrece, su facilidad de configuración, además de contemplar configuraciones tanto básicas como avanzadas, que tributan al correcto funcionamiento de un servidor FTP. También quedó descrita la forma en que algunas de las herramientas existentes para la administración de servidores, manejan el servicio FTP. Dicho estudio arrojó como resultado que es necesario explotar todas las ventajas y funciones que brinda VsFTPd, considerando elementos fundamentales de este servicio, como son las configuraciones por tipo de usuarios, el límite de concurrencia, el ancho de banda de las transferencias de datos, así como otras configuraciones generales, brindándole la posibilidad al usuario de personalizarlas, con el propósito de lograr una mayor flexibilidad.

Además, se realizó una breve descripción del funcionamiento y las principales características de HMAST, lo que permitió tomar como punto de partida dichas consideraciones para concebir la solución al problema planteado. Por último, fueron detalladas cada una de las herramientas, lenguajes de programación y tecnologías que quedaron definidas para el desarrollo del sistema en general, las cuales lógicamente se aplican al módulo en cuestión. Todo este proceso está guiado por la metodología SXP, mediante la cual se logra gestionar de forma precisa el desarrollo del software garantizando la calidad del producto final.

Capítulo 2: Características del módulo

El presente capítulo se basa en describir la propuesta del software a desarrollar, tomando como punto de partida las características del sistema base HMAST. Además, son detalladas las funcionalidades que debe cumplir la aplicación, las cuales se especifican mediante las Historias de Usuario. También se define y describe la arquitectura y los patrones de diseño a utilizar en el desarrollo del módulo.

2.1 Propuesta del módulo a desarrollar

Con el propósito de darle solución al problema planteado, el presente trabajo de diploma propone desarrollar las funcionalidades necesarias para administrar la transferencia de archivos mediante el protocolo FTP en la herramienta HMAST, haciendo uso del software VsFTPd. El módulo a desarrollar permitirá realizar la transferencia de archivos entre equipos remotos, logrando una independencia entre el cliente y el servidor. El software brindará la posibilidad de realizar configuraciones necesarias por tipo de usuario, así como configuraciones generales y avanzadas, como configurar opciones para el control de acceso y limitaciones de ancho de banda. Además, será capaz de gestionar las conexiones pasivas y conexiones seguras, estas últimas mediante el protocolo SSL. Todas estas funcionalidades hacen posible el correcto funcionamiento del servicio en cuestión.

2.2 Requisitos funcionales y no funcionales

Los requerimientos se definen como las condiciones que el sistema debe cumplir o capacidades que debe tener, con el objetivo de establecer un entendimiento común entre el usuario y el proyecto de software. Los mismos se clasifican en funcionales y no funcionales. Los requisitos funcionales son las características o funcionalidades con las que cuenta el sistema, mientras que los no funcionales se refieren a cualidades que imponen restricciones en el diseño y la implementación [26].

Según la metodología utilizada se genera el artefacto Lista de Reserva del Producto (LRP), donde quedan definidas las características y cualidades con los que debe cumplir el módulo a desarrollar.

2.2.1 LRP

En el proceso de captura de requisitos se confecciona la LRP, en la cual, quedan detalladas las funcionalidades que tendrá el producto en forma de requisitos técnicos y de negocio. Es una lista priorizada y garantiza la organización de los requisitos funcionales y no funcionales, partiendo de la prioridad que tengan para el desarrollo del sistema. Además facilita el trabajo en la confección de las

Historias de Usuario [27].

En el desarrollo del módulo se definen diferentes prioridades para los requisitos funcionales, considerando las exigencias del cliente y la importancia que posean para el funcionamiento del servicio en cuestión. A continuación se expone la LRP del módulo a desarrollar, quedando detalladas las funcionalidades con las que debe cumplir, así como los requisitos no funcionales referentes a las categorías de diseño y software definidos para el proceso de desarrollo, debido a que las demás clasificaciones como son seguridad, disponibilidad, rendimiento, hardware, usabilidad, soporte, entre otras, son manejadas por el sistema de forma global y quedaron descritas en el epígrafe 1.4 Descripción de HMAST.

Asignado a	Ítem *	Descripción	Estimación	Estimado por
	Prioridad	Alta		
Programador	1	Instalar el servicio VsFTPd.	0.1	Adrianet Cobo Fonseca
Programador	2	Desinstalar el servicio VsFTPd.	0.1	Adrianet Cobo Fonseca
Programador	3	Establecer configuraciones predeterminadas del servicio VsFTPd.	0.3	Adrianet Cobo Fonseca
Programador	4	Iniciar servicio VsFTPd.	0.1	Adrianet Cobo Fonseca
Programador	5	Detener servicio VsFTPd.	0.1	Adrianet Cobo Fonseca
Programador	6	Reiniciar servicio VsFTPd.	0.1	Adrianet Cobo Fonseca
Programador	7	Recargar servicio VsFTPd.	0.1	Adrianet Cobo Fonseca
Programador	8	Permitir autenticación en el servidor FTP de usuarios locales.	0.1	Adrianet Cobo Fonseca
Programador	9	Denegar autenticación en el servidor FTP de usuarios locales.	0.1	Adrianet Cobo Fonseca
Programador	10	Compartir un directorio específico para usuarios locales.	0.1	Adrianet Cobo Fonseca
Programador	11	Modificar espacio compartido para usuarios locales.	0.1	Adrianet Cobo Fonseca
Programador	12	Mostrar espacio compartido para usuarios locales.	0.1	Adrianet Cobo Fonseca
Programador	13	Adicionar usuario local con acceso a directorio específico.	0.2	Adrianet Cobo Fonseca

Módulo de administración del servicio FTP
Capítulo 2: Características del módulo

Asignado a	Ítem *	Descripción	Estimación	Estimado por
Programador	14	Editar usuario en carpeta específica.	0.2	Adrianet Cobo Fonseca
Programador	15	Eliminar el acceso a usuarios sobre carpeta específica.	0.1	Adrianet Cobo Fonseca
Programador	16	Gestionar permisos sobre el nuevo directorio para usuarios que tienen acceso.	0.2	Adrianet Cobo Fonseca
Programador	17	Establecer sistema de salvas para usuarios locales.	0.2	Adrianet Cobo Fonseca
Programador	18	Modificar tamaño de cuota para usuarios locales.	0.2	Adrianet Cobo Fonseca
Programador	19	Mostrar tamaño de cuota para usuarios locales.	0.1	Adrianet Cobo Fonseca
Programador	20	Establecer tamaño de cuota para usuarios específicos.	0.1	Adrianet Cobo Fonseca
Programador	21	Modificar tamaño de cuota para usuarios específicos.	0.1	Adrianet Cobo Fonseca
Programador	22	Mostrar tamaño de cuota para usuarios específicos.	0.1	Adrianet Cobo Fonseca
Programador	23	Habilitar el enjaulado de usuarios locales.	0.1	Adrianet Cobo Fonseca
Programador	24	Deshabilitar el enjaulado de usuarios locales.	0.1	Adrianet Cobo Fonseca
Programador	25	Establecer lista de usuarios locales que no estarán enjaulados en su espacio raíz.	0.1	Adrianet Cobo Fonseca
Programador	26	Adicionar usuarios a la lista de usuarios locales que no estarán enjaulados en su espacio raíz.	0.1	Adrianet Cobo Fonseca
Programador	27	Eliminar usuarios de la lista de usuarios locales que no estarán enjaulados en su espacio raíz.	0.1	Adrianet Cobo Fonseca
Programador	28	Establecer ubicación de la lista de usuarios no enjaulados.	0.1	Adrianet Cobo Fonseca
Programador	29	Modificar ubicación de la lista de usuarios no enjaulados.	0.1	Adrianet Cobo Fonseca
Programador	30	Mostrar ubicación de la lista de usuarios no enjaulados.	0.1	Adrianet Cobo Fonseca
Programador	31	Establecer la velocidad de transferencia para usuarios autenticados.	0.1	Adrianet Cobo Fonseca
Programador	32	Modificar la velocidad de transferencia para usuarios autenticados.	0.1	Adrianet Cobo Fonseca

Módulo de administración del servicio FTP
Capítulo 2: Características del módulo

Asignado a	Ítem *	Descripción	Estimación	Estimado por
Programador	33	Mostrar la velocidad de transferencia establecida para usuarios autenticados.	0.1	Adrianet Cobo Fonseca
Programador	34	Permitir acceso a usuarios anónimos.	0.1	Adrianet Cobo Fonseca
Programador	35	Denegar acceso a usuarios anónimos.	0.1	Adrianet Cobo Fonseca
Programador	36	Otorgar a usuarios anónimos el permiso de subir archivos al servidor.	0.1	Adrianet Cobo Fonseca
Programador	37	Denegar a usuarios anónimos el permiso de subir archivos al servidor.	0.1	Adrianet Cobo Fonseca
Programador	38	Permitir a usuarios anónimos descargar archivos que tengan permisos de solo lectura.	0.1	Adrianet Cobo Fonseca
Programador	39	Denegar a usuarios anónimos descargar archivos que tengan permisos de solo lectura.	0.1	Adrianet Cobo Fonseca
Programador	40	Permitir a usuarios anónimos la creación de directorios.	0.1	Adrianet Cobo Fonseca
Programador	41	Denegar a usuarios anónimos la creación de directorios.	0.1	Adrianet Cobo Fonseca
Programador	42	Compartir un directorio específico para usuarios anónimos.	0.1	Adrianet Cobo Fonseca
Programador	43	Modificar espacio compartido para usuarios anónimos.	0.1	Adrianet Cobo Fonseca
Programador	44	Mostrar espacio compartido para usuarios anónimos.	0.1	Adrianet Cobo Fonseca
Programador	45	Establecer la velocidad de transferencia establecida para usuarios anónimos.	0.1	Adrianet Cobo Fonseca
Programador	46	Modificar la velocidad de transferencia para usuarios anónimos.	0.1	Adrianet Cobo Fonseca
Programador	47	Mostrar la velocidad de transferencia establecida para usuarios anónimos.	0.1	Adrianet Cobo Fonseca
Programador	48	Habilitar soporte SSL.	0.1	Adrianet Cobo Fonseca
Programador	49	Deshabilitar soporte SSL.	0.1	Adrianet Cobo Fonseca
Programador	50	Configurar el establecimiento de	0.1	Adrianet Cobo

Módulo de administración del servicio FTP
Capítulo 2: Características del módulo

Asignado a	Ítem *	Descripción	Estimación	Estimado por
		conexiones encriptadas mediante SSL para usuarios locales.		Fonseca
Programador	51	Permitir que las transferencias de datos locales sean encriptados con SSL.	0.1	Adrianet Cobo Fonseca
Programador	52	Denegar que las transferencias de datos locales sean encriptados con SSL.	0.1	Adrianet Cobo Fonseca
Programador	53	Habilitar soporte para diversas versiones de SSL.	0.1	Adrianet Cobo Fonseca
Programador	54	Deshabilitar soporte para diversas versiones de SSL.	0.1	Adrianet Cobo Fonseca
Programador	55	Generar certificado SSL por defecto.	0.2	Adrianet Cobo Fonseca
Programador	56	Subir certificado SSL y clave pública.	0.1	Adrianet Cobo Fonseca
Programador	57	Especificar la ubicación del certificado SSL.	0.1	Adrianet Cobo Fonseca
Programador	58	Mostrar ubicación del certificado.	0.1	Adrianet Cobo Fonseca
Programador	59	Modificar la ubicación del certificado SSL generado.	0.1	Adrianet Cobo Fonseca
Programador	60	Especificar la ubicación de la clave pública.	0.1	Adrianet Cobo Fonseca
Programador	61	Mostrar ubicación de la clave pública.	0.1	Adrianet Cobo Fonseca
Programador	62	Modificar la ubicación de la clave pública.	0.1	Adrianet Cobo Fonseca
Programador	63	Permitir conexiones pasivas.	0.1	Adrianet Cobo Fonseca
Programador	64	Denegar conexiones pasivas.	0.1	Adrianet Cobo Fonseca
Programador	65	Establecer rango de puertos para conexiones en modo pasivo.	0.1	Adrianet Cobo Fonseca
programador	66	Modificar rango de puertos para conexiones pasivas.	0.1	Adrianet Cobo Fonseca
Programador	67	Mostrar rango de puerto para conexiones en modo pasivo.	0.1	Adrianet Cobo Fonseca
Programador	68	Activar la generación de registros.	0.1	Adrianet Cobo Fonseca
Programador	69	Desactivar la generación de registros.	0.1	Adrianet Cobo Fonseca
Programador	70	Definir ubicación del fichero de	0.1	Adrianet Cobo

Módulo de administración del servicio FTP
Capítulo 2: Características del módulo

Asignado a	Ítem *	Descripción	Estimación	Estimado por
		los registros.		Fonseca
Programador	71	Modificar ubicación del fichero de los registros.	0.1	Adrianet Cobo Fonseca
Programador	72	Mostrar ubicación del fichero de los registros.	0.1	Adrianet Cobo Fonseca
Prioridad		Media		
Programador	73	Activar modo de escritura.	0.1	Adrianet Cobo Fonseca
Programador	74	Desactivar modo de escritura.	0.1	Adrianet Cobo Fonseca
Programador	75	Establecer máscara para los archivos cuando son creados.	0.1	Adrianet Cobo Fonseca
Programador	76	Modificar máscara para los archivos cuando son creados.	0.1	Adrianet Cobo Fonseca
Programador	77	Mostrar máscara para los archivos cuando son creados.	0.1	Adrianet Cobo Fonseca
Programador	78	Establecer la cantidad máxima permitida de clientes conectados simultáneamente.	0.1	Adrianet Cobo Fonseca
Programador	79	Modificar la cantidad máxima permitida de clientes simultáneos conectados.	0.1	Adrianet Cobo Fonseca
Programador	80	Mostrar la cantidad máxima permitida de clientes conectados simultáneamente.	0.1	Adrianet Cobo Fonseca
Programador	81	Establecer la cantidad máxima permitida de conexiones simultáneas por IP.	0.1	Adrianet Cobo Fonseca
Programador	82	Modificar la cantidad máxima permitida de conexiones simultáneas por IP.	0.1	Adrianet Cobo Fonseca
Programador	83	Mostrar la cantidad permitida de conexiones simultáneas por IP.	0.1	Adrianet Cobo Fonseca
Programador	84	Establecer la máxima cantidad permitida de intentos de autenticación fallidos por sesión.	0.1	Adrianet Cobo Fonseca
Programador	85	Modificar la máxima cantidad permitida de intentos de autenticación fallidos por sesión.	0.1	Adrianet Cobo Fonseca
Programador	86	Mostrar la máxima cantidad permitida de intentos de autenticación fallidos por	0.1	Adrianet Cobo Fonseca

Asignado a	Ítem *	Descripción	Estimación	Estimado por
		sesión.		
Programador	87	Establecer tiempo máximo permitido de espera para mantener conexiones inactivas.	0.1	Adrianet Cobo Fonseca
Programador	88	Modificar tiempo máximo permitido de espera para mantener conexiones inactivas.	0.1	Adrianet Cobo Fonseca
Programador	89	Mostrar tiempo máximo permitido de espera para mantener conexiones inactivas.	0.1	Adrianet Cobo Fonseca
Programador	90	Establecer tiempo máximo permitido de espera para mantener conexiones de datos inactivas.	0.1	Adrianet Cobo Fonseca
Programador	91	Modificar tiempo máximo permitido de espera para mantener conexiones de datos inactivas.	0.1	Adrianet Cobo Fonseca
Programador	92	Mostrar tiempo máximo permitido de espera para mantener conexiones de datos inactivas.	0.1	Adrianet Cobo Fonseca
Programador	93	Activar la ejecución del servidor en modo standalone (independiente).	0.1	Adrianet Cobo Fonseca
Programador	94	Desactivar la ejecución del servidor en modo standalone (independiente).	0.1	Adrianet Cobo Fonseca
Programador	95	Establecer texto de bienvenida que aparece cuando el usuario accede al servidor.	0.1	Adrianet Cobo Fonseca
Programador	96	Cambiar texto de bienvenida.	0.1	Adrianet Cobo Fonseca
Programador	97	Eliminar texto de bienvenida.	0.1	Adrianet Cobo Fonseca
RNF (Requisitos No Funcionales)				
Software				
Programador	1	La aplicación se ejecutará en sistemas operativos libres (Nova, Ubuntu o Debian).		
Diseño				
Programador	2	Se utilizará el lenguaje de programación Java con el framework de desarrollo Spring.		

Asignado a	Ítem *	Descripción	Estimación	Estimado por
Programador	3	Se utilizará la librería augeas-java, para el manejo de la información presente en los ficheros de configuración.		
Programador	4	Se utilizará el patrón arquitectónico N-capas orientado al dominio.		
Programador	5	La herramienta de programación a utilizar será NetBeans 7.2.1.		
Programador	6	Se utilizará Visual Paradigm 8.0 como herramienta de modelado.		

Tabla 2: Lista de Reserva del Producto (LRP).

2.3 Historias de Usuarios (HU)

En la metodología SXP, las HU constituyen la técnica utilizada con el propósito de especificar los requisitos del software. Sirven de guía en la construcción de las pruebas de aceptación y son utilizadas para estimar tiempos de desarrollo. En este sentido, solo proveen detalles suficientes para hacer una estimación razonable del tiempo que llevará implementarlas [28].

A continuación se describen las HU correspondientes para el software a implementar, teniendo en cuenta los requisitos funcionales asociados a cada una de ellas:

Número	Nombre de la HU	Requisitos asociados	Descripción
HMAST_FTP_1	Instalar y desinstalar servicio VsFTPD.	1, 2, y 3.	El software permite instalar el servicio, así como establecer opciones de configuración predeterminadas antes de instalarlo.
HMAST_FTP_2	Cambiar estado del servicio.	4, 5, 6 y 7.	El sistema brinda la posibilidad de Iniciar, Detener, Reiniciar y Recargar el servicio.
HMAST_FTP_3	Realizar configuración de usuarios locales en el servidor.	8, 9 10, 11, 12, 13, 14, 15, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 y 33.	El software es capaz de permitir o denegar la autenticación de los usuarios locales. En caso de que se

Módulo de administración del servicio FTP
Capítulo 2: Características del módulo

Número	Nombre de la HU	Requisitos asociados	Descripción
			<p>permita, entonces el sistema brinda la posibilidad de realizar una serie de configuraciones como son: establecer, mostrar y modificar el directorio raíz por defecto cuando se autentican en el servidor, establecer un sistema de salvas para este tipo de usuarios, así como limitar y mostrar el tamaño de la cuota. Además brinda la posibilidad de adicionar o eliminar usuarios de la lista que muestra los usuarios locales que no serán enjaulados en su espacio raíz y establecer, mostrar y modificar el valor de la velocidad de transferencia de los datos para este tipo de usuarios, así como establecer y modificar el espacio personal para usuarios específicos, definiendo además los permisos que tendrán sobre el mismo.</p>
HMAST_FTP_4	Realizar configuración de acceso para usuarios anónimos al servidor.	34, 35, 36, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46 y 47.	El software es capaz de permitir o denegar la autenticación de usuarios anónimos en el servidor, en caso de que se permita,

Módulo de administración del servicio FTP
Capítulo 2: Características del módulo

Número	Nombre de la HU	Requisitos asociados	Descripción
			entonces el sistema brinda la posibilidad de realizar una serie de configuraciones como son: establecer, mostrar y modificar el espacio al cual tienen acceso en el servidor, otorgar o denegar permisos a usuarios anónimos y establecer, mostrar y modificar el valor de la velocidad de transferencia de los datos para este tipo de usuarios.
HMAST_FTP_5	Configurar conexiones SSL.	48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61 y 62.	El software permite habilitar o deshabilitar las conexiones SSL y en caso de que se habilite permite realizar una serie de configuraciones así como gestionar los certificados SSL.
HMAST_FTP_6	Configurar conexiones pasivas.	63, 65, 66 y 67.	El software permite al usuario activar o desactivar el establecimiento de conexiones pasivas, así como especificar, modificar y mostrar el rango de puertos para ese tipo de conexiones.
HMAST_FTP_7	Realizar configuraciones generales.	73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96 y 97.	El software permite activar o desactivar el modo de escritura en el servidor. Es capaz de establecer, modificar y mostrar

Módulo de administración del servicio FTP
Capítulo 2: Características del módulo

Número	Nombre de la HU	Requisitos asociados	Descripción
			<p>la máscara que tendrán los archivos cuando son creados, así como gestionar el texto de bienvenida, que no es más que el texto que se muestra cuando el usuario se autentifica en el servidor. También permite activar o desactivar los mensajes de directorios. Brinda la posibilidad de configurar el tiempo máximo de espera para mantener conexiones inactivas y conexiones de datos inactivas. Permite activar o desactivar la ejecución del servidor en modo independiente (<i>standalone</i>). Además, establecer la cantidad máxima permitida de intentos de acceso fallidos por sesión y clientes e IP conectados de forma simultánea.</p>
HMAST_FTP_8	Configurar registros en el servidor.	68, 70, 71, y 72.	<p>El software permite activar o desactivar la generación de registros en el servidor y si está activado permite establecer, mostrar y modificar la ubicación de los registros en el servidor.</p>

Tabla 3: Listado de las HU.

2.4 Arquitectura del software

El diseño de la arquitectura de un sistema es el proceso por el cual se define una solución para los requisitos técnicos y operacionales del mismo. Este proceso define qué componentes forman el sistema, cómo se relacionan entre ellos, y cómo mediante su interacción llevan a cabo la funcionalidad especificada, cumpliendo con los criterios de calidad indicados como seguridad, disponibilidad, eficiencia o usabilidad [29].

2.4.1 Estilos arquitectónicos

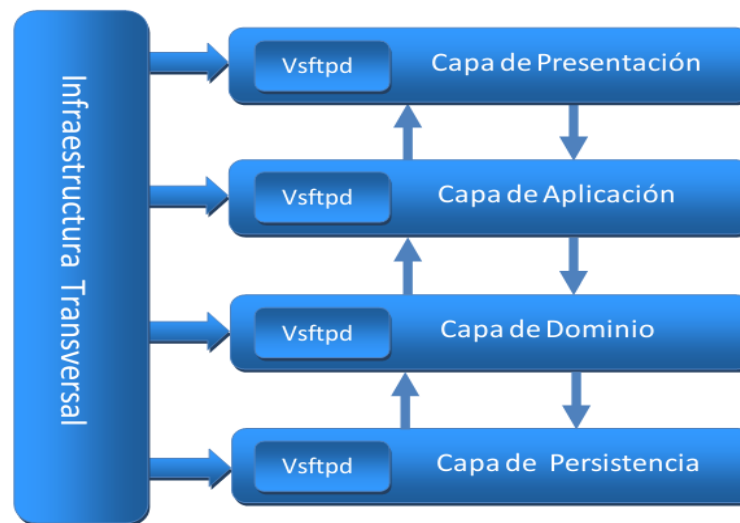
Un estilo arquitectural se puede entender como un conjunto de principios que definen a alto nivel un aspecto de la aplicación. Está definido por un grupo de componentes, un conjunto de conexiones entre dichos componentes y restricciones sobre cómo se comunican dos componentes cualesquiera conectados. Los estilos arquitecturales pueden organizarse según el aspecto de la aplicación sobre el que se centran. Los principales aspectos son: comunicaciones, despliegue, dominio, interacción y estructura [29].

El sistema base del módulo a implementar define el uso del estilo arquitectónico basado en el dominio. A continuación se describe el funcionamiento del mismo.

Estilo arquitectónico Arquitectura N-Capas Orientada al Dominio

La Arquitectura Orientada al Dominio no es solo un estilo arquitectural, es también una forma de afrontar los proyectos a nivel de trabajo del equipo de desarrollo. Razones por las cuales es importante el uso de una Arquitectura N-Capas Orientada al Dominio es especialmente en los casos donde el comportamiento del negocio a automatizar está sujeto a muchos cambios y evoluciones. El objetivo de la misma es estructurar de forma clara la complejidad de una aplicación empresarial basada en las diferentes capas de la arquitectura siguiendo el patrón N-Capas y las tendencias de arquitecturas orientadas al dominio [29].

En el diseño de HMAST quedó definida la utilización de una arquitectura N-Capas orientada al Dominio. Teniendo en cuenta que el módulo a desarrollar será integrado a dicha herramienta y persiguiendo el propósito de mantener una homogeneidad entre los componentes del sistema, es necesario guiar el diseño a través del patrón arquitectónico definido para la misma (ver Figura 3), incluyendo en este la distribución del módulo para la administración del servicio FTP en cada una de las capas, considerando las responsabilidades que cada una establezca.



Una de las ventajas que trae asociadas el uso de una Arquitectura N-Capas orientada al Dominio, es que facilita el mantenimiento de la aplicación, debido a que las responsabilidades están perfectamente distribuidas en las diferentes capas de la misma. Además es recomendable para aquellos proyectos cuya lógica del negocio esté sujeta a constantes cambios y mantenimientos.

2.5 Diagrama de Paquetes

El Diagrama de paquetes muestra cómo un sistema está dividido en agrupaciones lógicas reflejando 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 [30].

Sus elementos son:

- Paquetes: Agrupación de elementos.
- Dependencias: Indican que un elemento de un paquete requiere a otro paquete distinto.

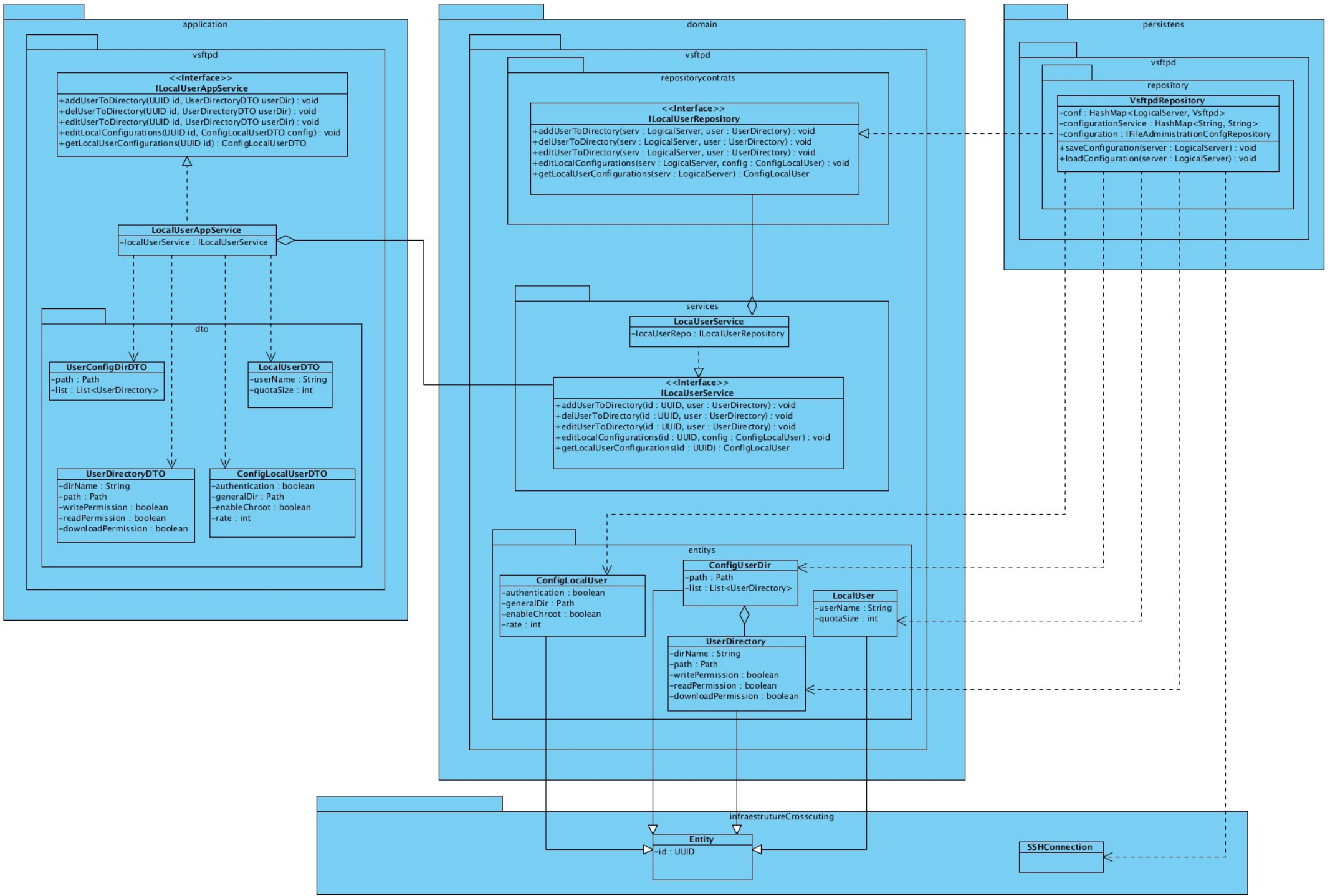
El diagrama que se muestra en la Figura 4 representa la distribución de las clases para la implementación de la historia de usuario Realizar configuraciones para usuarios locales en el servidor. A continuación se explica la relación entre las clases comprendidas en cada uno de los paquetes. Tomando en consideración que el módulo a desarrollar no contará con una interfaz visual, no se contempla en el diseño del diagrama de paquetes capa de Presentación, debido a que es precisamente esta capa la encargada de interactuar

con el usuario mediante una interfaz gráfica.

En la capa de Aplicación (*application*) se encuentra la clase interfaz `ILocalUserAppService` que contiene los métodos necesarios para permitir el acceso desde la capa de presentación, los cuales son implementados por la clase `LocalUserAppService`. 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 `localUserService` contenido en esta clase. Esta capa contiene además, el paquete DTO (*Data Object Transfer*), el cual es el encargado de almacenar clases `LocalUserDTO`, `ConfigLocalUserDTO` y `UserDirectoryDTO`, las cuales representan la información que llega desde la capa de presentación.

Por su parte la capa de Dominio (*domain*) contiene los subpaquetes Entidades (*entitys*), Servicios (*services*) y Contratos de repositorios (*repositoriescontrats*). El subpaquete Servicio contiene la clase `ILocalUserService` que define los métodos que son accedidos desde la capa de aplicación, e implementados por la clase `LocalUserService`, quien es la encargada de realizar las validaciones de los datos antes de realizar las operaciones en el repositorio, con el propósito de garantizar que la información sea confiable. El acceso al repositorio se realiza a través del atributo `localUserRepo` definido en esta clase. En el subpaquete Contratos de repositorios se encuentra la clase `ILocalUserRepository`, la cual define los métodos que son accedidos desde el subpaquete Servicio y son implementados por la clase `VsftpdRepository` encontrada en la capa de persistencia de los datos (*persistens*). Esta clase es responsable de cargar y salvar la información en los repositorios. Por último, en esta capa se encuentran las Entidades `LocalUser`, `ConfigLocalUser`, `ConfigUserDir` y `UserDirectory`, quienes son la base para manejar los datos del servicio FTP relacionados en este caso con los usuarios locales del sistema.

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

En la terminología de objetos, el patrón es una descripción de un problema y su solución que recibe un nombre y que puede emplearse en otros contextos; en teoría, indica la manera de utilizarlo en circunstancias diversas. Muchos patrones ofrecen orientación sobre cómo asignar las responsabilidades a los objetos ante determinada categoría de problemas [31]. 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.

2.6.1 Patrones GRASP

GRASP es un acrónimo que significa *General Responsibility Assignment Software Patterns*, del español Patrones Generales de Software para Asignar Responsabilidades. Los patrones GRASP describen los principios fundamentales para la asignación de responsabilidades a objetos, expresado en forma de patrones [31]. 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.

En la aplicación se necesita adicionar, eliminar y editar un objeto de tipo `UserDirectory` que define los permisos para usuarios específicos sobre un directorio determinado. En este caso la clase `UserConfigDir` que contiene una lista de `UserDirectory` es la experta en información y por tanto, la encargada de realizar las operaciones anteriormente mencionadas sobre la lista.

- **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 la aplicación el uso de este patrón se evidencia cuando las clases presentes en el dominio crean instancias de las entidades. Un ejemplo es cuando la clase LocalUserService adiciona un objeto de tipo UserDirectory.

- **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.

Los patrones de alta cohesión y bajo acoplamiento se evidencian en el uso de la inyección de dependencia, la cual consiste en hacer que las clases del software sean independientes comunicándose únicamente a través de interfaces.

2.6.2 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 [32]. A continuación son detallados los que han sido utilizados 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 la aplicación para establecer la conexión con el servidor que se desee administrar se hace una única

instancia del objeto SSHConnection, presente en el paquete infrastructureCrosscutting.

2.7 Conclusiones parciales

El módulo a implementar tendrá un total de 97 funcionalidades, las cuales están organizadas para conformar un total de 8 Historias de Usuario y 6 requisitos no funcionales. La descripción de las Historias de Usuario permitió un mejor entendimiento entre el programador y el cliente, llegando a un acuerdo sobre las capacidades y cualidades con las que el software debe cumplir. Además, la correcta definición de los patrones de arquitectura y los de diseño, permitieron que se obtuviera una base del sistema capaz de soportar posteriores cambios en los requerimientos.

El estudio realizado en este capítulo ha facilitado un entendimiento de la dinámica y estructura de la aplicación, contribuyendo a una mejor comprensión del problema que el módulo a desarrollar debe resolver. Además se sentaron las bases para las restantes fases del proceso.

Capítulo 3: Desarrollo de la solución propuesta

El presente capítulo se centra en describir el proceso realizado para la implementación del módulo, tomando como punto de partida la planificación de las iteraciones con el propósito de dar cumplimiento a los requisitos que han sido identificados en la propuesta de solución. Además, son descritas las Tareas de Ingeniería propuestas para las Historias de Usuarios (HU) definidas. Por otra parte, en el flujo de trabajo de prueba, se verificará el resultado de la implementación a través de la ejecución de pruebas de unidad mediante el método de Caja Blanca, donde se comprueba el funcionamiento de cada funcionalidad definida para el módulo.

3.1 Plan de Iteraciones

Una vez analizadas las 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 etapa 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 Historias de Usuario de la 1 a la 6. Estas HU contemplan las funcionalidades más importantes, que responden a las características principales de un servidor FTP.
- Iteración 2: La segunda iteración tiene como objetivo dar cumplimiento a las Historias de Usuario 7 y 8. Con la implementación de estas funcionalidades se completan las demás configuraciones básicas con las que debe contar el servidor.

3.2 Tareas de ingeniería

Las Tareas de Ingeniería se llevan a cabo con el fin de detallar de forma más amplia las HU, facilitando con ello, el entendimiento en el proceso de implementación. Cada HU puede contener una o más tareas en caso de necesitarla, explicando paso a paso las acciones que se realizan en la misma. Los puntos de estimación asignados a cada tarea son medidos por días según las características y la complejidad que posea la misma. Estas tareas definen cada una de las actividades asociadas a las HU y permiten organizar el proceso de implementación, así como conocer el grado de complejidad de cada HU, teniendo en cuenta la cantidad de tareas asociadas [33]. A continuación se exponen las Tareas de Ingeniería

identificadas en el desarrollo del módulo.

Tareas de ingeniería	HU asociadas
Probar el comando para realizar la instalación del servidor.	HMAST_FTP_1
Probar los comandos que permiten Iniciar, Detener, Recargar y Reiniciar el servicio.	HMAST_FTP_2
Comprobar el funcionamiento de la directiva que permite la autenticación de usuarios locales.	HMAST_FTP_3
Comprobar el funcionamiento de la directiva que permite establecer el directorio raíz de usuarios locales.	
Comprobar el funcionamiento de las directivas que permiten establecer el enjaulado de usuarios locales en el espacio raíz definido.	
Comprobar el funcionamiento de la directiva que permiten establecer la velocidad de transferencia para usuarios locales.	
Comprobar el funcionamiento de las directivas que permiten establecer configuraciones específicas para determinados usuarios locales.	
Comprobar el funcionamiento de los paquetes quota y quotatool.	
Comprobar el funcionamiento de la directiva para permitir o no la autenticación de usuarios anónimos en el servidor.	
Comprobar el funcionamiento de la directiva para establecer directorio al cual acceden los usuarios anónimos en el servidor.	
Comprobar el funcionamiento de las directivas para configurar los permisos de los usuarios anónimos en el servidor.	
Comprobar el funcionamiento de la directiva que permiten establecer la velocidad de transferencia para usuarios anónimos.	
Comprobar el funcionamiento de las directivas para configurar las conexiones SSL.	HMAST_FTP_5
Comprobar el funcionamiento de las directivas y comandos para gestionar certificados SSL.	
Comprobar el funcionamiento de las directivas para configurar las conexiones pasivas.	HMAST_FTP_6
Comprobar el funcionamiento de las directivas relacionadas con las	HMAST_FTP_7

Tareas de ingeniería	HU asociadas
configuraciones generales del servicio.	
Comprobar el funcionamiento de las directivas para la configuración de los registros del servicio.	HMAST_FTP_8

Tabla 4: Listado de Tareas de Ingeniería.

3.3 Estándares de Codificación

En general, un estándar de codificación es el conjunto de reglas que se siguen para la escritura del código fuente; de tal manera que otros programadores puedan entenderlo sin muchas dificultades. Entre los elementos claves dentro de los estándares de código se encuentran los identificadores de las variables, las funciones o métodos, así como todos los demás elementos dentro del código fuente del software a desarrollar, con el propósito de lograr que sea más legible y de fácil mantenimiento.

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: *VsftpdRepository*.

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 utilizará la notación Camel, y con solo leerlo se reconoce el propósito de la misma.

Ejemplo: *editGeneralConfigurations*.

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: *writePermission*.

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 (vsftpd).

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

kkk → clases o subpaquetes.

Ejemplo: *cu.uci.hmast.domain.vsftpd.entitys.LocalUser.*

3.4 Pruebas de software

Persiguiendo el propósito de verificar la calidad de un software, se realizan un conjunto de pruebas que demuestran si las funcionalidades cumplen con lo especificado por el cliente. Para probar cualquier producto de software son considerados dos enfoques, uno donde conociendo la función específica para la que fue diseñada el producto, se pueden llevar a cabo pruebas dirigidas a demostrar que cada funcionalidad es completamente operativa, a medida que se prueban las mismas; un segundo enfoque sería cuando conociendo el funcionamiento del software, se puedan desarrollar pruebas que aseguren que todos los componentes internos se han comprobado de forma adecuada, lo cual se logra realizando pruebas a la interfaz del producto. Al primer enfoque se le denomina prueba de Caja Blanca y al segundo, prueba de Caja Negra [34].

A continuación se describe el método de Caja Blanca, el cual será utilizado en el desarrollo de las pruebas para el módulo en cuestión. No se tendrá en cuenta en método de Caja Negra debido a que el alcance del software a desarrollar no contempla la implementación de una interfaz gráfica.

Pruebas de Caja Blanca

La prueba de Caja Blanca es un método para el diseño de casos de prueba que usa la estructura de control del diseño procedimental y está dirigido a las funcionalidades internas de un sistema. Entre las técnicas más efectivas dentro de este método se encuentra la técnica del camino básico, la cual permite al diseñador de casos de prueba obtener una medida de la complejidad lógica de un diseño procedimental y usar esa medida como guía para la definición de un conjunto básico de caminos de ejecución. Los casos de prueba obtenidos del conjunto básico, garantizan que durante la prueba se ejecuta al menos una vez cada sentencia del programa [34].

Para la realización del proceso de pruebas se aplica una estrategia incremental por niveles, la cual inicia con las pruebas de unidades individuales, pasa a pruebas diseñadas para facilitar la integración de dichas unidades y culmina con pruebas que se realizan al sistema en general. Además, se verifica la aceptación del cliente para el cual fue concebido el producto final. Las pruebas realizadas al software en cuestión serán a nivel unitario, las cuales se centran en el proceso de verificación en la menor unidad del diseño

del software: el módulo. A continuación se describe cómo se realizó el diseño de los casos de prueba.

3.4.1 Diseño de casos de prueba

El caso de prueba especifica la forma de probar un sistema incluyendo las entradas, salidas y resultados esperados, así como bajo qué condiciones debe probarse el software. Para probar la calidad del módulo, fue realizada una primera etapa de pruebas correspondiente a la iteración inicial del Plan de iteraciones definido, en el cual se contemplan las HU críticas dentro de la aplicación, donde fueron identificadas dos no conformidades asociadas a errores de validación, las cuales fueron solucionadas de forma satisfactoria. Una vez culminada esta etapa, se procede a realizar una segunda fase de pruebas correspondiente a las HU definidas en la segunda iteración, identificándose una no conformidad vinculada con errores de sintaxis e igualmente se le dio solución. Las pruebas aplicadas a cada una de las funcionalidades se realizaron mediante la técnica del camino básico, la cual permitió obtener una medida de la complejidad lógica de un diseño procedimental y usarla como guía para la definición de un conjunto básico de caminos de ejecución, a los cuales se le realizaron los casos de prueba. Para aplicar esta técnica primeramente se enumera el código del método a probar como se muestra en la Figura 5:

```
1 public void editLocalConfigurations(UUID id, ConfigLocalUser configLocal) throws ELogicalServerNotExist {  
2     LogicalServer logicalServer = serverAdministration.getLogicalServer(id);  
3     if (logicalServer != null) {  
4         ILocalUserRepository.editLocalConfigurations(logicalServer, configLocal);  
5     } else {  
6         throw new ELogicalServerNotExist("Logical server not exist");  
7     }  
8 }
```

A partir de dicha enumeración y teniendo en cuenta la notación para las instrucciones, se construye el grafo de flujo (ver Figura 6).

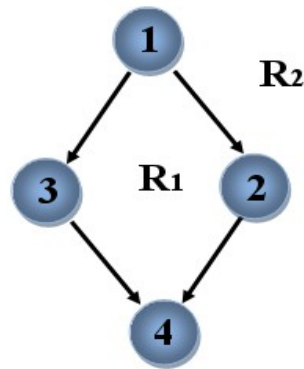


Figura 6: Grafo de flujo.

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

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

$$V(G) = \text{Nodos predicados} + 1 = 1 + 1 = 2$$

$$V(G) = \text{Número de regiones} = 2$$

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 2, 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.

Este número también se corresponde con el valor mínimo de casos de pruebas que deben realizarse teniendo en cuenta el procedimiento escogido. A continuación se realizan los casos de pruebas por cada uno de los caminos identificados.

Camino	Secuencia de nodos
Camino básico # 1	1 – 2 - 4
Camino básico # 2	1 – 3 - 4

Tabla 5: Conjunto de caminos básicos.

Caso de Prueba para el Camino básico # 1	
Nombre de la persona que realiza la prueba: Adrianet Cobo.	Estado de evaluación: Satisfactorio.
Descripción: Para realizar esta prueba se debe introducir dos objetos, uno de tipo UUID, y el otro de tipo ConfigLocalUser. En el caso de no existir un servidor en la lista de LogicalServer, con el UUID pasado por parámetro se lanza una excepción de tipo ElogicalServerNotExist.	
Entradas:	<p>id = 753af260-c804-4ec2-8786-c6c8e93f3528.</p> <p>configLocal = new ConfigLocalUser(true, new Path ("/home"), true, false, new Path ("/etc/vsftpd.chroot_list"), 150, 600).</p> <p>* cada valor pasado por parámetro se corresponde con los siguientes atributos respectivamente: permitir la autenticación de usuarios locales, espacio raíz al que acceden, habilitar el enjaulado de usuarios, habilitar la lista de usuarios que no serán enjaulados, ubicación de ella lista de usuarios que no serán enjaulados, tamaño de cuota y velocidad de transferencia.</p>
Condición de ejecución: El valor de la variable id pasada por parámetro no se corresponde con el UUID de ninguno de los elementos de la lista de LogicalServer.	
Resultado: Se lanza la excepción de tipo ElogicalServerNotExist, por tanto, no se edita el objeto ConfigLocalUser con los nuevos valores pasados por parámetro.	

Tabla 6: Caso de prueba para el camino básico 1.

Caso de Prueba para el Camino básico # 2	
Nombre de la persona que realiza la prueba: Adrianet Cobo.	Estado de evaluación: Satisfactorio.
Descripción: Para realizar esta prueba se debe introducir dos objetos, uno de tipo UUID, y el otro de tipo ConfigLocalUser. En el caso de existir un servidor en la lista de LogicalServer, con el UUID pasado por parámetro, se procede a editar el objeto ConfigLocalUser de ese servidor, con	

los nuevos valores pasados por parámetro.	
Entradas:	id = 643at280-c805-4ed2-8786-c6c8e93f3528.
	<pre>configLocal = new ConfigLocalUser(true, new Path ("/home"), false, false, new Path ("/etc/vsftpd.chroot_list"), 200, 650).</pre> <p>* cada valor pasado por parámetro se corresponde con los siguientes atributos respectivamente: permitir la autenticación de usuarios locales, espacio raíz al que acceden, habilitar el enjaulado de usuarios, habilitar la lista de usuarios que no serán enjaulados, ubicación d ella lista de usuarios que no serán enjaulados, tamaño de cuota y velocidad de transferencia.</p>
Condición de ejecución: El valor de la variable id pasada por parámetro se corresponde con el UUID de alguno de los elementos de la lista de LogicalServer.	
Resultado: Se sustituye el objeto ConfigLocalUser del servidor identificado por la variable configLocal pasada por parámetro.	

Tabla 7: Caso de prueba para el camino básico 2.

3.4.2 Pruebas realizadas por el cliente

Uno de los objetivos principales de las pruebas del software es demostrarle tanto al desarrollador como al cliente que el producto obtenido satisface los requerimientos bajo los cuales fue concebido. Persiguiendo el propósito de darle cumplimiento a este objetivo, una vez culminadas las pruebas de unidad, fueron realizadas pruebas por parte del cliente con el fin de constatar el nivel de aceptación del mismo. Al no contar con una interfaz gráfica, este proceso se realizó mediante la creación de la clase vsftpdPruebas.java, la cual contiene un método *main* que permitió ejecutar llamadas a las funcionalidades del módulo.

3.4.3 Resultados de las pruebas

Buscar los elementos más relevantes de las pruebas es el mayor peso a la hora de analizar los resultados una vez ejecutadas. Un factor importante a tener en cuenta es que a medida que se fue desarrollando el software se fueron aplicando pruebas a nivel del programador, las cuales consisten en ir probando las funcionalidades implementadas con el propósito de verificar su correcto funcionamiento. Este procedimiento permitió obtener un menor número de errores al realizar las pruebas. Una vez culminadas

las etapas de prueba, donde los casos de prueba obtenidos del conjunto básico de caminos de ejecución garantizaron que se ejecutara al menos una vez cada sentencia del programa, se encontraron un total de tres no conformidades, asociadas a errores de validación y de sintaxis, la cuales quedaron resueltas de forma precisa. Por otra parte, en las pruebas realizadas por el cliente no se encontraron problemas significativos que impidieran lograr un alto grado de aceptación por parte del mismo. Por tanto, el software se encuentra listo para ser integrado con el sistema.

3.5 Conclusiones parciales

En el presente capítulo quedaron definidas las iteraciones para realizar la implementación de los requisitos del módulo, estableciendo un orden de prioridad, lo que permitió desarrollar en una primera iteración las HU críticas de la aplicación, para luego desarrollar las restantes. Además se describieron las Tareas de Ingeniería, las cuales facilitaron el trabajo en el desarrollo de la solución, permitiendo un ahorro considerable de tiempo ya que indicaban el camino a seguir para darle cumplimiento a las HU. Una vez desarrolladas las funcionalidades del software, se procedió a la documentación y ejecución de las Pruebas Unitarias basadas en el método de Caja Blanca, con el objetivo de verificar la calidad del producto, obteniéndose resultados satisfactorios. A partir de los resultados de estas pruebas se comprobó que el módulo cumple con el objetivo propuesto, motivo por el cual se encuentra listo para ser integrado a HMAST y ser usado en entornos reales.

Conclusiones

El desarrollo del presente trabajo de diploma permitió elaborar el módulo para la Administración del servicio FTP desde la herramienta HMAST del centro CESOL perteneciente a la UCI, dando cumplimiento a los objetivos trazados, destacándose de manera general los siguientes aspectos:

- Se realizó un estudio de los software dirigidos a realizar la transferencia de archivos en entornos libres, arrojando como resultado la utilización de VsFTPD para realizar la administración del servicio FTP desde la herramienta HMAST, por ser el más completo de los que fueron analizados.
- La correcta utilización de las herramientas, lenguajes y tecnologías descritas, hicieron posible la obtención de un diseño e implementación acertados para el módulo desarrollado, proceso que estuvo guiado sobre la metodología SXP.
- Mediante la implementación de la propuesta de solución se obtuvo un software capaz de realizar configuraciones por tipo de usuario que tienen acceso al servicio FTP, así como configurar parámetros dirigidos a incrementar la seguridad del sistema como son el soporte para SSL, permitir conexiones pasivas y activar la generación de registros. Además, brinda la posibilidad de realizar configuraciones generales entre las que se destacan el establecimiento de la cantidad máxima de intentos fallidos por sesión y la cantidad de usuarios e IP conectados de forma simultánea.
- Los casos de pruebas descritos y efectuados, corroboraron el correcto funcionamiento del software y el cumplimiento de todos los requerimientos definidos para el mismo. Por lo que el módulo, una vez que se le implemente la interfaz gráfica, podrá integrarse sin problemas a HMAST.

Recomendaciones

Una vez cumplido con el objetivo general propuesto, se recomienda:

- Realizar una interfaz gráfica que permita administrar el servicio FTP desde la herramienta HMAST, basada en la descripción del diseño y la implementación realizada en el presente trabajo de diploma.

Referencias bibliográficas

- [1] GÓMEZ FERRERA, MIRIAM CECILIA. A pesar de la inevitable renovación y el carácter consumista en los últimos años de los productores de las Tecnologías de la. In: [online]. [Accessed 22 May 2013]. Available from: <http://www.pedagogiaprofesional.rimed.cu/especial%202/Joe.htm>.
- [2] HERNÁNDEZ SAMUDIO, MARÍA DEL CARMEN. Protocolo FTP. In: [online]. [Accessed 22 May 2013]. Available from: <http://www.slideshare.net/meryetc/dn12-u3-a8mchs>.
- [3] GÓMEZ LABRADOR, RAMÓN. Internet Linux. In: [online]. [Accessed 10 December 2012]. Available from: <http://www.scribd.com/doc/62378381/Internet-Linux>.
- [4] SCIP AG. wu-ftpd hasta 2.6.2 fb_realpath() desbordamiento de búfer. In: [online]. [Accessed 29 May 2013]. Available from: <http://www.scip.ch/es/?vuldb.217>.
- [5] BARRIOS DUEÑAS, JOEL. Cómo configurar pure-ftpd. In: [online]. [Accessed 22 May 2013]. Available from: <http://www.alcancelibre.org/staticpages/index.php/como-pure-ftpd>.
- [6] Pure-FTPd - About. In: [online]. [Accessed 22 May 2013]. Available from: <http://www.pureftpd.org/project/pure-ftpd>.
- [7] *Servicios de red e Internet, FTP* [online]. MADRID GALLEGO, NICOLÁS. [no date]. Available from: <http://nikosri.files.wordpress.com/2012/02/ftp.pdf>.
- [8] Vulnerability analysis and exploit for CVE-2011-4130 in ProFTPd < 1.3.3g/1.3.4 – (Part I) | BugBlog. In: [online]. [Accessed 26 January 2012]. Available from: <http://blog.buguroo.com/?p=3951&lang=en>.
- [9] Giftpd. In: [online]. [Accessed 12 February 2013]. Available from: <http://es.goldenmap.com/Giftpd>.
- [10] vsftpd - Secure, fast FTP server for UNIX-like systems. In: [online]. [Accessed 12 February 2013]. Available from: <https://security.appspot.com/vsftpd.html#people>.
- [11] Tutorial - Instalación Webmin - Google Drive. In: [online]. [Accessed 23 May 2013]. Available from: <https://docs.google.com/document/d/1NAcdhCRVLyQewCqQmsBvym1yeVz-auLEA7nELzQZxaU/edit?hl=es&pli=1>.
- [12] Presentación — Documentación de Zentyal 3.0. In: [online]. [Accessed 18 February 2013]. Available from: <http://doc.zentyal.org/es/presentation.html>.
- [13] 4.1 Características del lenguaje | Curso de Introducción a Java. In: [online]. [Accessed 25 January 2013]. Available from: http://www.mundojava.net/caracteristicas-del-lenguaje.html?Pg=java_inicial_4_1.html.
- [14] UNIDAD 1-Los Lenguajes de Marcas. In: [online]. [Accessed 30 January 2013]. Available from: <http://www.scribd.com/doc/37990287/UNIDAD-1-Los-Lenguajes-de-Marcas>.

- [15] XML. In: [online]. [Accessed 1 January 2013]. Available from: <http://www.hipertexto.info/documentos/xml.htm>.
- [16] Sistemas Computin: junio 2010. In: [online]. [Accessed 23 May 2013]. Available from: http://sistemacomputin.blogspot.com/2010_06_01_archive.html.
- [17] Modelado de Sistemas con UML - Monografias.com. In: [online]. [Accessed 23 May 2013]. Available from: <http://www.monografias.com/trabajos94/modelado-sistemas-uml/modelado-sistemas-uml.shtml>.
- [18] ¿Qué es un Framework? Global Mentoring - Cursos Java Online |. In: [online]. [Accessed 23 May 2013]. Available from: <http://globalmentoring.com.mx/cursos-java/java-frameworks/que-es-un-framework/>.
- [19] BRIANO, FERNANDO. Introducción a Spring Framework Java | Picando Código. In: [online]. [Accessed 23 November 2012]. Available from: <http://picandocodigo.net/2010/introduccion-a-spring-framework-java/>.
- [20] CÓRDOBA, FERNANDO DARÍO. Programación Orientada a Aspectos - Fernando Córdoba. In: [online]. [Accessed 23 November 2012]. Available from: <http://www.slideshare.net/2008PA2Info3/programacin-orientada-a-aspectos-fernando-crdoaba>.
- [21] PEÑALVER ROMERO, GLADYS MARSÍ. *Metodología ágil para proyectos de software libre* [online]. UCI. S.I., 2008.
- [22] Welcome to NetBeans. In: [online]. [Accessed 15 December 2012]. Available from: <http://netbeans.org/>.
- [23] Herramientas CASE para BD - Monografias.com. In: [online]. [Accessed 13 February 2013]. Available from: <http://www.monografias.com/trabajos24/herramientas-case/herramientas-case.shtml#historia>.
- [24] Visual Paradigm for UML (ME) - (Paradigma Visual para UML (ME)) (Visual Paradigm for UML (ME)) por Visual Paradigm International Ltd. - reporte y descarga. In: [online]. [Accessed 7 February 2013]. Available from: http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_%28M%C3%8D%29_14720_p/.
- [25] BAPTISTE, JUAN LUIOS. Kde4 Se Libre. In: [online]. [Accessed 25 May 2013]. Available from: <http://www.slideshare.net/campuspartycolombia/kde4-se-libre>.
- [26] *El proceso unificado de desarrollo de software*. IVAR, Booch, Grady Jacobson, and RUMBAUGH, JAMES. 2004.
- [27] PEÑALVER ROMERO, GLADYS MARSÍ. *Propuesta de un expediente, para los proyectos productivos del Polo de Software Libre, de la Facultad 10*. S.I.: s.n. Página 8
- [28] PEÑALVER ROMERO, GLADYS MARSÍ. *Propuesta de un expediente, para los proyectos productivos del Polo de Software Libre, de la Facultad 10*. S.I.: s.n. Página 9.

- [29] *Guía de Arquitectura N-Capas orientada al Dominio con .NET 4.0*. CÉSAR, Zorrilla Castro and RAMOS BARROSO, MIGUEL ANGEL, Calvarro Nelson, Javier. 2010. 422 p. ISBN 978-84-936696-3-8.
- [30] UML - Lenguaje de Modelado Unificado. In: [online]. [Accessed 24 May 2013]. Available from: <http://kuainasi.ciens.ucv.ve/ads2010-2/uml/index.html#>.
- [31] *UML y Patrones, Introducción al análisis y diseño orientado a objetos*. LARMAN, CRAIG. [no date]. 536 p. ISBN 970-17-0261-1.
- [32] Patrones GoF. In: [online]. [Accessed 24 May 2013]. Available from: <http://geektheplanet.net/5462/patrones-gof.xhtml>.
- [33] PEÑALVER ROMERO, GLADYS MARSÍ. *Propuesta de un expediente, para los proyectos productivos del Polo de Software Libre, de la Facultad 10*. S.l.: s.n. Página 12.
- [34] *Ingeniería del Software, Un enfoque práctico*. PRESSMAN, ROGER S. Sexta Edición. 2005. 900 p. ISBN 9701054733.

Bibliografía

- ¿Qué es un Framework? Global Mentoring - Cursos Java Online |. In: [online]. [Accessed 23 May 2013]. Available from: <http://globalmentoring.com.mx/cursos-java/java-frameworks/que-es-un-framework/>.
- 4.1 Características del lenguaje | Curso de Introducción a Java. In: [online]. [Accessed 25 January 2013]. Available from: http://www.mundojava.net/caracteristicas-del-lenguaje.html?Pg=java_inicial_4_1.html.
- BAPTISTE, JUAN LUIOS. Kde4 Se Libre. In: [online]. [Accessed 25 May 2013]. Available from: <http://www.slideshare.net/campuspartycolombia/kde4-se-libre>.
- BARRIOS DUEÑAS, JOEL. Cómo configurar pure-ftpd. In: [online]. [Accessed 22 May 2013]. Available from: <http://www.alcancelibre.org/staticpages/index.php/como-pure-ftpd>.
- BAUTISTA JIMÉNEZ, JUAN. Notas Arquitectura Marco N-Capas. In: [online]. [Accessed 24 May 2013]. Available from: http://prezi.com/qneoiwlgg_3t/notas-arquitectura-marco-n-capas/.
- BRIANO, FERNANDO. Introducción a Spring Framework Java | Picando Código. In: [online]. [Accessed 23 November 2012]. Available from: <http://picandocodigo.net/2010/introduccion-a-spring-framework-java/>.
- Características del lenguaje Java. In: [online]. [Accessed 25 January 2013]. Available from: <http://www.iec.csic.es/criptonomicon/java/quesjava.html>.
- CÓRDOBA, FERNANDO DARÍO. Programación Orientada a Aspectos - Fernando Córdoba. In: [online]. [Accessed 23 November 2012]. Available from: <http://www.slideshare.net/2008PA2Info3/programacin-orientada-a-aspectos-fernando-crdoaba>.
- *El proceso unificado de desarrollo de software*. IVAR, Booch, Grady Jacobson, and RUMBAUGH, JAMES. 2004.
- Glftpd. In: [online]. [Accessed 12 February 2013]. Available from: <http://es.goldenmap.com/Glftpd>.
- GÓMEZ FERRERA, MIRIAM CECILIA. A pesar de la inevitable renovación y el carácter consumista en los últimos años de los productores de las Tecnologías de la. In: [online]. [Accessed

- 22 May 2013]. Available from: <http://www.pedagogiaprofesional.rimed.cu/especial%202/Joe.htm>.
- GÓMEZ LABRADOR, RAMÓN. Internet Linux. In: [online]. [Accessed 10 December 2012]. Available from: <http://www.scribd.com/doc/62378381/Internet-Linux>.
 - *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.
 - HERNÁNDEZ SAMUDIO, MARÍA DEL CARMEN. Protocolo FTP. In: [online]. [Accessed 22 May 2013]. Available from: <http://www.slideshare.net/meryetc/dn12-u3-a8mchs>.
 - Herramientas CASE para BD - Monografias.com. In: [online]. [Accessed 13 February 2013]. Available from: <http://www.monografias.com/trabajos24/herramientas-case/herramientas-case.shtml#historia>.
 - *Ingeniería del Software, Un enfoque práctico*. PRESSMAN, ROGER S. Sexta Edición. 2005. 900 p. ISBN 9701054733.
 - Lenguaje Unificado de Modelado o UML. In: [online]. [Accessed 23 May 2013]. Available from: <http://resumenuml.bloggratis.es/>.
 - Lenguajes de Marcas en EntornosWeb. In: [online]. [Accessed 22 February 2013]. Available from: <http://www.slideshare.net/ilatorrep/lengmarcasentornosweb>.
 - Linux Six Blog: RapidSVN SVN en Linux. In: [online]. [Accessed 8 February 2013]. Available from: <http://linuxsix.blogspot.com/2011/10/rapidsvn-svn-en-linux.html>.
 - Modelado de Sistemas con UML - Monografias.com. In: [online]. [Accessed 23 May 2013]. Available from: <http://www.monografias.com/trabajos94/modelado-sistemas-uml/modelado-sistemas-uml.shtml>.
 - ORTIZ, HENRY. Guía de Arquitectura en N Capas. In: [online]. [Accessed 24 May 2013]. Available from: <http://guiaarquitecturancapas.blogspot.com/>.
 - PATRONES DE DISEÑO. In: [online]. [Accessed 24 May 2013]. Available from: <http://es.scribd.com/doc/27239031/PATRONES-DE-DISENO>.

- Patrones GoF. In: [online]. [Accessed 24 May 2013]. Available from: <http://geektheplanet.net/5462/patrones-gof.xhtml>.
- PEÑALVER ROMERO, GLADYS MARSÍ. Propuesta de un expediente, para los proyectos productivos del Polo de Software Libre, de la Facultad 10. S.l.: s.n.
- PEÑALVER ROMERO, GLADYS MARSÍ. Metodología ágil para proyectos de software libre [online]. UCI. S.l., 2008.
- Presentación — Documentación de Zentyal 3.0. In: [online]. [Accessed 18 February 2013]. Available from: <http://doc.zentyal.org/es/presentation.html>.
- Pure-FTPd - About. In: [online]. [Accessed 22 May 2013]. Available from: <http://www.pureftpd.org/project/pure-ftp>.
- SCIP AG. wu-ftpd hasta 2.6.2 `fb_realpath()` desbordamiento de búfer. In: [online]. [Accessed 29 May 2013]. Available from: <http://www.scip.ch/es/?vuldb.217>.
- *Servicios de red e Internet, FTP* [online]. MADRID GALLEGO, NICOLÁS. [no date]. Available from: <http://nikosri.files.wordpress.com/2012/02/ftp.pdf>.
- Sistemas Computin: junio 2010. In: [online]. [Accessed 23 May 2013]. Available from: http://sistemacomputin.blogspot.com/2010_06_01_archive.html.
- Tutorial - Instalación Webmin - Google Drive. In: [online]. [Accessed 23 May 2013]. Available from: <https://docs.google.com/document/d/1NAcdhCRVLYQewCqQmsBvym1yeVz-auLEA7nELzQZxaU/edit?hl=es&pli=1>.
- UML - Lenguaje de Modelado Unificado. In: [online]. [Accessed 24 May 2013]. Available from: <http://kuainasi.ciens.ucv.ve/ads2010-2/uml/index.html#>.
- *UML y Patrones, Introducción al análisis y diseño orientado a objetos*. LARMAN, CRAIG. [no date]. 536 p. ISBN 970-17-0261-1.
- UNIDAD 1-Los Lenguajes de Marcas. In: [online]. [Accessed 30 January 2013]. Available from: <http://www.scribd.com/doc/37990287/UNIDAD-1-Los-Lenguajes-de-Marcas>.
- Visual Paradigm for UML (ME) - (Paradigma Visual para UML (ME)) (Visual Paradigm for UML (ME)) por Visual Paradigm International Ltd. - reporte y descarga. In: [online]. [Accessed 7

- February 2013]. Available from: http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_%28M%C3%8D%29_14720_p/.
- vsftpd - Secure, fast FTP server for UNIX-like systems. In: [online]. [Accessed 12 February 2013]. Available from: <https://security.appspot.com/vsftpd.html#people>.
 - Vulnerability analysis and exploit for CVE-2011-4130 in ProFTPD < 1.3.3g/1.3.4 – (Part I) | BugBlog. In: [online]. [Accessed 26 January 2012]. Available from: <http://blog.buguroo.com/?p=3951&lang=en>.
 - Webmin. In: [online]. [Accessed 22 May 2013]. Available from: <http://www.webmin.com/>.
 - Welcome to NetBeans. In: [online]. [Accessed 15 December 2012]. Available from: <http://netbeans.org/>.
 - XML. In: [online]. [Accessed 1 January 2013]. Available from: <http://www.hipertexto.info/documentos/xml.htm>.
 - Zentyal 3.0 Documentación Oficial — Documentación de Zentyal 3.0. In: [online]. [Accessed 22 February 2013]. Available from: <http://doc.zentyal.org/es/>.

Anexo 1: Historias de Usuario.

Historia de Usuario	
Número: HMAST_FTP_1	Nombre Historia de Usuario: Instalar y desinstalar servicio VSFTPD.
Modificación de Historia de Usuario Número: ninguna	
Usuario: Adrianet Cobo Fonseca	Iteración Asignada: 1
Prioridad en Negocio: Alta	Puntos Estimados: 0.5 semanas
Riesgo en Desarrollo: Alto	Puntos Reales: 0.5 semanas
Descripción: El software permite instalar el servicio, así como establecer opciones de configuración predeterminadas antes de instalarlo.	
<p>Observaciones: Al seleccionar la opción de habilitar módulo para el servicio FTP en el menú principal de la herramienta HMAST, se brinda la posibilidad de instalar el servicio VsFTPd. Una vez que el usuario seleccione dicha opción, se comienza a realizar el proceso de instalación mediante la ejecución del comando <i>apt-get install vsftpd</i>. A partir de ese instante se comienza a instalar el servicio y mientras se realiza este proceso se le muestran al usuario una serie de opciones para configurar de manera predeterminada las funcionalidades más importantes del servicio. El software también permite cancelar la instalación en caso de que el usuario así lo desee, así como guardar las configuraciones realizadas, este último no será permitido hasta que no termine de instalarse el servicio. Las opciones predeterminadas que brinda son las siguientes:</p> <p>Habilitar o no autenticación de usuarios locales: Se permite habilitar o deshabilitar la autenticación de usuarios locales. Además en dependencia de si se habilita esta opción, se brinda la posibilidad de establecer el espacio raíz para este tipo de usuarios, en caso de que este parámetro no sea una ruta válida, se muestra un mensaje de error alertando al usuario.</p> <p>Habilitar o no autenticación de usuarios anónimos: Se permite habilitar o deshabilitar la autenticación de usuarios anónimos en el servidor. Además en dependencia de si se habilita esta opción, se brinda la posibilidad de establecer el espacio raíz para este tipo de usuarios, en caso de que este parámetro no sea una ruta válida, se muestra un mensaje de error alertando al usuario.</p>	

Habilitar o no soporte SSL: Se brinda la posibilidad de habilitar o deshabilitar el soporte SSL en el servidor.

Habilitar o no el enjaulado de usuarios: Se permite habilitar o deshabilitar el enjaulado de usuarios. En el caso de habilitar esta opción, por defecto todos los usuarios locales serán enjaulados en el espacio raíz seleccionado previamente.

Activar o no el permiso de escritura: Se brinda la posibilidad de activar o desactivar el permiso de escritura en el servidor.

Establecer máscara de archivos cuando son creados: Se permite escribir una máscara para los archivos cuando son creados por primera vez. En el caso que la máscara no sea válida se muestra un mensaje de error.

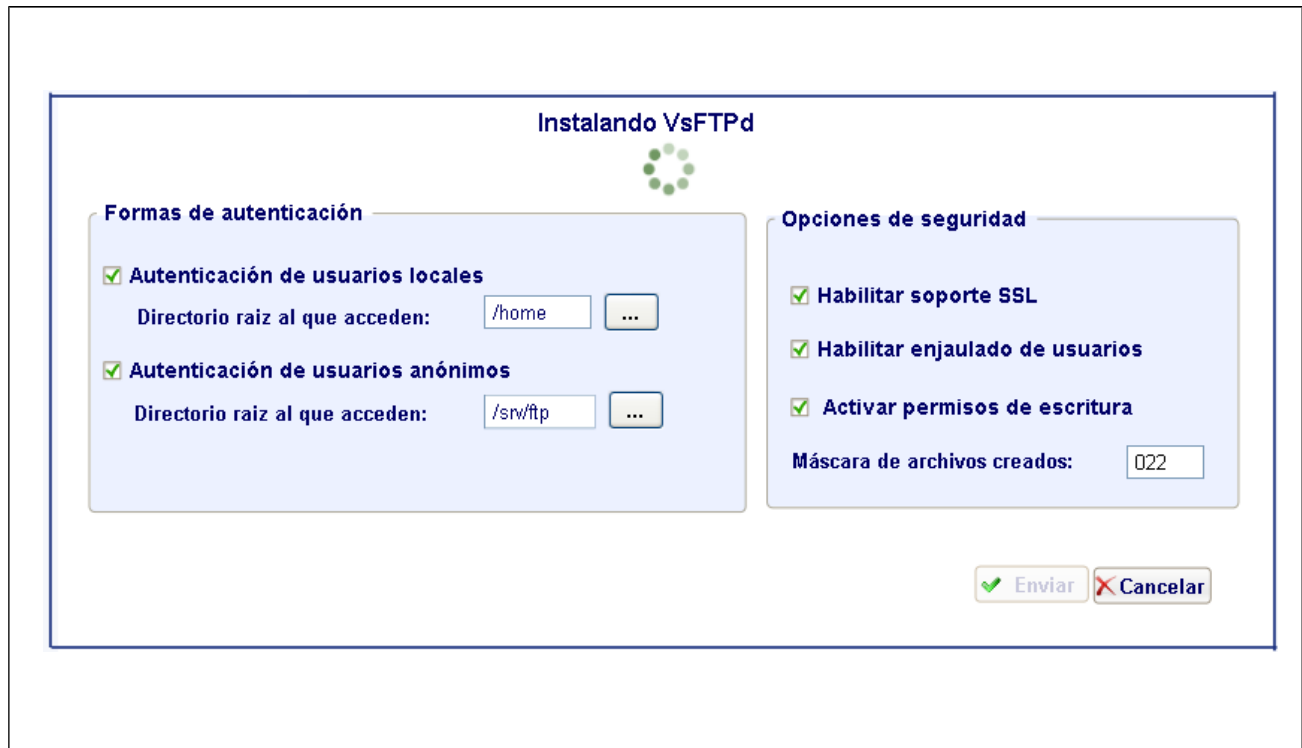
Una vez que el usuario selecciona las configuraciones, este podrá guardar las configuraciones realizadas o cancelarlas en caso de que así lo desee.

- **Desinstalar el servicio:** la aplicación brinda además la posibilidad de desinstalar el servicio si el usuario así lo desea mediante la ejecución del comando `apt-get remove -purge vsftpd`.

Prototipo de interfaz:

Para el funcionamiento del módulo es necesario la instalación del servicio VsFTPd:

Instalar servicio VsFTPd



Historia de Usuario	
Número: HMAST_FTP_2	Nombre Historia de Usuario: Cambiar estado del servicio.
Modificación de Historia de Usuario Número: ninguna	
Usuario: Adrianet Cobo Fonseca	Iteración Asignada: 1
Prioridad en Negocio: Alta	Puntos Estimados: 0.4 semanas
Riesgo en Desarrollo: Medio	Puntos Reales: 0.4 semanas
Descripción: El sistema brinda la posibilidad de Iniciar, Detener, Reiniciar y Recargar el servicio.	
Observaciones: Se brinda la opción de realizar las siguientes acciones: <ul style="list-style-type: none"> • Iniciar el servicio: El software permite realizar el proceso de iniciar el servicio cada vez que el usuario lo desee. Al seleccionar dicha opción el sistema ejecuta el comando <code>service vsftpd start</code>. • Reiniciar el servicio: El software brinda la posibilidad de reiniciar el servicio cada vez que el usuario lo desee. Al seleccionar dicha opción el sistema ejecuta el comando 	

service vsftpd restart.

- **Detener el servicio:** El software permite realizar el proceso de detener el servicio cada vez que el usuario lo desee. Al seleccionar dicha opción el sistema ejecuta el comando service vsftpd stop.
- **Recargar servicio:** Este proceso debe realizarse de forma automática cuando se envían las configuraciones contempladas en una misma interfaz, ejecutándose el comando service vsftpd reload.

Prototipo de interfaz:



Historia de Usuario	
Número: HMAST_FTP_3	Nombre Historia de Usuario: Realizar configuración de usuarios locales en el servidor.
Modificación de Historia de Usuario Número: ninguna	
Usuario: Adrianet Cobo Fonseca	Iteración Asignada: 1
Prioridad en Negocio: Alta	Puntos Estimados: 3.1 semanas
Riesgo en Desarrollo: Alto	Puntos Reales: 3.1 semanas
Descripción: El software es capaz de permitir o denegar la autenticación de los usuarios locales. En caso de que se permita, entonces el sistema brinda la posibilidad de realizar una serie de configuraciones como son: establecer, mostrar y modificar el directorio raíz por defecto cuando se autentican en el servidor, establecer un sistema de salvas para este tipo de usuarios, así como limitar y mostrar el tamaño de la cuota, adicionar o eliminar usuarios de la lista que muestra los usuarios locales que no serán enjaulados en su espacio raíz y establecer, mostrar y modificar el valor de la velocidad de transferencia de los datos para este tipo de usuarios, así como establecer y modificar el espacio personal para usuarios específicos, definiendo además los permisos que tendrán sobre el mismo.	
Observaciones: La opción de permitir o no la autenticación de usuarios locales en el servidor se especifica mediante la directiva local_enable. Se le brinda la posibilidad al cliente de activar o	

desactivar este parámetro.

- **Permitir autenticación de usuarios locales:** Al seleccionar la opción de habilitar, el valor de la directiva es YES y se permite la autenticación de usuarios locales.
- **Denegar autenticación de usuarios locales:** En caso de seleccionar la opción para deshabilitar, el valor de la directiva es NO y por lo tanto se deniega la autenticación de usuarios locales.

El directorio por defecto al cual acceden los usuarios locales, es su propio directorio raíz en el sistema /home, lo cual se especifica mediante la directiva local_root. Es necesario validar que la ruta sea válida, en caso de no ser válida se muestra un mensaje de error.

- **Establecer directorio raíz de los usuarios locales:** En un primer momento los usuarios locales tienen definido por defecto el directorio que fue seleccionado en el proceso de instalación.
- **Mostrar ruta del directorio raíz para usuarios locales:** Se muestra al usuario, la ruta que ha sido establecida para el directorio raíz.
- **Modificar directorio raíz para usuarios locales:** Una vez mostrada la ruta del directorio raíz de los usuarios locales, el usuario puede modificarla, teniendo en cuenta que la nueva ruta debe ser válida, en caso contrario se muestra un mensaje de error.

La directiva chroot_list_enable permite especificar una lista con los usuarios locales a los cuales no se les enjaulará cuando la opción chroot_local_user = YES.

- **Habilitar o deshabilitar el enjaulado de usuarios locales en su espacio raíz:** Se brinda la posibilidad de enjaular o no a determinados usuarios locales en su espacio raíz.

Una vez habilitado el enjaulado de usuarios se procede a gestionar los usuarios enjaulados.

- **Crear lista de usuarios no enjaulados:** El sistema crea una lista, la cual está ubicada por defecto en /etc/vsftpd/chroot_list.
- **Mostrar ubicación de la lista de usuarios no enjaulados:** El software permite mostrar dicha ruta.
- **Modificar ruta de la lista de usuarios no enjaulados:** Una vez mostrada la ruta, se brinda la posibilidad de modificarla. Es necesario validar la ruta, en caso de no ser válida se muestra un mensaje de error.
- **Adicionar usuarios locales a la lista:** El software permite adicionar usuarios a lista de

usuarios que no serán enjaulados, para esto es necesario validar que no se hay adicionado el mismo usuario anteriormente y que sea un usuario válido del sistema.

- **Eliminar usuarios locales de la lista:** Se brinda la posibilidad de eliminar usuarios de dicha lista, validando el usuario que se desea eliminar se encuentre en ella.

La máxima velocidad de transferencia de datos se especifica mediante la directiva `local_max_rate`, para usuarios locales. El valor por defecto de esta directiva es 0 lo que no limita el ratio de transferencia y la unidad de medida del valor de la velocidad que toma esta directiva está dada en bytes por segundo.

- **Establecer máxima velocidad de transferencia de datos:** El sistema permite establecer la velocidad máxima de transferencia para usuarios autenticados, la cual será un número entero. En caso que el valor que se introduzca no sea válido se muestra un mensaje de error.
- **Mostrar velocidad máxima de transferencia de datos:** Se muestra al usuario la velocidad máxima de transferencia de datos para usuarios autenticados.
- **Modificar velocidad máxima de transferencia de datos:** Una vez mostrada la velocidad máxima de transferencia de datos para usuarios autenticados, se le brinda la posibilidad al usuario de modificar este parámetro, siempre y cuando, el nuevo valor sea válido.

La gestión del sistema de salvos será manejada por el paquete `quota` y `quotatool`.

- **Establecer sistema de salvos para usuarios autenticados:** Se le brinda al usuario la posibilidad de establecer un sistema de salvos. En el caso de que se desee activar el sistema de cuota, el software es instala los paquetes `quota` y `quotatool`, mediante el comando `apt-get install -y quota quotatool`. Una vez instalados, se procede a modificar el fichero ubicado en `/etc/fstab` indicando en él la partición que tendrá el sistema de cuotas (se toma el directorio raíz del sistema). Luego se remonta la partición usando el comando: `mount -o remount /ruta/de/la/partición`. Posteriormente se inicializa el sistema de cuotas usando el comando: `quotacheck -avugm`, al realizar este paso se muestra un aviso, el cual no resulta un problema, ya que su origen es porque como es la primera vez que se inicia el sistema de cuotas se crean los ficheros `aquota.user` y `aquota.group`. El próximo paso es activar las cuotas mediante la orden: `quotaon /ruta/de/la/partición`. A continuación se crean las cuotas para los usuarios, usando el comando: `edquota -u`

usuario, al ejecutar este comando, se abrirá con el editor de textos predeterminado en el sistema, lo que permite configurar la cantidad de espacio disponible a cada usuario en su directorio personal. La cuota se especifica en kilo bytes y por defecto el sistema establece 100 Mb que serían 100000 kb. Una vez realizado este paso se ejecuta: `getent passwd | cut -d : -f1 > logins`, el cual coloca en el fichero `logins` todos los usuarios del sistema, luego se ejecuta el comando: `for x in `cat logins` ; do edquota -p usuario $x;done`, de esta forma todos los usuarios tendrán la misma cuota.

- **Gestionar sistema de cuotas:** El software permite gestionar las cuotas para los usuarios autenticados en el sistema, teniendo en cuenta las siguientes funcionalidades:
 - **Establecer tamaño de cuota para todos los usuarios:** Se brinda la posibilidad de establecer una cuota general para todos los usuarios del sistema. Debe ser un número mayor que 0 y menor que la capacidad del disco duro de la máquina, en caso de que el valor no se encuentre en el rango definido se muestra un mensaje de error.
 - **Mostrar el tamaño de la cuota:** Se muestra al usuario el tamaño de la cuota asignada, lo cual está reflejado en un campo numérico.
 - **Modificar el tamaño de la cuota para todos los usuarios:** Una vez mostrado el tamaño de la cuota, se le brinda la posibilidad al usuario de modificar este parámetro, teniendo en cuenta que el nuevo valor se corresponda con las restricciones.
 - **Editar cuota para usuario específico:** Se brinda la posibilidad de permite modificar el tamaño de la cuota para un usuario determinado. Es necesario tener en cuenta que dicho usuario forme parte de la lista de los que se le aplicará el sistema de cuotas.
 - **Adicionar usuario al sistema de cuota:** La aplicación permite adicionar usuarios locales a la lista de usuarios a los que se le aplicará el sistema de cuotas.
 - **Eliminar usuario del sistema de cuota:** El software brinda la posibilidad de eliminar usuarios de la lista de aquellos usuarios que no serán incluidos en el sistema de cuotas.

El servicio también brinda la posibilidad de cambiarle el espacio al cual acceden cuando se autentican a usuarios locales específicos y qué permisos tendrán sobre el mismo. Con el propósito de lograr este objetivo la directiva `user_config_dir` especifica un directorio donde son almacenados archivos para cada usuario que se desee configurar donde se especifican la carpeta a la cual tienen acceso y los permisos sobre la misma. Para realizar este proceso la aplicación permite realizar las siguientes funcionalidades:

- **Adicionar usuarios con acceso a directorio específico:** Se brinda la posibilidad de adicionar una nueva configuración para los usuarios locales a los que se les desee realizar una configuración específica, posibilitando establecer el directorio raíz al cual acceden, así como los permisos que tendrán sobre dicho directorio.
- **Editar usuario:** Se permite modificar las configuraciones personalizadas realizadas a usuarios específicos. Brindando la posibilidad de modificar los parámetros a tener en cuenta que son el espacio al cual acceden cuando se autentican en el servidor, así como los permisos que tienen sobre el mismo.
- **Eliminar usuario:** Se brinda la posibilidad de eliminar las configuraciones realizadas para usuarios específicos.

Una vez realizadas todas las configuraciones la aplicación brinda la posibilidad de guardar los cambios o cancelarlos.

Prototipo de interfaz:

The screenshot shows a web-based administration interface for FTP. It is divided into two main sections: 'Configuraciones generales' and 'Usuarios con directorios específicos'.

Configuraciones generales:

- Habilitar autenticación de usuarios locales**
- Directorio raíz al que acceden por defecto:
- Habilitar enjaulado de usuarios**
- Habilitar sistema de cuotas**
- Velocidad de transferencia de datos:

Usuarios con directorios específicos:

Buttons: **Mostrar:**

Usuarios	Directorios
<input checked="" type="checkbox"/> juan	/Datos
<input type="checkbox"/> adry	/home/ftp
<input type="checkbox"/> cristina	
<input type="checkbox"/> elio	

Navigation: << 1 2 3 >>

Gestionar usuarios enjaulados:

Enjaular usuarios locales

Lista de usuarios no enjaulados: ...

Mostrar: 5

Usuarios	No enjaular
reidiel	<input type="checkbox"/>
pablo	<input type="checkbox"/>
cobo	<input type="checkbox"/>
prueba	<input checked="" type="checkbox"/>
	<input type="checkbox"/>

<< 1 2 >>

Gestionar sistema de cuotas:

Gestionar cuotas para usuarios

Tamaño de cuota general:

Cuotas específicas para usuarios

Mostrar: 5

Usuarios	Cuota
<input type="checkbox"/> reidiel	
<input type="checkbox"/> pablo	
<input type="checkbox"/> adry	
<input type="checkbox"/> prueba	
<input type="checkbox"/>	

<< 1 2 >>

Adicionar usuarios con acceso a directorio específico:

Adicionar usuarios con acceso a directorio específico

Directorio: ...

usuario Mostrar: 3 ▾

Usuarios	Escritura	Lectura	Descargar
<input type="checkbox"/> pepe	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> user1	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> jose	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

<< 1 2 3 >>

Editar con configuraciones específicas:

Editar usuario elio:

Cambiar carpeta compartida ...

Permiso de escritura

Permiso de lectura

Permiso para descargar archivos

Historia de Usuario	
Número:HMAST_FTP_4	Nombre Historia de Usuario: Realizar configuración acceso de usuarios anónimos al servidor.
Modificación de Historia de Usuario Número: ninguna	
Usuario: Adrianet Cobo Fonseca	Iteración Asignada: 1
Prioridad en Negocio: Alta	Puntos Estimados: 1.5 semanas
Riesgo en Desarrollo: Medio	Puntos Reales: 1.5 semanas
<p>Descripción: El software es capaz de permitir o denegar la autenticación de usuarios anónimos en el servidor, en caso de que se permita, entonces el sistema brinda la posibilidad de realizar una serie de configuraciones como son: establecer, mostrar y modificar el espacio al cual tienen acceso en el servidor, otorgar o denegar permisos a usuarios anónimos y establecer, mostrar y modificar el valor de la velocidad de transferencia de los datos para este tipo de usuarios.</p>	
<p>Observaciones: Las opciones de permitir el la autenticación o no de usuarios anónimos en el servidor se especifica mediante la directiva anonymous_enable.</p> <ul style="list-style-type: none"> • Permitir autenticación de usuarios anónimos: El software permite habilitar la autenticación de usuarios locales, al seleccionar esta opción, el valor de la directiva es YES, este es el valor que toma por defecto. • Denegar autenticación de usuarios anónimos: En caso de que no se seleccione la opción el valor de la directiva es NO y por lo tanto se deniega la autenticación de usuarios anónimos. <p>La directiva para especificar el espacio al cual acceden los usuarios anónimos es anon_root. Es necesario validar que la ruta sea correcta, en caso que no sea válida se muestra un mensaje de error al usuario.</p> <ul style="list-style-type: none"> • Establecer directorio al que acceden los usuarios anónimos: En un primer momento los usuarios anónimos tienen definido por defecto el /srv/ftp como directorio al que acceden en el servidor. • Mostrar ruta del directorio raíz para usuarios anónimos: Se muestra al usuario, la ruta que ha sido establecida para el directorio raíz. • Modificar directorio raíz para usuarios anónimos: Una vez mostrada la ruta del directorio raíz de los usuarios anónimos, el usuario puede modificarla, teniendo en cuenta 	

que la nueva ruta sea válida.

Los usuarios anónimos, debido a su condición tienen permisos restringidos en el servidor. Con el propósito de otorgarle o no dichos permisos la aplicación permite realizar las siguientes funcionalidades:

- **Permitir a los usuarios anónimos descargar archivos con permisos de lectura:** Se brinda la opción de permitirle a los usuarios anónimos descargar aquellos archivos que tengan permisos de lectura. Si se selecciona esta opción, entonces la directiva `anon_world_readable_only` toma valor YES, siendo este el valor que toma por defecto.
- **Denegar a los usuarios anónimos descargar archivos con permisos de lectura:** Se brinda la opción de denegarle a los usuarios anónimos el permiso de descargar aquellos archivos que tengan permisos de lectura. En este caso la directiva `anon_world_readable_only` toma valor NO.
- **Permitir a los usuarios anónimos subir archivos:** Se brinda la posibilidad de permitirle a los usuarios anónimos subir archivos al servidor, lo cual hace que la directiva `anon_upload_enable` tome valor YES.
- **Denegar a los usuarios anónimos subir de archivos:** Se brinda la posibilidad de denegarle a los usuarios anónimos el permiso de subir archivos al servidor, lo cual hace que la directiva `anon_upload_enable` tome valor NO, este es el valor por defecto.
- **Permitir a los usuarios anónimos crear directorios:** Se brinda la opción de permitirle a los usuarios anónimos crear directorios en el servidor. Si se selecciona esta opción, la cual es mostrada en un checkbox, entonces la directiva `anon_mkdir_write_enable` toma valor YES.
- **Denegar a los usuarios anónimos crear directorios:** Se brinda la posibilidad de denegarle a los usuarios anónimos el permiso de crear directorios, lo cual hace que la directiva `anon_mkdir_write_enable` tome valor NO, este es el valor por defecto.

La máxima velocidad de transferencia de datos se especifica mediante la directiva `anon_max_rate`, para usuarios anónimos. El valor por defecto de esta directiva es 0 lo que no limita el ratio de transferencia y la unidad de medida del valor de la velocidad que toma esta directiva está dada en bytes por segundo.

- **Establecer máxima velocidad de transferencia de datos:** El sistema establece por

defecto la velocidad máxima de transferencia para usuarios anónimos, la cual será un número entero mayor que 0.

- **Mostrar velocidad máxima de transferencia de datos:** Se muestra al usuario la velocidad máxima de transferencia de datos para usuarios anónimos.
- **Modificar velocidad máxima de transferencia de datos:** Una vez mostrada la velocidad máxima de transferencia de datos para usuarios autenticados, se le brindará la posibilidad al usuario de modificar este parámetro, teniendo en cuenta el nuevo valor sea correcto.

Una vez realizadas todas las configuraciones la aplicación brinda la posibilidad de guardar los cambios o cancelarlos.

Prototipo de interfaz:

Configuración de los usuarios anónimos

Habilitar autenticación de usuarios anónimos

Directorio raíz al que acceden por defecto:

Velocidad de transferencia:

Gestionar permisos

Descargar archivos Subir archivos Crear archivos

Historia de Usuario	
Número: HMAST_FTP_5	Nombre Historia de Usuario: Configurar conexiones SSL.
Modificación de Historia de Usuario Número: ninguna	
Usuario: Adrianet Cobo Fonseca	Iteración Asignada: 1
Prioridad en Negocio: Alta	Puntos Estimados: 1.5 semanas
Riesgo en Desarrollo: Medio	Puntos Reales: 1.5 semanas
Descripción: El software permite habilitar o deshabilitar las conexiones SSL y en caso de que se habilite permite realizar una serie de configuraciones así como gestionar los certificados ssl.	
Observaciones: La directiva que permite establecer las conexiones SSL es ssl_enable. El valor por defecto para esta directiva es NO.	
<ul style="list-style-type: none"> • Habilitar conexiones SSL: Se permite habilitar el soporte para SSL, de esta forma la directiva ssl_enable toma valor YES. • Deshabilitar conexiones SSL: El usuario puede deshabilitar el soporte SSL, lo que hace que la directiva ssl_enable tome valor NO. 	
Existen una serie de configuraciones regidas por diferentes directivas para configurar la conexión SSL.	
<ul style="list-style-type: none"> • Obligar o no, el establecimiento de conexiones encriptadas para usuarios locales: Se brinda la posibilidad de seleccionar o no la opción de establecer conexiones encriptadas para usuarios locales. En correspondencia con esta operación la directiva force_local_logins_ssl toma valor YES o NO. El valor por defecto para esta directiva es YES. • Obligar o no, a que las transferencias de datos locales sean encriptados con SSL: Se brinda la posibilidad de seleccionar o no la opción de establecer que las transferencias de datos locales sean encriptados con SSL. En correspondencia con esta operación la directiva force_local_data_ssl tomará valor YES o NO. El valor por defecto para esta directiva es YES. • Habilitar o deshabilitar soporte para las diversas versiones de SSL: Hasta el momento solo existen 3 versiones de SSL, por lo que la aplicación brinda la posibilidad de habilitar una o las versiones que desee el usuario. Las directivas que permiten 	

realizar esta operación de forma efectiva son `ssl_tlsv1`, `ssl_sslv2`, `ssl_sslv3`, las cuales se corresponden con cada una de las versiones de SSL y toman valor YES o NO, en dependencia de la selección que realice el usuario. Los valores por defecto de estas directivas son YES para `ssl_tlsv1` y NO para las dos restantes.

Mediante la ejecución del comando: `openssl req -new -x509 -nodes -out` (ubicación de los certificados) `-keyout` (ubicación de la clave pública), se generan los certificados ssl. La directiva encargada definir la ubicación de los certificados generados es `rsa_cert_file` y `rsa_private_key_file` para la ubicación de ella clave pública.

- **Generar certificados SSL por defecto:** Se le permite al usuario la opción de generar el certificado por defecto.
- **Subir certificado SSL y clave pública:** El usuario tiene la posibilidad de en lugar de generar el certificado por defecto, poder subirlo, al igual que su clave pública.
- **Establecer ubicación por de los certificados SSL:** Se establece como ubicación por defecto `/etc/ssl/vsftpd/vsftpd.pem`. Es necesario validar que la ruta sea correcta. En caso de no ser válida se muestra un mensaje de error.
- **Mostrar ubicación del certificado SSL:** Se muestra al usuario la ubicación de los certificados SSL, en un campo de texto.
- **Modificar la ubicación del certificado SSL:** Una vez que sea mostrada la ubicación de los certificados, se brinda la posibilidad de modificarla, teniendo en cuenta que la nueva ruta sea válida.
- **Establecer ubicación para de la clave del certificado SSL:** Se permite establecer la ruta de la clave pública para esto es necesario validar que la ruta sea correcta. En caso de no ser válida se muestra un mensaje de error.
- **Mostrar ubicación de la clave pública:** Se muestra al usuario la ubicación de la clave pública, en un campo de texto.
- **Modificar la ubicación de la clave pública:** Una vez que sea mostrada la ubicación de la clave pública, se brinda la posibilidad de modificarla, es necesario tener en cuenta que la nueva ruta sea válida.

Una vez realizadas todas las configuraciones la aplicación brinda la posibilidad de guardar los cambios o cancelarlos.

- **Desactivar conexiones pasivas:** Se le permite al usuario desactivar las conexiones pasivas.
- **Establecer rango de puertos para conexiones pasivas:** Se le permite al usuario establecer el rango de puertos para las conexiones pasivas. Se debe validar que el rango de números en el componente visual para el número de puerto, esté entre 0 y 65535. Además comprobar que los números seleccionados no coincidan con puertos que estén siendo usados, teniendo en cuenta también, que existen puertos que están siendo usados por otros servicios.
- **Mostrar rango de puertos para conexiones pasivas:** Se muestra el rango de puertos para conexiones pasivas.
- **Modificar rango de puerto para conexiones pasivas:** Una vez mostrado el rango de puertos para conexiones pasivas, se permite modificar los mismos, teniendo en cuenta que los nuevos valores introducidos para los puertos mínimo y máximo sean adecuados.

Una vez realizadas todas las configuraciones la aplicación brinda la posibilidad de guardar los cambios o cancelarlos.

Prototipo de interfaz:



Conexiones Pasivas

Habilitar conexiones en modo pasivo

Puerto mínimo: Puerto máximo:

Historia de Usuario	
Número: HMAST_FTP_7	Nombre Historia de Usuario: Realizar configuraciones generales.
Modificación de Historia de Usuario Número: ninguna	
Usuario: Adrianet Cobo Fonseca	Iteración Asignada: 1
Prioridad en Negocio: Media	Puntos Estimados: 0.3 semanas
Riesgo en Desarrollo: Medio	Puntos Reales: 0.3 semanas
<p>Descripción: El software permite activar o desactivar el modo de escritura en el servidor. Es capaz de establecer, modificar y mostrar la máscara que tendrán los archivos cuando son creados, así como gestionar el banner de bienvenida, que no es más que el texto que se muestra cuando el usuario se autentifica en el servidor. También permite activar o desactivar los mensajes de directorios. Brinda la posibilidad de configurar el tiempo máximo de espera para mantener conexiones inactivas y conexiones de datos inactivas. Permite activar o desactivar la ejecución del servidor en modo independiente (standalone). Además establece cantidad máxima permitida de intentos de acceso fallidos por sesión y clientes e IP conectados de forma simultánea.</p>	
<p>Observaciones: La directiva encargada activar o desactivar el modo de escritura en el servidor es write_enable.</p> <ul style="list-style-type: none"> • Activar modo de escritura: Se brinda al usuario la posibilidad de activar el modo de escritura en el servidor, en este caso la directiva correspondiente toma valor YES. • Desactivar el modo de escritura: Se brinda al usuario la posibilidad de desactivar el modo de escritura en el servidor, en este caso la directiva correspondiente toma valor NO. <p>La especificación de la máscara de los archivos cuando son creados en el servidor se realiza mediante la directiva local_umask. Los permisos que se le pueden establecer a directorios son de lectura, escritura y ejecución los cuales son reflejados en un número de tres dígitos donde el primer especifica los permisos del propietario, el segundo el de los grupos y el tercero el del resto de los usuarios. A su vez cada número es una combinación binaria que activa el bit según el permiso que se desee establecer, ejemplo:</p>	

Código	Binario	Permisos efectivos
0	000	- - -
1	001	- - x
2	010	- w -
3	011	- w x
4	100	r - -
5	101	r - x
6	110	r w -
7	111	r w x

Un ejemplo de máscara sería 777 donde tanto propietario como grupos y otros tienen todos los permisos.

- **Establecer máscara por defecto de los archivos cuando son creados:** El sistema establece por defecto la máscara 022.
- **Mostrar máscara de los archivos cuando son creados:** Se le muestra al usuario la máscara establecida para los archivos cuando son creados.
- **Modificar máscara de los archivos cuando son creados:** Una vez mostrada la máscara que tendrán los archivos después de su creación, se brinda la posibilidad de modificarla, teniendo en cuenta que la nueva máscara sea válida.

La directiva que permite gestionar el banner de bienvenida es `ftpd_banner`.

- **Establecer banner de bienvenida:** Se le brinda la posibilidad al usuario de escribir el texto que desea que se muestre cuando los usuarios se autentifiquen en el servidor en un campo de texto.
- **Mostrar banner de bienvenida:** Se muestra el texto del banner de bienvenida.
- **Modificar banner de bienvenida:** Una vez que se muestra el texto del banner de bienvenida el usuario puede modificarlo si así lo desea, escribiendo el mismo en el campo de texto donde se muestra.

La directiva para especificar el tiempo máximo para mantener conexiones inactivas es `idle_session_timeout` y la que define el tiempo máximo de conexiones de datos inactivas es `data_connection_timeout`. El sistema es capaz de establecer por defecto 300 segundos para dichas directivas.

- **Establecer tiempo máximo para mantener conexiones inactivas:** Se le permite al usuario establecer el valor de este parámetro, el cual tiene que ser un número entero entre 0 y 86400, un equivalente a 1 día, tiempo máximo permitido para mantener conexiones inactivas. En caso que el valor que se introduzca no sea válido se muestra un mensaje de error.
- **Mostrar tiempo máximo para mantener conexiones inactivas:** Se muestra al usuario el tiempo máximo para mantener conexiones inactivas.
- **Modificar tiempo máximo para mantener conexiones inactivas:** Una vez mostrado el valor de este parámetro, se le brinda la posibilidad al usuario de modificar este parámetro, teniendo en cuenta que el nuevo valor cumpla con las restricciones establecidas.
- **Establecer tiempo máximo para mantener conexiones de datos inactivas:** Se le permite al usuario establecer el valor de este parámetro el cual tiene que ser un número entero entre 0 y 86400, un equivalente a 1 día, tiempo máximo permitido para mantener aquellas conexiones que realicen transferencias de datos sin progreso. En caso que el valor que se introduzca no sea válido se muestra un mensaje de error.
- **Mostrar tiempo máximo para mantener conexiones de datos inactivas:** Se muestra al usuario el tiempo máximo para mantener conexiones de datos inactivas.
- **Modificar tiempo máximo para mantener conexiones de datos inactivas:** Una vez mostrado el valor de este parámetro, se le brinda la posibilidad al usuario de modificar este parámetro, teniendo en cuenta que el nuevo valor sea correcto.

La ejecución del servidor en modo independiente se especifica mediante la directiva `listen`. El sistema establece el valor YES por defecto para esta directiva.

- **Activar la ejecución en modo independiente:** Se le brinda al usuario la opción de ejecutar el servidor en modo independiente.
- **Desactivar la ejecución en modo independiente:** El usuario puede desactivar la ejecución del servidor en modo independiente.

La cantidad máxima de clientes simultáneos conectados en el servidor se especifica mediante la directiva `max_clients`. El valor por defecto que se establece es 0 lo que no limita las conexiones.

- **Establecer la cantidad máxima permitida de clientes conectados simultáneamente:**

Se le permite al usuario establecer el valor de este parámetro el cual tiene que ser un número entero entre 0 y 50000, valor máximo permitido. En caso que el valor que se introduzca no sea válido se muestra un mensaje de error.

- **Mostrar la cantidad máxima permitida de clientes conectados simultáneamente:** Se muestra la cantidad máxima permitida de usuarios conectados de forma simultánea en el servidor.
- **Modificar la cantidad máxima permitida de clientes conectados simultáneamente:** Una vez mostrado el valor de dicho parámetro, el usuario puede modificarlo, teniendo en cuenta que el nuevo valor se corresponda con las restricciones establecidas.

La cantidad máxima de IP simultáneos conectados en el servidor se especifica mediante la directiva `max_per_ip`. El valor por defecto que se establece es 0 lo que no limita las conexiones.

- **Establecer la cantidad máxima permitida de IP conectados simultáneamente:** Se le permite al usuario establecer el valor de este parámetro el cual tiene que ser un número entero entre 0 y 50000, valor máximo permitido. En caso que el valor que se introduzca no sea válido se muestra un mensaje de error.
- **Mostrar la cantidad máxima permitida de IP conectados simultáneamente:** Se muestra la cantidad máxima permitida de IP conectados de forma simultánea en el servidor, en el propio campo donde quedó establecida.
- **Modificar la cantidad máxima permitida de IP conectados simultáneamente:** Una vez mostrado el valor de dicho parámetro, el usuario puede modificarlo, teniendo en cuenta que el nuevo valor sea correcto.

La especificación de la cantidad máxima de intentos fallidos por sesión lo realiza la directiva `max_login_fails`. El valor por defecto que establece el sistema es 3.

- **Establecer cantidad máxima permitida de intentos fallidos por sesión:** Se le permite al usuario establecer la cantidad máxima permitida de intentos fallidos por sesión. El valor tiene que ser un número entero entre 0 y 10. En caso que el valor que se introduzca no sea válido se muestra un mensaje de error.
- **Mostrar cantidad máxima permitida de intentos fallidos por sesión:** Se muestra la cantidad máxima permitida de intentos fallidos por sesión.
- **Modificar la cantidad máxima permitida de intentos fallidos por sesión:** Una vez

mostrado el valor de dicho parámetro el usuario podrá modificarlo si así lo desea, teniendo en cuenta que el nuevo valor debe ser válido.

Una vez realizadas todas las configuraciones la aplicación brinda la posibilidad de guardar los cambios o cancelarlos.

Prototipo de interfaz:

The screenshot shows a configuration window titled "Configuraciones" with a light blue background. It contains several settings:

- Activar permisos de escritura
- Activar mensajes de bienvenida
- Mensaje:
- Ejecutar el servidor en modo independiente
- Máscara de los archivos cuando son creados:
- Tiempo máximo para conexiones inactivas : min
- Tiempo máximo para conexiones de datos inactivas : min
- Cantidad máxima de clientes conectados:
- Cantidad máxima de IP conectados:
- Cantidad máxima de intentos fallidos por sesión:

At the bottom right, there are two buttons: "Enviar" with a green checkmark icon and "Cancelar" with a red X icon.

Historia de Usuario	
Número: HMAST_FTP_8	Nombre Historia de Usuario: Configurar registros en el servidor.
Modificación de Historia de Usuario Número: Ninguna	
Usuario: Adrianet Cobo Fonseca	Iteración Asignada: 2
Prioridad en Negocio: Alta	Puntos Estimados: 0.7 semanas
Riesgo en Desarrollo: Medio	Puntos Reales: 0.7 semanas
<p>Descripción: El software permite activar o desactivar la generación de logs en el servidor y si está activado permite establecer, mostrar y modificar la ubicación de los registros en el servidor.</p> <p>Observaciones: La generación de logs en el servidor se especifica mediante la directiva xferlog_enable. El valor por defecto de esta directiva es NO.</p> <ul style="list-style-type: none"> • Activar la generación de logs: Se permite al usuario activar la generación de registros en el servidor. • Desactivar la generación de logs: En caso de que esté activada la generación de logs, el usuario puede desactivarla. <p>La directiva que se encarga de especifica la ubicación de los registros en el servidor es vsftpd_log_file.</p> <ul style="list-style-type: none"> • Establecer ubicación de los logs: El software definirá por defecto que los logs se encontrarán en /var/log/vsftpd.log. Es necesario validar que la ruta sea correcta, en caso de no serlo se muestra un mensaje de error. • Mostrar ubicación de los logs: Se mostrará al usuario la ubicación de los registros, mediante un campo de texto. • Modificar la ubicación de los logs: Una vez que se muestre la ubicación de los logs en el servidor, el usuario podrá modificarla si lo desea, teniendo en cuenta que la nueva ruta sea válida. <p>Una vez realizadas todas las configuraciones la aplicación brinda la posibilidad de guardar los cambios o cancelarlos.</p>	

Prototipo de interfaz:

The image shows a software configuration window titled "Configuraciones". It contains a checked checkbox labeled "Activar la generación de registros". Below this is a label "Fichero de ubicación de los registros:" followed by a text input field containing the path "/etc/ftp/vsftpd.log" and a button with three dots "...". At the bottom right of the window are two buttons: "Enviar" with a green checkmark icon and "Cancelar" with a red X icon.

Anexo 2: Tareas de Ingeniería.

Tarea de Ingeniería para la HU Instalar servicio VsFTPD:

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: HMAST_FTP_1
Nombre Tarea: Probar el comando para realizar la instalación del servidor.	
Tipo de Tarea : Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Adrianet Cobo Fonseca	
Descripción: Se ejecuta el comando apt-get install vsftpd, el cual tiene como finalidad la instalación del servidor vsftpd. Posteriormente se comprueba su correcto funcionamiento.	

Tarea de Ingeniería para la HU Gestionar servicio:

Tarea de Ingeniería	
Número Tarea: 2	Número Historia de Usuario: HMAST_FTP_2
Nombre Tarea: Probar los comandos que permiten Iniciar, Detener, Recargar y Reiniciar el servicio.	
Tipo de Tarea : Desarrollo	Puntos Estimados: 0.2
Fecha Inicio:	Fecha Fin:
Programador Responsable: Adrianet Cobo Fonseca	
Descripción: Los comandos que permiten gestionar el servicio son los siguientes: - Iniciar: service vsftpd start - Detener: service vsftpd stop - Reiniciar: service vsftpd restart - Recargar: service vsftpd upload. Se ejecutan cada uno de estos comandos y se comprueban los resultados.	

Tareas de Ingeniería para la HU Gestionar configuración de usuarios locales en el servidor:

Tarea de Ingeniería	
Número Tarea: 3	Número Historia de Usuario: HMAST_FTP_3
Nombre Tarea: Comprobar el funcionamiento de la directiva que permite la autenticación de usuarios locales.	
Tipo de Tarea : Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Adrianet Cobo Fonseca	
Descripción: El fichero de configuración del servidor VsFTPd, ubicado en /etc/vsftpd/vsftpd.conf, cuenta con la directiva local_enable, la cual, en dependencia del valor que se le asigne, permite o no, la autenticación de usuarios locales en el servidor. Los posibles valores que puede tomar son YES o NO. Es necesario probar esta directiva en los posibles valores que pueda tomar, con el fin de comprobar su correcto funcionamiento.	

Tarea de Ingeniería	
Número Tarea: 4	Número Historia de Usuario: HMAST_FTP_3
Nombre Tarea: Comprobar el funcionamiento de la directiva que permite establecer el directorio raíz de usuarios locales.	
Tipo de Tarea : Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Adrianet Cobo Fonseca	
Descripción: El fichero de configuración del servidor VsFTPd, ubicado en /etc/vsftpd/vsftpd.conf, cuenta con la directiva local_root, la cual permite definir el fichero al cual tendrán acceso los usuarios locales en el servidor. En caso de que esta directiva tome un valor inválido, o sea, una ruta que no se encuentre en el sistema de archivos del servidor, al autenticarse los usuarios locales tendrán acceso al sistema raíz.	

Tarea de Ingeniería	
Número Tarea: 5	Número Historia de Usuario: HMAST_FTP_3
Nombre Tarea: Comprobar el funcionamiento de las directivas que permiten establecer el enjaulado de usuarios locales en el espacio raíz definido.	
Tipo de Tarea : Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Adrianet Cobo Fonseca	
<p>Descripción: El fichero de configuración vsftpd.conf, cuenta con tres directivas que tienen como finalidad enjaular en su espacio raíz a los usuarios locales seleccionado. La directiva chroot_local_user acepta dos valores YES o NO, si está activada, a los usuarios locales se les cambia el directorio raíz (se hace un chroot) a su directorio principal luego de la conexión. Por otro lado la directiva chroot_list_file especifica el archivo que contiene una lista de los usuarios locales que no serán enjaulados en su espacio raíz y por último la directiva chroot_list_enable permite habilitar o no la lista especificada en la directiva chroot_list_file y su funcionamiento depende de que la opción chroot_local_user tome valor YES.</p>	

Tarea de Ingeniería	
Número Tarea: 6	Número Historia de Usuario: HMAST_FTP_3
Nombre Tarea: Comprobar el funcionamiento de la directiva que permiten establecer la velocidad de transferencia para usuarios locales.	
Tipo de Tarea : Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Adrianet Cobo Fonseca	
<p>Descripción: El fichero de configuración vsftpd.conf, cuenta con la directiva local_max_rate, la cual especifica la velocidad de transferencia de datos para los usuarios locales en bytes de segundo. En caso de no establecer ningún valor y dejar comentariada esta directiva, la velocidad es ilimitada.</p>	

Tarea de Ingeniería	
Número Tarea: 7	Número Historia de Usuario: HMAST_FTP_3
Nombre Tarea: Comprobar el funcionamiento de las directivas que permiten establecer configuraciones específicas para determinados usuarios locales.	
Tipo de Tarea : Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Adrianet Cobo Fonseca	
Descripción: El fichero de configuración vsftpd.conf, cuenta con la directiva user_config_dir, la cual especifica la ruta a un directorio que contiene los archivos de configuración con los nombres de los usuarios locales. Contiene información específica sobre ese usuario como son el directorio al cual tiene acceso, así como los permisos sobre el mismo. Cualquier directriz en el archivo de configuración del usuario ignora aquellas encontradas en vsftpd.conf.	

Tarea de Ingeniería	
Número Tarea: 8	Número Historia de Usuario: HMAST_FTP_3
Nombre Tarea: Comprobar el funcionamiento de los paquetes quota y quotatool.	
Tipo de Tarea : Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Adrianet Cobo Fonseca	
Descripción: Se instala el paquete quota y quotatool y se realizan diferentes configuraciones con el fin de comprender, teniendo en cuenta los comandos necesarios con el fin de comprobar su funcionamiento. Esto permite establecer una cuota, ya sea general para todos los usuarios locales en su espacio raíz o específica un tamaño de cuota para usuarios determinados. En el caso de que no se desee que algún usuario se contemple en dicho sistema de cuotas, se puede eliminar del mismo. Los comandos empleados son los siguientes: - apt-get install -y quota quotatool: Para instalar los paquetes.	

- quotacheck -avugm: Para inicializar el sistema de cuotas.
- quotaon /home: Para activar las cuotas.
- edquota -u *usuario*: Al ejecutar este comando, se abrirá con el editor de textos predeterminado en el sistema, la utilidad que permitirá configurar la cantidad de espacio que dejará disponible a cada usuario en su directorio personal. Esta utilidad mostrará en la sección blocks la cuota utilizada por el usuario, inicialmente en cero y en hard, la cuota que tendrán los usuarios, en Kilo bytes, por lo que, por ejemplo, si quiere darles un espacio de 5 MB, debe colocar 5000. Un inconveniente es que si tiene muchos usuarios tendría que repetir este comando muchas veces, una para cada usuario. Para esto, puede tomar un usuario como base, por ejemplo, el usuario, *usuario*.
- getent passwd | cut -d: -f1 > logins: Este comando se ejecuta una vez configurada la cuota para el usuario *usuario* (tomado de ejemplo), se volcará en un fichero de texto todos los usuarios del sistema. Fichero del cual se pueden eliminar los usuarios que no se desee incluir en el sistema de cuotas.
- for x in `cat logins`; do edquota -p *usuario* \$x;done: Este comando se ejecuta con el propósito de asignarle la misma cuota a todos los usuarios locales que se encuentran en el fichero logins.
- /etc/init.d/quota restart: Comando para reiniciar su sistema de cuotas.

Tareas de Ingeniería para la HU Gestionar configuraciones de usuarios anónimos al servidor.

Tarea de Ingeniería	
Número Tarea: 9	Número Historia de Usuario: HMAST_FTP_5
Nombre Tarea: Comprobar el funcionamiento de la directiva para permitir o no la autenticación de usuarios anónimos en el servidor.	
Tipo de Tarea : Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Adrianet Cobo Fonseca	
Descripción: El fichero de configuración vsftpd.conf, cuenta con la directiva anonymous	

_enable la cual, en dependencia del valor que se le asigne, permite o no, la autenticación de usuarios anónimos en el servidor. Los posibles valores que puede tomar son YES o NO. Es necesario probar esta directiva en los posibles valores que pueda tomar, con el fin de comprobar su correcto funcionamiento.

Tarea de Ingeniería	
Número Tarea: 10	Número Historia de Usuario: HMAST_FTP_5
Nombre Tarea: Comprobar el funcionamiento de la directiva para establecer directorio al cual acceden los usuarios anónimos en el servidor.	
Tipo de Tarea : Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Adrianet Cobo Fonseca	
Descripción: El fichero de configuración vsftpd.conf cuenta con la directiva anon_root, la cual permite definir el fichero al cual tendrán acceso los usuarios locales en el servidor. En caso de que esta directiva tome un valor inválido, o sea, una ruta que no se encuentre en el sistema de archivos del servidor o esta directiva se encuentre comentariada, al autenticarse los usuarios anónimos tendrán acceso al directorio ubicado en srv/ftp.	

Tarea de Ingeniería	
Número Tarea: 11	Número Historia de Usuario: HMAST_FTP_5
Nombre Tarea: Comprobar el funcionamiento de las directivas para configurar los permisos de los usuarios anónimos en el servidor.	
Tipo de Tarea : Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Adrianet Cobo Fonseca	
Descripción: Debido a la condición de este tipo de usuario que como su nombre lo indica son	

anónimos, es necesario restringirles los permisos en el servidor con el propósito de lograr una mayor seguridad. Las directivas encargadas de esto son:

- anon_world_readable_only: Solo acepta dos valores YES o NO, en caso de que esté activa los usuarios anónimos solamente pueden descargar archivos legibles por todo el mundo.
- anon_upload_enable: Solo acepta dos valores YES o NO, en caso de que esté activa los usuarios anónimos pueden cargar archivos al directorio padre que tiene permisos de escritura.
- anon_mkdir_write_enable: Solo acepta dos valores YES o NO, en caso de que esté activa los usuarios anónimos pueden crear nuevos directorios dentro de un directorio que tiene permisos de escritura.

Tarea de Ingeniería	
Número Tarea: 12	Número Historia de Usuario: HMAST_FTP_5
Nombre Tarea: Comprobar el funcionamiento de la directiva que permiten establecer la velocidad de transferencia para usuarios anónimos.	
Tipo de Tarea : Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Adrianet Cobo Fonseca	
Descripción: El fichero de configuración vsftpd.conf, cuenta con la directiva anon_max_rate, la cual especifica la velocidad de transferencia de datos para los usuarios anónimos en bytes de segundo. En caso de no establecer ningún valor y dejar comentariada esta directiva, la velocidad es ilimitada.	

Tareas de Ingeniería para la HU Configurar conexiones SSL:

Tarea de Ingeniería	
Número Tarea: 13	Número Historia de Usuario: HMAST_FTP_6
Nombre Tarea: Comprobar el funcionamiento de las directivas para configurar las conexiones SSL.	

Tipo de Tarea : Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Adrianet Cobo Fonseca	
<p>Descripción: El fichero de configuración vsftpd.conf cuenta con una serie de directivas que permiten configurar las conexiones SSL en el servidor, las cuales son:</p> <ul style="list-style-type: none"> - ssl_enable: Permite dos valores YES o NO y su propósito es activar el soporte para conexiones SSL. - force_local_logins_ssl: Permite dos valores YES o NO, en caso de que esté activa, se obliga el establecimiento de conexiones encriptadas mediante SSL para usuarios locales. - force_local_data_ssl: Permite dos valores YES o NO, en caso de estar activa, se obliga a que las transferencias de datos locales sean encriptados con SSL. - ssl_tlsv1, ssl_sslv2, ssl_sslv3: Permiten dos valores YES o NO y se usan para habilitar los soportes en las diferentes versiones de SSL. 	

Tarea de Ingeniería	
Número Tarea: 14	Número Historia de Usuario: HMAST_FTP_6
Nombre Tarea: Comprobar el funcionamiento de las directivas y comandos para gestionar certificados SSL.	
Tipo de Tarea : Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Adrianet Cobo Fonseca	
<p>Descripción: El fichero vsftpd.conf cuenta con la directiva rsa_cert_file la cual se encarga de especificar la ruta del certificado en el servidor y la directiva rsa_private_key_file la cual especifica la ruta de la llave pública. En caso de que se desee generar un certificado por defecto, se ejecuta el siguiente comando openssl req -new -x509 -nodes -out (ubicación del certificado) -keyout (ubicación de la clave).</p>	

Tarea de Ingeniería para la HU Configurar conexiones pasivas:

Tarea de Ingeniería	
Número Tarea: 15	Número Historia de Usuario: HMAST_FTP_7
Nombre Tarea: Comprobar el funcionamiento de las directivas para configurar las conexiones pasivas.	
Tipo de Tarea : Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Adrianet Cobo Fonseca	
Descripción: El fichero vsftpd.conf cuenta con tres directivas para configurar las conexiones en modo pasivo del servidor: <ul style="list-style-type: none">- pasv_enable: Esta directiva solo soporta dos valores YES o NO, en caso de estar activa se permiten conexiones en modo pasivo.- pasv_min_port: Especifica el puerto más bajo posible para los clientes FTP para las conexiones en modo pasivo. Esta configuración es utilizada para limitar el intervalo de puertos para que las reglas del cortafuego sean más fáciles de implementar.- pasv_max_port: Especifica el puerto más alto posible enviado a los clientes FTP para las conexiones en modo pasivo. Esta configuración es utilizada para limitar el intervalo de puertos para que las reglas del cortafuego sean más fáciles de crear. El valor no puede exceder de 65535.	

Tarea de Ingeniería para la HU Gestionar configuraciones generales:

Tarea de Ingeniería	
Número Tarea: 16	Número Historia de Usuario: HMAST_FTP_8
Nombre Tarea: Comprobar el funcionamiento de las directivas relacionadas con las configuraciones generales del servicio.	

Tipo de Tarea : Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Adrianet Cobo Fonseca	
<p>Descripción: El fichero de configuración vsftpd.conf cuenta con una serie de directivas encaminadas a realizar las configuraciones generales del servicio:</p> <ul style="list-style-type: none"> - write_enable: Soporta dos valores YES o NO y en caso de que esté activa se permite modificar el sistema de archivos. - local_umask: Especifica el valor de umask para la creación de archivos. - ftpd_banner: Si está activada, se muestra la cadena de caracteres especificada en esta directriz cuando se establece una conexión con el servidor. - idle_session_timeout: Especifica la cantidad máxima de tiempo entre comandos desde un cliente remoto. Una vez disparado, se cierra la conexión al cliente remoto. - data_connection_timeout: Especifica la cantidad máxima de tiempo que las conexiones se pueden aplazar en segundos. Una vez lanzado, se cierra la conexión con el cliente remoto. - listen: Cuando está activada, vsftpd se ejecuta en modo independiente. - max_clients: Especifica el número máximo de clientes simultáneos que tienen permitido conectarse al servidor cuando se ejecuta en modo independiente. Cualquier conexión adicional resultará en un mensaje de error. - max_per_ip: Especifica el máximo número de clientes que tienen permitido conectarse desde la misma dirección IP fuente. - max_login_fails: Especifica el máximo número de intentos de acceso fallidos por sesión. 	

Tarea de Ingeniería para la HU Gestionar configuración de registros en el servidor:

Tarea de Ingeniería	
Número Tarea: 17	Número Historia de Usuario: HMAST_FTP_9
Nombre Tarea: Comprobar el funcionamiento de las directivas para la configuración de los registros del servicio.	

Tipo de Tarea : Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Adrianet Cobo Fonseca	
Descripción: El fichero de configuración vsftpd.conf cuenta con una serie de directivas que tienen como finalidad configurar los logs del servicio: <ul style="list-style-type: none">- xferlog_enable: Soporta dos valores YES o NO y cuando se activa, vsftpd registra las conexiones, solamente formato vsftpd.- vsftpd_log_file: Especifica el archivo de registro de vsftpd. Para que se utilice este archivo, xferlog_enable debe estar activado.	

Glosario de términos

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

DHCP: Protocolo de configuración de *host* dinámico. Es un protocolo que permite que un equipo conectado a una red pueda obtener su configuración de red en forma dinámica.

TCP/IP: Son las siglas de Protocolo de Control de Transmisión/Protocolo de Internet (en inglés *Transmission Control Protocol/Internet Protocol*), un sistema de protocolos que hacen posibles servicios Telnet, FTP, E-mail, y otros entre ordenadores que no pertenecen a la misma red.

Sistemas operativos GNU/Linux: El sistema operativo Linux junto con las utilidades del proyecto GNU constituyen lo que se conoce como GNU/Linux.

Demonios : En sistemas UNIX se conoce como demonio o *daemon (Disk And Execution Monitor)* a un proceso que se ejecuta en segundo plano del sistema operativo, se ejecuta en todo momento y no posee interacción directa con el usuario, también se le conoce genéricamente como servicio o proceso, del cual no se percibe su ejecución. Un demonio realiza una operación específica en tiempos predefinidos o en respuesta a ciertos eventos del sistema. El origen del nombre se refiere a su función de "vigilante" y no a una connotación negativa como se podría pensar en un principio.

Modelo Cliente-Servidor: La arquitectura Cliente-Servidor permite al usuario en una máquina, llamada el cliente, requerir algún tipo de servicio de una máquina a la que está unido, llamado el servidor, mediante una red como una LAN (Red de Área Local) o una WAN (Red de Área Mundial). Estos servicios pueden ser peticiones de datos de una base de datos, de información contenida en archivos o los archivos en sí mismos, o peticiones de imprimir datos en una impresora asociada.

Telnet: Es el acrónimo de *Telecommunication Network*. Se trata del nombre de un protocolo de red que se utiliza para acceder a una computadora y manejarla de forma remota. El término también permite nombrar al programa informático que implementa el cliente.

SSL/TLS: SSL son las siglas en inglés de *Secure Socket Layer* (en español Capa de Conexión Segura). Es un protocolo criptográfico empleado para realizar conexiones seguras entre un cliente. TLS son las siglas en inglés de *Transport Layer Security* (en español Seguridad en la Capa de Transporte). Es un protocolo criptográfico empleado en redes.

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.

directivas: Norma o conjunto de normas e instrucciones que dirigen, guían u orientan una acción.

buffer: Área de almacenamiento temporal de información.

HTML: siglas de *HyperText Markup Language* («lenguaje de marcado hipertextual»), hace referencia al lenguaje de marcado predominante para la elaboración de páginas web que se utiliza para describir y traducir la estructura y la información en forma de texto, así como para complementar el texto con objetos tales como imágenes.

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

OpenSSL: Consiste en un robusto paquete de herramientas de administración y bibliotecas relacionadas con la criptografía, que suministran funciones criptográficas a otros paquetes como OpenSSH y navegadores web (para acceso seguro a sitios HTTPS).

Cortafuegos: En su traducción al español significa muro de fuego o cortafuego y es un sistema de defensa basado en el hecho de que todo el tráfico de entrada o salida a la red debe pasar obligatoriamente por un sistema de seguridad capaz de autorizar, denegar, y tomar nota de todo aquello que ocurre, de acuerdo con una política de control de acceso entre redes.

Apache: El servidor HTTP Apache es un servidor web HTTP de código abierto, para plataformas Unix, Microsoft Windows, Macintosh y otras, que implementa el protocolo HTTP.

servidores virtuales: Es como tener un servidor de Internet pero creado mediante un procedimiento software, que simula un servidor que no existe físicamente (como máquina independiente), que ofrece las mismas prestaciones que uno real.

Inet/xnetd: Conocidos como “súper servidores”, cuya función principal es escuchar a través de los puertos de red configurados a la espera de peticiones de los demonios del sistema y, a la vez, realizar comprobaciones de seguridad antes de aceptar dichas conexiones.

Pool: Es un término en inglés que se traduce al español como aliar, aunar, combinar, poner en un fondo

común.

Unix: es un sistema operativo multitarea y multiusuario, lo cual significa que puede ejecutar varios programas simultáneamente, y que puede gestionar a varios usuarios de forma concurrente.

Encriptación: Es el proceso mediante el cual cierta información o texto es cifrado de forma que el resultado sea ilegible a menos que se conozcan los datos necesarios para su interpretación.

usuarios virtuales: es aquel usuario que se conecta a distancia y tiene características y necesidades específicas.

certificados SSL: Sirve para brindar seguridad al visitante de su página web, una manera de decirles a sus clientes que el sitio es auténtico, real y confiable para ingresar datos personales.

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

depurador: Es un programa usado para probar y eliminar los errores.