

**Universidad de las Ciencias Informáticas**  
**Facultad 2**



**Título: Plataforma de Gestión de Servicios Telemáticos en GNU/Linux.**

**Módulo de DHCP v1.0**

Trabajo de Diploma para optar por el título de  
Ingeniero Informático

**Autor(es):** Osmani Acosta Cantero.

**Tutor(es):** Ing. Ramón Alexander Anglada Martínez.

**Co-tutor:** Ing. Álvaro Luis Padilla Moya

Junio de 2011

DECLARACIÓN DE AUTORÍA

Declaro que soy el único autor de este trabajo y autorizo al Centro de Telemática de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Para que así conste firmo la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del año \_\_\_\_\_.

Osmani Acosta Cantero

---

Ing. Ramón Alexander Anglada Martínez

---

## DATOS DE CONTACTO

**Ing. Ramón Alexander Anglada Martínez:** Formó parte del proyecto CTAISC desde el segundo año de su carrera universitaria asumiendo el rol de programador. Presentó varios trabajos en la JCE. Organizó varios festivales de Software Libre en la universidad y centros educacionales en la provincia Ciudad Habana. Cumplió misión internacionalista en Venezuela como parte del proyecto productivo CTAISC. Fue candidato al premio del rector en las categorías de estudiante más destacado del año 2007 y mejor profesor en proyecto del año 2009. Actualmente es líder del proyecto Servicios Telemáticos, cursa la maestría Telemática y es Jefe del Dpto. de Seguridad Informática del Centro de Telemática.

## AGRADECIMIENTOS

A mi mamá Rosa María y a mi papá Osmany por darme la vida, la enseñanza y los valores que hoy forman parte de mi personalidad, por dejarme escoger mi camino, y darme la confianza y el apoyo durante estos cinco años de carrera, por el ejemplo que siempre han sido para mí. A mis hermanos Osbel y Orelvis que son una parte importante de mi vida y siempre me han apoyado, siendo verdaderos hermanos en todo momento.

A mi novia Dilonyx por su amor y cariño, por enseñarme a ser mejor persona y por estar siempre a mi lado brindándome su apoyo y compañía, por ser mi compañera de la vida. A su familia Vikky, Espino, María y Dailenis por recibirme como un miembro más, por todo el apoyo que me han brindado en todos estos años de carrera.

A mi tía Esmildre (mima y familia) por su apoyo desde mi etapa pioneril, por ser la tutora de toda mi vida profesional, los por consejos y recomendaciones que me ha brindado, y mis tíos Tati, Nena y Toly y sus familias que siempre han estado al tanto y se han preocupado por mi carrera. A mi tío Narciso (chino) por estar constantemente al tanto de mí y de mi familia, por brindarme todo el apoyo que ha estado a su alcance.

A mis tutores Álvaro y Anglada por todo el esfuerzo realizado para que salga este trabajo de diploma, porque han sido líderes, compañeros y amigos. Y a todos los integrantes del proyecto Servicios Telemáticos que han estado presentes a lo largo de mi carrera profesional.

A mis amigos de la carrera Wilfredo y Alberto que siempre me han apoyado y son como hermanos para mí, y a todos mis amigos de contramaestre entre ellos Maykel, Julio, Yanier, Luis Carlos y David.

A mis tíos Aleida y Vega que a lo largo de sus vidas siempre me ofrecieron un apoyo incondicional.

A mi tía Odalis y su familia por estar siempre que los necesito y brindarme su hogar como si fuese el mío. A mi abuelo Orlando e Irene, a mi tío Willin y su esposa Flavia (Cosi que hoy no se encuentra entre nosotros) que desde pequeño he compartido con ellos como familia y vecinos.

A mis padrinos Máximo y Noelia que desde pequeño me han apoyado y están al tanto de mí.

Y todos los profesores y compañeros que han formado parte de mi carrera y a todas las personas que de una forma u otra han estado presentes y me han apoyado en mi vida y mi carrera estudiantil.

DEDICATORIA

A mis padres y hermanos

A mi tia Esmildre.

A mi novia Dilonyx.

## RESUMEN

Cuba es un país en vías de desarrollo que ha optado por lograr una soberanía tecnológica. Tomando una serie de medidas entre las que se encuentran el uso de software libre para su desarrollo económico, por lo que se hace necesario una migración completa de los servicios telemáticos que se encuentran en sus empresas, instituciones, organismos y centros, ya que estos en su gran mayoría usan software propietario, dándole gran dependencia al país con las grandes empresas productoras de estos, además la economía se ve afectada con la compra de las licencias a elevados precios para el uso de los mismos y esto sin contar que Cuba es un país bloqueado al que se le imposibilita el acceso a muchas de estas licencias.

En el complicado proceso de migración de los servicios de red se encuentra uno de los mayores problemas que es el uso de consolas de comando para la gestión de las configuraciones de los servidores libres, haciendo difícil y complejo el trabajo de los administradores de red ya que estos tienen que dominar todas las estructuras y términos que se utilizan en los ficheros de configuración. Además son escasas las herramientas que facilitan el proceso de migración de estos servicios sobre sistemas libres y que ofrezcan una administración de manera profesional e intuitiva, en especial el servicio DHCP (Protocolo de Configuración Dinámica de Host).

Por estas razones el presente trabajo de diploma tiene como objetivo desarrollar una aplicación de escritorio que permita administrar de forma agradable y práctica el servidor DHCP de los sistemas GNU/Linux. Esta aplicación permitirá al administrador de red gestionar las configuraciones del servidor a través de interfaces visuales, además de importar las configuraciones de un DHCP de Windows Server 2003, facilitando el proceso de migración.

## PALABRAS CLAVE

Servidor DHCP, proceso de migración, Software Libre, soberanía tecnológica, servicios telemáticos.

TABLA DE CONTENIDOS

|   |    |
|---|----|
| AGRADECIMIENTOS .....                               | I  |
| DEDICATORIA.....                                    | I  |
| RESUMEN .....                                       | II |
| TABLA DE CONTENIDOS .....                           | I  |
| INTRODUCCIÓN .....                                  | 5  |
| CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA .....            | 9  |
| 1.1    Introducción.....                            | 9  |
| 1.2    Soluciones existentes en el mundo. ....      | 9  |
| 1.2.1    Microsoft Windows NT Server. ....          | 9  |
| 1.2.2    Microsoft Windows Server 2003.....         | 10 |
| 1.2.3    SuSE Linux Enterprise Server.....          | 11 |
| 1.2.4    Red Hat Enterprise Linux 5 Server. ....    | 12 |
| 1.2.5    Webmin.....                                | 13 |
| 1.3    Lenguaje de modelado.....                    | 13 |
| 1.4    Metodología de desarrollo.....               | 14 |
| 1.5    Modelación de funciones mediante IDEF0. .... | 16 |
| 1.6    Lenguaje de programación Python.....         | 17 |
| 1.7    Interfaz gráfica Qt Designer.....            | 18 |
| 1.8    Entorno de desarrollo EasyEclipse.....       | 18 |
| 1.9    Herramienta case Visual Paradigm .....       | 19 |
| 1.10    Conclusiones .....                          | 19 |
| CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA .....       | 20 |

|         |  |    |
|---------|--|----|
| 2.1     | Introducción.....  | 20 |
| 2.2     | Problema y situación problemática .....                            | 20 |
| 2.3     | Información que se maneja .....                                    | 20 |
| 2.4     | Propuesta de sistema.....  | 20 |
| 2.5     | Modelo de negocio.....   | 21 |
| 2.5.1   | Representación de los procesos con IDEF0.....                      | 21 |
| 2.5.1.1 | Diagrama A-0.....  | 21 |
| 2.5.1.2 | Diagrama A0 .....  | 22 |
| 2.5.2   | Descripción de los procesos del negocio.....                       | 23 |
| 2.5.2.1 | Descripción de procesos del diagrama A-0. ....                     | 23 |
| 2.5.2.2 | Descripción de procesos del diagrama A0. ....                      | 23 |
| 2.6     | Especificación de los requisitos de software.....                  | 25 |
| 2.6.1   | Requisitos Funcionales. ....                                       | 26 |
| 2.6.2   | Requerimientos no funcionales .....                                | 29 |
| 2.7     | Modelo de Casos de Uso del Sistema.....                            | 31 |
| 2.7.1   | Definición de los actores del sistema a automatizar .....          | 31 |
| 2.7.2   | Diagrama de Casos de Uso del Sistema. ....                         | 32 |
| 2.7.3   | Especificación de los Casos de Uso.....                            | 33 |
| 2.7.3.1 | Descripción del Casos de Uso Gestionar Subred.....                 | 33 |
| 2.7.3.2 | Descripción del Casos de Uso GestionarGrupo .....                  | 41 |
| 2.7.3.3 | Descripción del Casos de Uso GestionarRed Compartida.....          | 42 |
| 2.7.3.4 | Descripción del Casos de Uso GestionarHost .....                   | 42 |
| 2.7.3.5 | Descripción del Casos de Uso Gestionar Opciones y Parámetros ..... | 43 |
| 2.7.3.6 | Descripción del Casos de Uso Nuevo Servidor Remoto.....            | 43 |



|   |  |    |
|---|--|----|
| 2.7.3.7                                   | Descripción del Casos de Uso Preparar integración DHCP con DNS .....   | 43 |
| 2.7.3.8                                   | Descripción del Casos de Uso Mostrar Reportes .....                    | 44 |
| 2.7.3.9                                   | Descripción del Casos de Uso Exportar Configuraciones.....             | 45 |
| 2.7.3.10                                  | Descripción de Casos de Uso Importar Configuraciones.....              | 45 |
| 2.8                                       | Conclusiones:.....   | 48 |
| CAPÍTULO 3: DISEÑO DEL SISTEMA.....       |  | 49 |
| 3.1                                       | Introducción.....  | 49 |
| 3.2                                       | Modelo Diseño .....  | 49 |
| 3.2.1                                     | Estructura detallada del Sistema .....                                 | 50 |
| 3.3                                       | Diagrama de clases del diseño.....                                     | 51 |
| 3.3.1                                     | Diagrama de Clases del diseño. CU Gestionar Subred .....               | 51 |
| 3.3.2                                     | Diagrama de Clases del diseño. CU Gestionar Host .....                 | 51 |
| 3.3.3                                     | Diagrama de Clases del diseño. CU Gestionar Red Compartida.....        | 52 |
| 3.3.4                                     | Diagrama de Clases del diseño. CU Gestionar Grupo.....                 | 52 |
| 3.3.5                                     | Diagrama de Clases del diseño. CU Gestionar Servidor .....             | 53 |
| 3.3.6                                     | Diagrama de Clases del diseño. CU Gestionar Opciones y Parámetros..... | 53 |
| 3.3.7                                     | Diagrama de Clases del diseño. CU Importar Configuraciones .....       | 54 |
| 3.3.8                                     | Diagrama de Clases del diseño. CU Exportar Configuraciones.....        | 54 |
| 3.3.9                                     | Diagrama de Clases del diseño. CU Mostrar Reportes .....               | 55 |
| 3.4                                       | Conclusiones.....  | 55 |
| CAPÍTULO 4: IMPLMANTACIÓN Y PRUEBAS ..... |  | 56 |
| 4.1                                       | Introducción.....  | 56 |
| 4.2                                       | Diagrama de Despliegue.....  | 56 |
| 4.2.1                                     | Descripción de los nodos del diagrama dedespliegue.....                | 56 |

|         |   |    |
|---------|---|----|
| 4.2.2   | Diagrama de despliegue.....                                   | 56 |
| 4.3     | Diagramas de Componentes.....                                 | 56 |
| 4.3.1   | Diagrama General de Componentes .....                         | 57 |
| 4.3.2   | Diagrama de Componentes. CU Gestionar Subred.....             | 57 |
| 4.3.3   | Diagrama de Componentes. CU Gestionar Host.....               | 58 |
| 4.3.4   | Diagrama de Componentes. CU Gestionar Grupo.....              | 58 |
| 4.3.5   | Diagrama de Componentes. CU Gestionar Red Compartida.....     | 58 |
| 4.3.6   | Diagrama de Componentes. CU Exportar Configuraciones.....     | 59 |
| 4.3.7   | Diagrama de Componentes. CU Gestionar Servidores Remotos..... | 59 |
| 4.3.8   | Diagrama de Componentes. CU Importar Configuraciones.....     | 59 |
| 4.3.9   | Diagrama de Componentes. CU Mostrar Estadísticas.....         | 60 |
| 4.4     | Pruebas.....  | 60 |
| 4.4.1   | Diseño del caso de prueba.....                                | 62 |
| 4.4.1.1 | Prototipo de interfaz gestionar Subred.....                   | 62 |
| 4.4.1.2 | Secciones y Escenarios del Caso de Uso Gestionar Subred.....  | 63 |
| 4.4.1.3 | Descripción de las variables.....                             | 63 |
| 4.4.1.4 | Sección eliminar subred.....                                  | 63 |
| 4.4.1.5 | Sección adicionar subred y modificar subred.....              | 64 |
| 4.4.1.6 | Registro de defectos y dificultades detectados.....           | 67 |
| 4.5     | Conclusiones.....   | 67 |
|         | CONCLUSIONES.....   | 68 |
|         | RECOMENDACIONES .....   | 69 |
|         | BIBLIOGRAFÍA .....  | 70 |

## INTRODUCCIÓN

La informática en los días actuales tiene una estrecha relación entre el hardware y software, estos son de vital importancia para los servicios telemáticos que mueven la red. Aunque es válido decir que en un inicio del desarrollo de las primeras computadoras el software era sólo un añadido que se integraba a ellas, es decir, no tenía un valor comercial. A principio de los años 70 algunas empresas decidieron convertir los programas informáticos en un producto comercial, y actualmente gobiernan con sus productos la mayor parte del mercado mundial, afectando a los países más pobres con sus elevados precios de licencias.

El software libre representa una alternativa para los países en vía de desarrollo. Cuba ha optado por el uso del software libre para su desarrollo respaldado en el Acuerdo 84, de abril de 2004 del Consejo de Ministros, ya que en la mayor parte de sus empresas, organismos y centros de trabajo se utilizan software propietario, siendo los servicios telemáticos una de las principales aéreas que se encuentran en proceso de migración, colaborando así con la soberanía tecnológica a la que quiere llegar el país.

Hoy en día la administración de la red ha tomado un lugar importante dentro de las organizaciones y empresas que poseen un sistema de cómputo y que necesitan conectividad para realizar sus actividades, existen muchos servicios que se brindan en la red, entre los principales o básicos que se necesitan para mantenerla y gestionarla se encuentran:

- Servidor DHCP (*del Inglés, Dynamic Host Configuration Protocol*) para la asignación dinámica de direcciones IP a las máquinas.
- Servidor LDAP (*del Inglés, Lightweight Directory Access Protocol*) para proveer un acceso rápido a la información.
- Servidor DNS (*del Inglés, Domain Name System*), para asignar nombres descriptivos a las direcciones IP y poder localizar fácilmente los recursos de la red.

En un sistema de computadoras conectadas entre sí, cada uno de los equipos necesitan una dirección IP única, que junto a su máscara de subred identifica al Host y la subred a la que se encuentra conectado, entre otros parámetros necesarios para utilizar los servicios que se brindan en la red y para realizar la configuración de estos parámetros en cada PC conectada existen dos maneras:

1. De forma manual, que consiste en pasar por cada uno de los equipos y asignarles los parámetros de configuración.
2. Asiendo uso del DHCP.

El DHCP es el protocolo para la asignación dinámica de direcciones IP a los equipos que se conectan a una red, el mismo se publicó en octubre 1993, existen varios servidores que implementan el protocolo y brindan este servicio. Un servidor DHCP reduce la complejidad y cantidad de trabajo administrativo para configurar los equipos, evitando los errores de configuración que se producen por la necesidad de escribir los parámetros manualmente en cada equipo. Así mismo, DHCP ayuda a evitar los conflictos de direcciones IP que se producen al configurar un equipo nuevo en la red con una dirección IP ya asignada. (2)

En la migración de los servicios de red de las entidades del país se encuentra una serie de inconvenientes que afectan este proceso, porque los servidores libres que ofrecen los servicios básicos de una red se instalan y se configuran a través de consolas o terminales de comandos, lo que implica rechazo por parte de los administradores de red ya que estos necesitan dominar las estructuras y términos que se utilizan en las configuraciones de los servidores haciendo engorroso y complejo su trabajo en estos sistemas.

En la actualidad en la Universidad de las Ciencias Informáticas, en el proyecto Servicios Telemáticos del centro de Telemática, se está desarrollando una Plataforma de Gestión de Servicios Telemáticos en GNU/Linux, la cual posee módulos para administrar los servicios DNS y LDAP, facilitando el trabajo y la integración de estos, pero aun no cuenta con un módulo para gestionar el servidor DHCP en Linux.

Dado el proceso de migración en que se encuentra enfrascado el país, los problemas que existen para gestionar los servidores libres y la necesidad de incorporar un nuevo módulo en la plataforma anteriormente mencionada, surge la necesidad de desarrollar un producto que permita administrar o gestionar de forma práctica y profesional el servicio de DHCP, además de facilitar la importación de las configuraciones del DHCP de Windows Server 2003 para un servidor en GNU/Linux, ya que este servidor privado es uno de los más utilizados en las redes del país. Por lo que se propone la creación de un sistema o aplicación que solucione o resuelva las necesidades anteriormente descritas.

Como parte de la solución de la problemática existente se identifica como **problema científico**: ¿Cómo mejorar la gestión y migración del servicio DHCP para los sistemas operativos GNU/Linux?, teniendo como **objeto de estudio** los procesos que garantizan la gestión de servicios telemáticos en redes. Con un **campo de acción** dirigido a los procesos que garantizan la administración del servicio de DHCP en GNU/Linux, definiendo como **objetivo general**: Desarrollar una aplicación de escritorio multiplataforma, que ofrezca la posibilidad de administrar y gestionar de forma práctica y profesional el servicio de DHCP.

Para dar cumplimiento al objetivo de la investigación y la problemática presente se han enfocado las **tareas investigativas** a:

- Realizar un estudio del estado del arte de los servidores o sistemas similares, ya sean libres o propietarios.
- Realizar una investigación sobre el proceso de migración del servicio DHCP que implementa Windows Server 2003 hacia el servidor DHCP en GNU/Linux.
- Implementar un sistema que permita gestionar la configuración y administración de forma fácil y sencilla del servicio de DHCP, de manera local o remota.
- Hacer una selección de las metodologías, herramienta y tecnologías que existen para el desarrollo del Software, que permitan implementar una aplicación profesional.
- Desarrollar las funcionalidades para el Módulo DHCP v1.0.

Para lograr el cumplimiento de las tareas de investigativas se utilizaron los **métodos de investigación científica**. Estos métodos permiten elaborar y profundizar en dichas tareas. Los métodos **empíricos y teóricos** son los que se utilizaron en el desarrollo de la aplicación.

El método empírico es un modelo de investigación científica que se basa en la lógica de experiencias, en la percepción directa del objeto de investigación (objeto de estudio) y del problema. (3) El método empírico engloba otros métodos, de ellos se seleccionaron la observación y la experimentación.

La observación dentro del método empírico es contemplativa, y es la base del conocimiento de toda ciencia. (3) Para un desarrollo óptimo de la aplicación se realizó la observación en la investigación sobre el servidor DHCP dhcp3-server y otros servidores privativos como Windows Server 2003, con el objetivo de lograr un mejor entendimiento de su funcionamiento.

El experimento es utilizado para obtener conocimiento sobre el objeto de estudio, se realizaron experimentos con el servidor DHCP dhcp3-server donde se probaron sus posibles configuraciones, para luego analizar su funcionamiento y obtener resultados, además se realizó el mismo proceso al Windows Server 2003.

Los métodos teóricos permiten la construcción y desarrollo de la teoría científica, enfocados generalmente en los problemas de la ciencia. Por ello los métodos teóricos permiten profundizar en el conocimiento de las regularidades y cualidades esenciales de los fenómenos. (4) Dentro de los métodos teóricos se van a utilizar el inductivo-deductivo y el analítico-sistémico.

El método inductivo-deductivo se divide en la inducción que es un procedimiento mediante el cual a partir de hechos singulares se pasa a generalizaciones y la deducción es un procedimiento que se apoya en las aseveraciones y generalizaciones a partir de las cuales se realizan demostraciones o inferencias.

El método analítico-sistémico permite descomponer todo lo complejo en diversas partes y cualidades, para analizarlo y luego realizar la unión entre las partes previamente analizadas, lo que posibilita descubrir las relaciones esenciales y características. (4)

El presente documento consta de 4 capítulos:

Capítulo 1: “Fundamentación Teórica”, contiene los principales conceptos que se utilizan, además incluye un estudio de las soluciones existentes en el mundo, también se definen las técnicas, tecnologías, metodologías y herramientas que van a utilizarse para desarrollar el software.

Capítulo 2: “Características del sistema”, se define la propuesta del sistema con la construcción de un modelo de negocio adecuado, la especificación de los requerimientos del sistema, además de la elaboración de los casos de uso del sistema, descripción y los prototipos de interfaces.

Capítulo 3: “Diseño del sistema”, se concreta una propuesta del desarrollo del sistema para una próxima implementación, elaborando los diagramas de clases del Diseño.

Capítulo 4: “Implementación y Prueba”, se realizaron los diagramas de componentes, así como la estructura de despliegue del sistema dirigida a sus servidores de red. Además se aplicarán las pruebas al sistema.

## CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

### 1.1 Introducción.

El presente capítulo contiene los principales conceptos tratados en este trabajo, además de un análisis de las soluciones existentes en estos momentos que gestionan el servicio DHCP, ya sean libres o propietarias. También se hace una caracterización de las tecnologías, metodologías, lenguajes y herramientas propuestas para el desarrollo del módulo DHCP v1.0 de la Plataforma de Gestión de Servicios Telemáticos en GNU/Linux.

### 1.2 Soluciones existentes en el mundo.

En esta sección del presente capítulo se abordarán las soluciones informáticas que existen en el mundo para realizar la gestión de los servidores de red y que se utilizan en las redes del país. El desarrollo de estas aplicaciones está asociado principalmente a sistemas operativos específicos, aunque existen soluciones independientes muy competitivas y se pueden utilizar en múltiples plataformas, estas también serán abordadas con el objetivo de estudiar sus funcionalidades y características para el mejoramiento del propio producto.

#### 1.2.1 Microsoft Windows NT Server.

Microsoft Windows NT Server es un sistema operativo diseñado para su uso en servidores de red de área local (LAN). Ofrece la potencia, la manejabilidad y la capacidad de ampliación de Windows NT en una plataforma de servidor e incluye características, como la administración centralizada de la seguridad y tolerancia a fallos más avanzada, que hacen de él un sistema operativo idóneo para servidores de red. Además de brindar a sus usuarios gran facilidad de uso y de administración de sus funcionalidades, también se puede ampliar su rendimiento, ya que es aplicable a sistemas simétricos y permite incorporar procesadores adicionales.

Entre sus prestaciones incluye los siguientes servicios: (5)

- Capacidad de controlador de dominio.
- Perfiles de usuarios centralizados.
- Replicación de directorio.
- Servicio de Acceso Remoto (RAS).
- Servicios para Macintosh.

- Dynamic Host Configuration Protocol (DHCP) y Windows Internet Named Services (WINS).

En grandes entornos de redes, Windows NT Server da soporte a los siguientes entornos de computación:

- Servidor de bases de datos.
- Servidor de mensajería.
- Servidor de archivos y de impresión.
- Servidor de comunicaciones.
- Servidor WEB.

Soporte a múltiples plataformas:

- Intel 80386, 80486, Pentium y procesadores futuros.
- Power PC.
- MIPS.
- DEC Alpha AXP.
- Computadoras con simple o múltiple procesador (SMP). (5)

### 1.2.2 Microsoft Windows Server 2003.

Windows Server 2003 es un sistema operativo de la familia Windows de la marca Microsoft para servidores que salió al mercado en el año 2003. Está basada en tecnología NT y su versión del núcleo NT es la 5.2. Es un sistema operativo de propósitos múltiples capaz de manejar una gran gama de funciones de servidor, en base a sus necesidades, tanto de manera centralizada como distribuida. Algunas de estas funciones del servidor son: (6)

- Servidor de archivos e impresión.
- Servidor Web y aplicaciones Web.
- Servidor de correo.
- Terminal Server.
- Servidor de acceso remoto/red privada virtual (VPN).
- Servicio de directorio, Sistema de dominio (DNS), y servidor DHCP.
- Servidor de transmisión de multimedia en tiempo real (Streaming).
- Servidor de infraestructura para aplicaciones de negocios en línea (tales como planificación de recursos de una empresa y software de administración de relaciones con el cliente).



Microsoft Windows Server 2003 ofrece más rapidez, fiabilidad, escalabilidad y disponibilidad que Microsoft Windows NT Server, siendo además mucho más fácil de gestionar. Puede ser implementado y gestionado en menos tiempo, con un esfuerzo menor, complejidad reducida y un coste total de propiedad inferior.

- Permite a los clientes ser más productivos.
- Está construido sobre la robustez y fiabilidad de Microsoft Windows 2000 Server.
- Es el Sistema Operativo Windows más rápido, fiable y seguro que jamás haya existido.

#### **Ventajas con versiones anteriores (7)**

- Como servidor de ficheros es de un 100% a un 139% más rápido que Windows 2000 Server y un 200% más que Windows NT Server 4.0.
- Como servidor de impresión, es un 135% más eficiente que Windows NT Server 4.0.
- Como servidor web es de un 100% a un 165% más rápido que Windows 2000 Server.
- Las características mejoradas del Directorio Activo permiten realizar tareas más fácilmente, entre las que destacan la habilidad de renombrar dominios, la posibilidad de redefinir el esquema y una replicación más eficiente.
- Ofrece la mejor conectividad, facilitando al máximo la configuración de enlaces entre delegaciones, acceso inalámbrico seguro y acceso remoto a aplicaciones a través de los Terminal Services, así como en su integración mejorada con dispositivos y aplicaciones.

Aunque estas plataformas son idóneas para el trabajo con redes por la amplia gama de servicios que ofrece y la comodidad con que se administran, son muy costosas y se necesita de una gran inversión de capital para el mantenimiento de sus actualizaciones, lo cual es una opción desechable para países como Cuba que poseen una economía de escasos ingresos. (5)

#### **1.2.3 SuSE Linux Enterprise Server.**

SuSE Linux Enterprise Server es uno de los sistemas operativos para servidores líderes en el campo de la implementación profesional para entornos informáticos heterogéneos de cualquier tamaño y estructura. Está disponible para las principales plataformas de hardware: desde procesadores AMD e Intel de 32 y 64 bits hasta mainframes, pasando por toda la gama de Server de IBM. SuSE Linux Enterprise Server incorpora todos los servicios de servidor y redes relevantes para la empresa: desde servicios de ficheros, impresión, web y seguridad, hasta soluciones de aplicaciones y middleware como son bases de datos, gestión empresarial integrada (ERP) y almacenamiento.

### Ventajas (5)

- **Reducción masiva de costes:** SuSE Linux Enterprise Server elimina los costes derivados de las licencias de acceso de clientes o de acceso a Internet. Asimismo, gracias a la consolidación de servidores evitará recursos y cursos de formación innecesarios.
- **Seguridad de inversión gracias al mantenimiento del sistema y al código fuente abierto:** Los servicios de soporte, actualizaciones y mantenimiento del software se han unificado en el SuSE Linux Maintenance Programm. Esto le permite beneficiarse de un servicio completo procedente de una sola fuente y, por ejemplo, acordar el mantenimiento "perpetuo" de sus sistemas.
- **Escalabilidad y estabilidad:** Para que implemente SuSE Linux Enterprise Server donde y como quiera: el producto incluye equilibrio de red, alta disponibilidad y posibilidad de clustering, y soporta 32 CPUs, 64 GB RAM y 600 discos duros en hardware basado en x86.
- **Administración eficaz:** Mantenimiento y administración remotos de forma segura, independientemente de donde se encuentren los ordenadores. SuSE Linux Enterprise Server permite automatizar los procesos de administración e incorpora para ello todas las herramientas necesarias.

A pesar de las ventajas que ofrece SuSE Linux Enterprise Server no es una buena opción para su implantación en Cuba, por estar desarrollado por una compañía como Novell que a consecuencia del bloqueo económico le imposibilitaría al país recibir asistencia técnica. También esta es una solución no genérica debido a que las funcionalidades que brindan están estrechamente ligadas al uso de este tipo de distribución Linux. (5)

#### 1.2.4 Red Hat Enterprise Linux 5 Server.

Red Hat Enterprise Linux está disponible en dos versiones para servidores. La versión básica Red Hat Enterprise Linux Server está diseñada para implantaciones en redes de pequeño tamaño, mientras que Red Hat Enterprise Linux Advanced Platform está diseñada para escenarios mucho más grandes, proporcionándoles el entorno más rentable, flexible y escalable. Ambas versiones comparten la misma base tecnológica, y ambas incorporan un amplio conjunto de aplicaciones de servidor de código abierto y prestaciones de virtualización.

Red Hat Enterprise Linux es compatible con las siguientes arquitecturas de sistemas:

- Intel y AMD x86/x86-64
- Intel Itanium2

- IBM POWER
- IBM z-Series y S/390

Proporciona un completo conjunto complementario de aplicaciones de servicios de red, como por ejemplo DHCP, DNS, cortafuegos, así como servicios de servidor de archivos y de impresión a clientes Microsoft Windows, con integración con Active Directory. En cuanto a seguridad la incorporación de SELinux Troubleshooter, un analizador dotado de una interfaz gráfica de usuario que ayuda a los administradores en la toma de decisiones en caso de alertas de seguridad, facilita considerablemente el uso.

Este es uno de los sistemas operativos de red más competitivos del mercado mundial, incluso restándole seguidores a los productos desarrollados por la Microsoft; por ser comercial en sus versiones posteriores a Red Hat Linux 9.0 no debe ser una opción a tener en cuenta por países en vías de desarrollo. (7)

### **1.2.5 Webmin.**

Webmin representa una herramienta para la configuración mediante la Web desde GNU/Linux y otros sistemas Unix de las características propias de sistemas operativos, como servicios, usuarios, cuotas de espacio, archivos, entre otros; así como la instalación y control de muchas aplicaciones entre las que destacan: servidores DNS, DHCP, LDAP, Web, Jabber, FTP, Correo, Proxy y SSH. Permite el control de todos los servidores no solo de forma local, sino también de forma remota. Esta solución tiene una interfaz gráfica muy intuitiva y bastante clara, fue creada para la Web, aunque no es necesario un monitor con interfaz gráfica para su funcionamiento, ya que puede ejecutarse en un navegador de consola. Es una aplicación modular, escrita en el lenguaje Perl, lo que brinda la posibilidad de agregarle y quitarle funcionalidades. Su licencia inicial, la BSD o Berkeley Software Distribution (en español, Distribución de Software Berkeley) no es exactamente GPL, aunque Webmin en su versión “free” es de libre distribución, siempre y cuando no se modifique, pues, en ese caso habrá que enviar las modificaciones realizadas a su autor. (7) Además este no brinda todas las posibilidades de relación entre las declaraciones que ofrece el servidor y no permite exportar ni importar configuraciones de otro servidor DHCP.

### **1.3 Lenguaje de modelado.**

UML (Unified Modeling Language) es un lenguaje para visualizar, especificar, construir y documentar sistemas de software. Permite modelar sistemas de información, y su objetivo es lograr modelos que, además de describir con cierto grado de formalismo los sistemas, puedan ser entendidos por los clientes o usuarios de aquello que se modela.

UML no es un lenguaje de programación, es importante destacar que un modelo UML describe lo que supuestamente hará un sistema, pero no dice cómo implementar dicho sistema. UML está compuesto por Elementos, Diagramas y Relaciones, los principales diagramas de UML están clasificados en:

- Diagramas de estructura, incluye diagrama de clases, diagrama de objetos, diagrama de componentes, diagrama de estructura compuesta, diagrama de paquetes, y diagrama de despliegue.
- Diagrama de comportamiento, incluye diagrama de actividades, diagrama de casos de uso, diagrama de estados.
- Diagramas de Iteración, estos derivan de los diagramas de comportamiento incluye diagrama de secuencia, diagrama de comunicación, diagrama de tiempos, diagrama de vista de interacción. (8)

Un buen proceso de desarrollo es crucial para el éxito de un desarrollo de software. Un objetivo final de UML era ser tan simple como fuera posible pero manteniendo la capacidad de modelar toda la gama de sistemas que se necesita construir. UML necesita ser lo suficientemente expresivo para manejar todos los conceptos que se originan en un sistema moderno, tales como la concurrencia y distribución, así como también los mecanismos de la ingeniería de software, tales como encapsulación y componentes. Debe ser un lenguaje universal, como cualquier lenguaje de programación de propósito general. (9)

#### **1.4 Metodología de desarrollo.**

Todo desarrollo de software debe completar una serie de tareas para obtener un producto de software. Cada una de estas tareas en el ciclo de vida del software son desarrolladas y resueltas de varias maneras, utilizando herramientas y técnicas distintas. Se hace necesario llevar a cabo un seguimiento y control para realizar de forma correcta y en tiempo dichas tareas, por ejemplo se necesita saber cuándo se puede dar por concluida una tarea, quién debe realizarla, qué tareas preceden o anteceden a una dada, qué documentación se utilizará para llevar a cabo esa tarea. Para mejorar todo este proceso y solucionar los problemas, surgen las metodologías de desarrollo de software para proporcionarle un seguimiento a la construcción del software, estableciendo los procesos, las políticas y los procedimientos que intervienen en el desarrollo del software, para garantizar la eficacia en dichos procesos. Una de estas metodologías es RUP.

RUP es un proceso de desarrollo de software que constituye una metodología, utilizando UML para la modelación es una de las más utilizadas para desarrollo de sistemas. Como RUP es un proceso, en su modelación define como sus principales elementos: (10)

- **Trabajadores (“quién”):** Define el comportamiento y responsabilidades (rol) de un individuo, grupo de individuos, sistema automatizado o máquina, que trabajan en conjunto.
- **Actividades (“cómo”):** Es una tarea que tiene un propósito claro, es realizada por un trabajador y manipula elementos.
- **Artefactos (“qué”):** Productos tangibles del proyecto que son producidos, modificados y usados por las actividades. Pueden ser modelos, elementos dentro del modelo, código fuente y ejecutables.
- **Flujo de actividades (“cuándo”):** Secuencia de actividades realizadas por trabajadores y que produce un resultado de valor observable.

RUP posee tres características principales: (10)

**Dirigido por casos de uso:** Los casos de uso reflejan lo que los usuarios futuros necesitan y desean, lo cual se capta cuando se modela el negocio y se representa a través de los requerimientos. A partir de aquí los casos de uso guían el proceso de desarrollo ya que los modelos que se obtienen, como resultado de los diferentes flujos de trabajo, representan la realización de los casos de uso.

**Centrado en la arquitectura:** La arquitectura muestra la visión común del sistema completo en la que el equipo de proyecto y los usuarios deben estar de acuerdo, por lo que describe los elementos del modelo que son más importantes para su construcción, los cimientos del sistema que son necesarios como base para comprenderlo, desarrollarlo y producirlo económicamente.

**Iterativo e Incremental:** Es práctico dividir el trabajo en partes más pequeñas o mini proyectos. Cada mini proyecto es una iteración que resulta en un incremento. Las iteraciones hacen referencia a pasos en los flujos de trabajo, y los incrementos, al crecimiento del producto. Cada iteración se realiza de forma planificada es por eso que se dice que son mini proyectos.

Las fases de RUP son:

- Conceptualización (Concepción o Inicio): Se describe el negocio y se delimita el proyecto describiendo sus alcances con la identificación de los casos de uso del sistema.
- Elaboración: Se define la arquitectura del sistema. A pesar de que se desarrolla a profundidad una parte del sistema, las decisiones sobre la arquitectura se hacen sobre la base de la comprensión del sistema completo y los requerimientos (funcionales y no funcionales) identificados de acuerdo al alcance definido.

- **Construcción:** Se obtiene un producto listo para su utilización que está documentado y tiene un manual de usuario. Se obtiene uno o varios *release* del producto que han pasado las pruebas. Se ponen estos *release* a consideración de un subconjunto de usuarios.
- **Transición:** El *release* ya está listo para su instalación en las condiciones reales. Puede implicar reparación de errores.

En RUP se han agrupado las actividades en grupos lógicos definiéndose 9 flujos de trabajo principales. Los 6 primeros son conocidos como flujos de ingeniería y los tres últimos como de apoyo.

### 1.5 Modelación de funciones mediante IDEF0.

IDEF0 (Integration Definition for Function Modeling) es una técnica de modelación que surge en la década de los 70, para representar de manera estructurada y jerárquica los procesos o actividades que se realizan en un sistema. Esta técnica posee dos notaciones diferentes que son la AS-IS y TO-BE.

La notación AS-IS es empleada para representar los procesos del negocio de la manera en que se encuentran antes de realizar la automatización de los mismos. Y la notación TO-BE es utilizada para transformar el proceso de la forma deseada luego de identificarse las mejoras posibles y teniendo en cuenta los objetivos de la organización.

Existen cinco elementos en un modelo IDEF0: Las flechas de las entradas, controles, salidas y mecanismos denominadas con el nombre de ICOMs (Input, Control, Output, Mechanism), y las actividades o procesos.

- La actividad (o proceso): Se representa por medio de cajas.
- Las entradas: Se representan mediante flechas que entran por la parte izquierda de una caja.
- Las salidas: Se representan por flechas saliendo por la parte derecha de las actividades.
- Los controles: Representados por flechas que entran por la parte superior de las cajas.
- Los mecanismos: Para poder llevar a cabo el proceso, que se representan mediante flechas entrantes a las cajas por su parte inferior. (11)

El sistema es jerárquico. Cada diagrama representa una actividad necesaria para la tarea, en un grado de detalle específico. Las actividades se subdividen en diagramas que siguen en niveles inferiores hasta un grado de detalle necesario.

Es importante destacar que esta técnica se utiliza como homóloga del flujo de trabajo Modelamiento del Negocio que propone RUP.

## 1.6 Lenguaje de programación Python.

Es lenguaje de programación de alto nivel creado por Guido Van Rossum en 1991, muy potente, dinámico y a la vez sencillo. Ofrece eficaces estructuras de datos de alto nivel y una simple, pero efectiva programación orientada a objetos. La sintaxis elegante de Python y su escritura dinámica, junto con su naturaleza interpretada lo convierte en un lenguaje ideal para el scripting y el desarrollo rápido de aplicaciones en muchas áreas y en la mayoría de las plataformas. Se desarrolla como un proyecto de código abierto, que lo hace libremente utilizable y distribuible, incluso para uso comercial. Su licencia es administrada por la Python Software Foundation. (12)

Python permite dividir el programa en módulos reutilizables. Incluye una gran colección de módulos estándar que se pueden utilizar como base de los programas que proporcionan funcionalidades para el manejo de ficheros, llamadas al sistema, sockets y hasta interfaces a librerías gráficas como (GTK, QT, TK).

Además posee tipado dinámico por lo que puede tomar una variable y usarla en diferentes tipos de datos durante la ejecución del programa y soporta varios paradigmas, puesto que no trata de forzar a los programadores a adoptar un determinado modelo de paradigma o estilo.

Entre sus principales características se encuentran: (12)

- Simple: Python es un lenguaje simple. El pseudo-código natural de Python es una de sus grandes fortalezas ya que permite concentrarse en la solución del problema en lugar de la sintaxis del propio lenguaje.
- Sencillo de Aprender: Con Python es extremadamente sencillo de iniciarse en la programación ya que ofrece una sintaxis extraordinariamente simple.
- Libre y Fuente Abierta: Python es un ejemplo de un FLOSS (Free/Libre and Open Source Software - Gratuito/Libre y Software de Fuente Abierta). En términos simples, se puede distribuir libremente copias de este software, leer su código fuente, hacerle cambios y usar partes del mismo en nuevos programas libres.
- Lenguaje de Alto Nivel: Cuando se escriben programas en Python no hay que preocuparse por detalles de bajo nivel, como manejar la memoria empleada por tu programa.
- Portable: Debido a su naturaleza de ser Open Source, Python ha sido portado a diversas plataformas. Se puede usar Python sobre Linux, Windows, Macintosh, Solaris, OS/2, AROS, AS/400, BeOS,

OS/390, z/OS, Palm OS, QNX, VMS, Psion, Acorn RISC OS, VxWorks, PlayStation, Sharp Zaurus, Windows CE y PocketPC.

- Interpretado: El programa se ejecuta desde el código fuente. Internamente, Python convierte el código fuente en una forma intermedia llamada bytecodes, después los traduce en el lenguaje nativo de tu computadora y ejecuta.
- Orientado a Objetos: En lenguajes orientados a objetos, el programa es construido sobre objetos los cuales combinan datos y funcionalidad.

### 1.7 Interfaz gráfica Qt Designer.

QT Designer es una herramienta para el desarrollo de formularios y presentaciones gráficas para las aplicaciones. Permite acelerar el desarrollo de interfaces de alto rendimiento, a la vez que proporciona una forma fácil de diseñar interfaces gráficas de usuario avanzadas generando el código fuente para las mismas, lo que permite al desarrollador ajustarlo a sus necesidades.

Fue creado inicialmente por la empresa TROLLTECH para trabajar en varias distribuciones Linux. En la actualidad puede instalarse en plataformas como Windows y Mac OsX. Qt Designer utiliza como base la librería gráfica de Qt, que ha sido transportada a diversas plataformas, lo que permite que el código generado por el Qt Designer pueda ser utilizado en diversas plataformas. Además, Qt funciona solo o asociándose a algunos entornos de desarrollo integrado. Esta herramienta provee características muy poderosas como la previsualización de la interfaz, soporte para widgets y un editor de propiedades bastante poderoso. (13)

### 1.8 Entorno de desarrollo EasyEclipse.

EasyEclipse es un lanzamiento de Eclipse previamente empaquetado que proporciona un completo y funcional Entorno de Desarrollo Integrado (IDE). Cada distribución contiene todo lo necesario para empezar a desarrollar el código. Una vez que se haya instalado una distribución, puede complementarlo mediante la instalación de más plugins según sea necesario. PyDev es un plugin que permite a los usuarios utilizar Eclipse para el desarrollo de aplicaciones en Python y Jython. Su distribución es bajo la licencia Eclipse Public License 1.0. Posee varias facilidades tales como, el auto completamiento de código, análisis de sintaxis resaltado, análisis de código, depuración, consola interactiva y muchos otros.

EasyEclipse es:

- Multiplataforma
- Gratuito y de código abierto



- Fácil de mantener, sin problemas de versión y de dependencia (14)

Por todas estas características es que EasyEclipse se ha utilizado como el IDE para el desarrollo del módulo.

### 1.9 Herramienta case Visual Paradigm

Visual Paradigm para UML (VP-UML) 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. Permite dibujar todos los tipos de diagramas de clases y generar documentación. (15)

Es una potente plataforma que ofrece disímiles ventajas, y fácil de usar en el modelado visual de UML. Ofrece a los desarrolladores de software crear aplicaciones más rápidas, de calidad, mejor y más barato. Además facilita una excelente interoperabilidad con otras herramientas CASE.

Entre sus principales características se encuentra:

- Ingeniería inversa y directa.
- Generación de código.
- Disponibilidad de integrarse en los principales IDE.
- Disponibilidad en múltiples plataformas.
- Generación de código fuente en varios lenguajes de programación.
- Generador de informes para creación de documentación.

Visual Paradigm facilita una rápida construcción de aplicaciones con calidad a un menor costo. También, soporta UML en su versión 2.1 y se encuentra disponible en múltiples versiones y plataformas. (15)

### 1.10 Conclusiones

En el presente capítulo se ha realizado un estudio de los principales sistemas y herramientas existentes y utilizadas en el mundo para gestionar y brindar un servicio de red DHCP, logrando visualizar las características que estos poseen y tomando de ellos sus mejores aportes para la aplicación a desarrollar, teniendo una mejor comprensión del tema en cuestión y contribuyendo de esta forma con la soberanía tecnológica de Cuba. También se realizó una selección de la metodología de desarrollo, los lenguajes de modelado y de programación, así como las herramientas de desarrollo de interfaz gráfica y el IDE, justificando su uso en el desarrollo del sistema.

## CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

### 2.1 Introducción

En el presente capítulo se hace una descripción de los procesos del negocio asociados a la administración del Servicio DHCP en sistemas operativos GNU/Linux. Se utiliza la notación IDEF0 para una mayor comprensión de todos los procesos y la manera en que se desarrollan en la actualidad. Se exponen los requisitos funcionales del sistema, el diagrama de casos de uso del sistema y la descripción de los mismos, así como los prototipos de interfaz de usuario para cada caso de uso crítico.

### 2.2 Problema y situación problemática

Hoy en día la administración de la red ha tomado un lugar importante dentro de las organizaciones y centros, para lo cual existe un conjunto de herramientas que en su mayoría son productos comerciables por las que hay que pagar un alto precio, las cuales tienen excesivas restricciones de licencias y no ofrecen su código fuente, además no se tiene derecho de asistencia y ayuda técnica en caso de necesitarla. Por lo que Cuba ha optado por la migración hacia el software libre de manera progresiva.

Uno de los mayores inconvenientes luego de migrar los servidores a sistemas libres es el uso de consolas de comando para la gestión de las configuraciones, haciendo difícil y complejo el trabajo de los administradores de red ya que estos tienen que dominar todas las estructuras y términos que se utilizan para conformar los ficheros de configuración.

En la UCI se han implementado módulos para la gestión de los servicios DNS y LDAP en GNU/Linux, pero no existe uno para DHCP. Por lo que surge la necesidad de desarrollar una aplicación que gestione dicho servicio, el cual va a contribuir con el desarrollo y la soberanía tecnológica, con sello propio y adecuado a las características de Cuba.

### 2.3 Información que se maneja

En el sistema existen múltiples procesos que usan y modifican la información que utiliza el servidor DHCP en GNU/Linux para sus configuraciones, por lo que es de suma importancia conocer dicha información y la forma en que se maneja, ya que, los ficheros de configuración por ser la principal fuente de información, debe tener la estructura adecuada para lograr un correcto funcionamiento, de lo contrario podría ocasionar un inadecuado funcionamiento o incluso que se detenga el servicio. La modificación de dichos ficheros de configuración tiene que ser con extremo cuidado.

### 2.4 Propuesta de sistema

Para el desarrollo de la herramienta de gestión del servidor DHCP, se decidió implementar múltiples funcionalidades, como la gestión de Subredes, Host, Grupo, Red Compartida y Servidores Remotos; también debe permitir integrar los servicios DHCP y DNS y mostrar reportes sobre las concesiones realizadas por el servidor, además de brindar la opción al administrador de migrar un servidor que reside en un Windows Server 2003 hacia el servidor en GNU/Linux.

La aplicación permitirá al administrador adicionar, modificar, eliminar, y mostrar las Subredes, Host, Grupos, Redes Compartidas o Servidores Remotos, logrando flexibilidad y sencillez a la hora de brindar y configurar el servicio DHCP, además de facilitar la migración o importación de las configuraciones.

Una herramienta como la que se propone no tiene limitantes de costo de la licencia, con soporte y mantenimiento para la comunidad cubana, con interfaces cómodas e intuitivas sin dejar de ser profesional para los administradores, y con un número considerable de funcionalidades que facilita el trabajo de administración de un servidor DHCP.

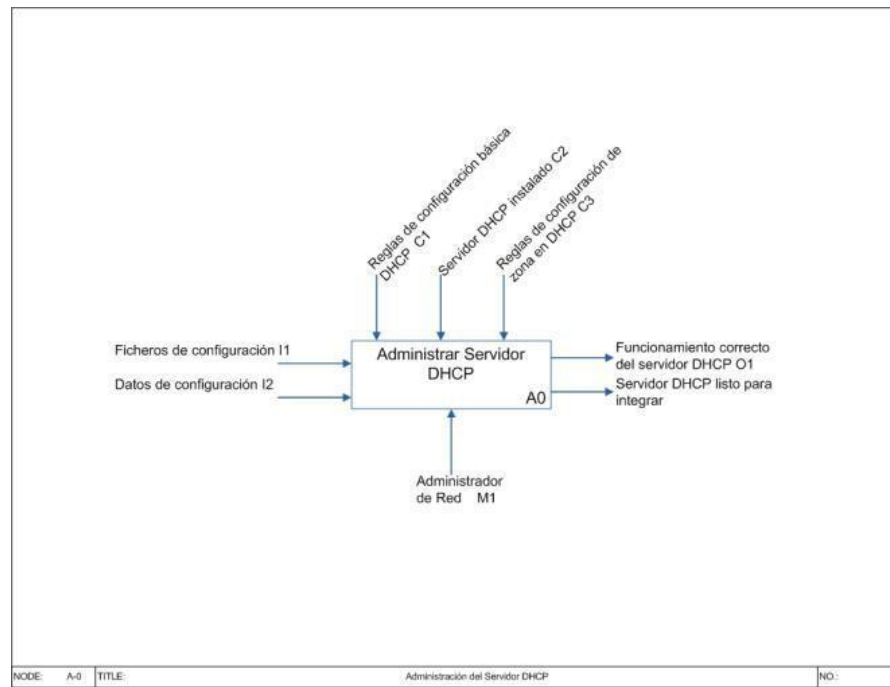
## **2.5 Modelo de negocio**

### **2.5.1 Representación de los procesos con IDEF0**

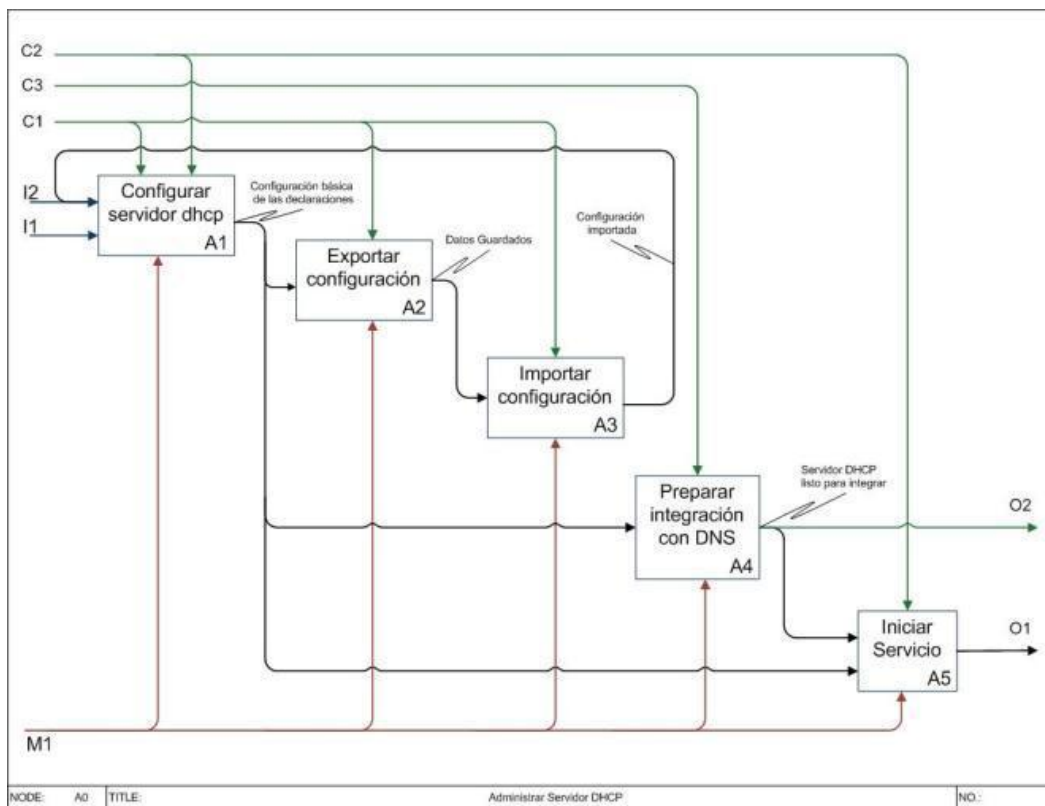
En la representación de los procesos se definió un proceso general denominado “Administrar Servidor DHCP”, que a su vez está dividido en otros cinco procesos que detallan su funcionamiento:

- Configurar básico servidor.
- Importar configuración.
- Preparar integración con DNS.
- Exportar configuración.
- Iniciar servicio.

#### **2.5.1.1 Diagrama A-0**



2.5.1.2 Diagrama A0



## 2.5.2 Descripción de los procesos del negocio

### 2.5.2.1 Descripción de procesos del diagrama A-0.

| Ficha de Proceso  |                              |   |
|---|------------------------------|---|
| Proceso   | Administración Servidor DHCP |   |
| Notación:   | A-0                          |   |
| Entradas:   | I1                           | <b>Fichero de configuración:</b> Es el fichero donde se guardan las configuraciones que utiliza el Servidor para su funcionamiento.   |
|   | I2                           | <b>Datos de configuración:</b> Son los datos que se utilizan para crear las declaraciones que se guardan en el fichero de configuración.  |
| Controles:  | C1                           | <b>Reglas de configuración básica del DHCP:</b> Estas reglas son las que establecen la estructura y los términos a utilizar en las declaraciones que adicionaran al fichero de configuración. |
|   | C2                           | <b>Servidor DHCP Instalado:</b> El servidor DHCP de GNU/Linux es el encargado de controlar que las configuraciones realizadas en el fichero del mismo estén correctas.                        |
|   | C3                           | <b>Regla de configuración de zona DHCP:</b> Estas establecen el formato que tendrán las zona que se van a actualizar en el fichero de configuración de servidor DHCP.                         |
| Salidas:  | O1                           | <b>Funcionamiento correcto del servidor DHCP:</b> Al adicionar las declaraciones al fichero DHCP e iniciar el servicio se obtiene como salidas un funcionamiento correcto del servidor DHCP.  |
|   | O2                           | <b>Servidor listo para integrar:</b> Las configuración queda lista para realizar una integración con el servidor DNS.   |
| Mecanismos:   | M1                           | <b>Administrador de Red.</b>  |
| Descripción del Proceso   |                              |   |
| <p>Proceso en el cual se administra y configura el servidor DHCP, entre las actividades de administración se encuentra la gestión de las declaraciones, preparar la integración de los servidores DHCP y DNS, además de la importación y exportación de las configuraciones del servidor DHCP en GNU/Linux.</p> |                              |   |

### 2.5.2.2 Descripción de procesos del diagrama A0.

#### 2.5.2.2.1 Proceso Configurar Servidor DHCP

| Ficha de Proceso |                                 |
|------------------|---------------------------------|
| Proceso          | <b>Configurar Servidor DHCP</b> |
| Notación:        | A1                              |

|             |    |  |
|-------------|----|--|
| Entradas:   | I1 | <b>Ficheros de Configuración.</b>                              |
| Controles:  | C1 | <b>Reglas de configuración básica del DHCP.</b>                |
|             | C2 | <b>Servidor DHCP Instalado.</b>                                |
| Salidas:    |    | <b>Configuración básica de las declaraciones del servidor.</b> |
| Mecanismos: | M1 | <b>Administrador de Red.</b>                                   |

#### Descripción del Proceso

Proceso en el cual se configuran las declaraciones (subred, máquina, red compartida y grupos) que necesarias para brindar el servicio DHCP, dichas configuraciones serán validadas por el servidor DHCP de GNU/Linux. La salida de este proceso tributará como entrada al proceso de Inicio de servicio.

#### 2.5.2.2.2 Proceso Exportar Configuración

##### Ficha de Proceso

|             |                                |  |
|-------------|--------------------------------|--|
| Proceso     | <b>Exportar Configuración.</b> |  |
| Notación:   | A2                             |  |
| Entradas:   |                                | <b>Configuración básica de las declaraciones del servidor.</b> |
| Controles:  | C1                             | <b>Reglas de configuración básica del DHCP.</b>                |
| Salidas:    |                                | <b>Datos Guardados.</b>  |
| Mecanismos: | M1                             | <b>Administrador de Red.</b>                                   |

#### Descripción del Proceso

Proceso en el cual se toman las configuraciones que deben estar implementadas en un servidor y exportarlas a un formato específico, para tener una copia o respaldo, luego aplicarlas a un servidor cuando el administrador estime conveniente. Este proceso tendrá como salida los datos guardados, que formarán parte de la entrada al proceso de importar configuración.

#### 2.5.2.2.3 Proceso Importar Configuración.

##### Ficha de Proceso

|             |                               |   |
|-------------|-------------------------------|---|
| Proceso     | <b>Importar Configuración</b> |   |
| Notación:   | A3                            |   |
| Entradas:   |                               | <b>Datos Guardados.</b>                         |
| Controles:  | C1                            | <b>Reglas de configuración básica del DHCP.</b> |
| Salidas:    |                               | <b>Configuraciones Importadas.</b>              |
| Mecanismos: | M1                            | <b>Administrador de Red.</b>                    |

### Descripción del Proceso

Proceso en el cual se realiza el proceso de importar a un servidor determinado por el administrador de red las configuraciones anteriormente exportadas. Luego de dicho proceso la salida del mismo va a ser una entrada al proceso de administración.

#### 2.5.2.2.4 Proceso Integrar Servicios.

### Ficha de Proceso

|             |                                      |  |
|-------------|--------------------------------------|--|
| Proceso     | <b>Preparar integración con DNS.</b> |  |
| Notación:   | A4                                   |  |
| Entradas:   |                                      | <b>Configuración básica de las declaraciones del servidor.</b> |
| Controles:  | C3                                   | <b>Regla de configuración de zona DHCP:</b>                    |
| Salidas:    | O2                                   | <b>Servidor listo para integrar.</b>                           |
| Mecanismos: | M1                                   | <b>Administrador de Red.</b>                                   |

### Descripción del Proceso

Proceso en el cual se prepara el servidor DHCP para realizar una futura integración con el DNS, utilizando las configuraciones. Luego de todo este proceso la salida del mismo va a dar paso el proceso de Inicio de Servicio.

#### 2.5.2.2.5 Proceso Iniciar Servicio.

### Ficha de Proceso

|             |                         |  |
|-------------|-------------------------|--|
| Proceso     | <b>Iniciar Servicio</b> |  |
| Notación:   | A5                      |  |
| Entradas:   |                         | <b>Servidor listo para integrar.</b>                           |
|             |                         | <b>Configuración básica de las declaraciones del servidor.</b> |
| Controles:  | C2                      | <b>Servidor DHCP Instalado.</b>                                |
| Salidas:    | O1                      | <b>Funcionamiento Correcto del Servidor.</b>                   |
| Mecanismos: | M1                      | <b>Administrador de Red.</b>                                   |

### Descripción del Proceso

Proceso en el cual se iniciará el servicio DHCP, luego de tener configurado el servidor o listo para hacer una integración, este puede dar como resultado un correcto funcionamiento del servidor o fallido el intento lo que va a dar paso al proceso Configurar servidor DHCP para corregir los errores por lo cual falla el inicio del servicio.

## 2.6 Especificación de los requisitos de software

Posiblemente la etapa más compleja a lo largo de la producción de un software sea la de captura de requerimientos. Es en esta etapa en la cual se determina qué desea el cliente. Para lograr esto se hace necesario el establecimiento de una visión común entre el cliente y el equipo de desarrollo sobre los procesos que se llevan a cabo en el negocio y cómo estos se verán reflejados y mejorados en el futuro software.

### 2.6.1 Requisitos Funcionales.

Los requisitos funcionales reflejan las capacidades con las que debe contar el sistema. Estos se mantienen invariables sin importar con qué propiedades o cualidades se relacionen. A continuación serán tratados los requerimientos funcionales asociados al sistema:

#### RF1. Gestionar Subred

##### RF1.1. Adicionar subred.

Permite adicionar una subred al DHCP luego de llenar los siguientes campos:

- Subred.
- Máscara de red.
- Rangos de IP.

##### RF1.2. Modificar subred.

Permite modificar una subred del DHCP luego de modificar algunos de los siguientes campos:

- Subred.
- Máscara de red.
- Rangos de IP.

##### RF1.3. Eliminar subred.

Permite eliminar una subred del DHCP luego de seleccionarla en el árbol.

##### RF1.4. Mostrar subred.

Muestra la subred seleccionada en el árbol.

#### RF2. Gestionar Grupo.

##### RF2.1. Adicionar grupo.

Permite adicionar un grupo al DHCP luego de especificarle un identificador y de seleccionar que declaraciones de subred, red compartida, host y otros grupos formaran parte de este.

##### RF2.2. Modificar grupo.



Permite modificar un grupo en el DHCP luego de modificar el identificador o de seleccionar que declaraciones de subred, red compartida, host y otros grupos formaran parte de este.

#### RF2.3. Eliminar grupo.

Permite eliminar un grupo seleccionado en el árbol.

#### RF2.4. Mostrar grupo

Permite mostrar un grupo seleccionado en el árbol.

### RF3. Gestionar Red Compartida

#### RF3.1. Adicionar red compartida.

Permite adicionar una red compartida luego de seleccionar que declaraciones de subred, host y grupo formaran parte de esta y después de especificar el campo siguiente:

- Nombre.

#### RF3.2. Modificar red compartida.

Permite modificar una red compartida luego de seleccionar que declaraciones de subred, host y grupo formaran parte de esta o después de modificar el campo siguiente:

- Nombre.

#### RF3.3. Eliminar red compartida.

Permite eliminar la red compartida seleccionada en el árbol.

#### RF3.4. Mostrar red compartida

Permite mostrarla red compartida seleccionada en el árbol.

### RF4. Gestionar Host

#### RF4.1. Adicionar host.

Permite adicionar un host al DHCP luego de llenar los siguientes campos:

- Dirección MAC.
- Dirección IP asignar.
- Nombre del host

#### RF4.2. Modificar host.

Permite modificar un host del DHCP luego de modificar algunos de los siguientes campos:

- Dirección MAC.
- Dirección IP asignar.
- Nombre del host.

#### RF4.3. Eliminar host.

Permite eliminar un host previamente seleccionado en el árbol.

RF4.4. Mostrar host.

Permite mostrar un host seleccionado en el árbol.

**RF5. Gestionar Opciones y Parámetros.**

RF5.1. Adicionar opción o parámetro.

Permite adicionar una opción o parámetro al DHCP luego de especificar los siguientes campos:

- Tipo de opción o parámetro.
- Valor.

RF5.2. Modificar opción o parámetro.

Permite modificar una opción o parámetro del DHCP luego de seleccionarlo en la lista y de alterar alguno de los siguientes campos:

- Tipo de opción o parámetro.
- Valor.

RF5.3. Eliminar Opción o Parámetro.

Permite eliminar la opción o parámetro seleccionado en el listado.

RF5.4. Mostrar Opción o Parámetro.

Permite mostrar la opción o parámetro seleccionado en el listado.

**RF6. Adicionar Servidor Remoto.**

Permite adicionar un nuevo servidor DHCP remoto para gestionar sus declaraciones y configuraciones.

**RF7. Preparar escenario para Integrar DHCP con DNS.**

Permite preparar el escenario para una futura la integración con el DNS, para lograr la comunicación entre ambos servicios.

**RF8. Importar configuraciones DHCP.**

Permite importar las configuraciones que se hayan exportado anteriormente.

**RF9. Exportar configuraciones DHCP.**

Permite exportar las configuraciones de un DHCP seleccionado en el árbol.

**RF10. Mostrar Reportes estadísticos sobre las asignaciones.**

Permite mostrar reportes estadísticos sobre el uso de las direcciones IP de un servidor DHCP.

**RF11. Buscar Subred.**

Permite buscar una Subred del DHCP luego de haber especificado la subnet de la misma.

**RF12. Buscar Red Compartida.**

Permite buscar una Red Compartida del DHCP luego de haber especificado el nombre de la misma.

**RF13. Buscar Grupo.**

Permite buscar grupo del DHCP luego de haber especificado el identificador del mismo.

**RF14. Buscar Host.**

Permite buscar host del DHCP luego de haber especificado el nombre del mismo.

**RF15. Buscar Opciones o Parámetros.**

Permite buscar parámetros u opciones luego de haber especificado el ámbito de los mismos.

### **2.6.2 Requerimientos no funcionales**

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener. Representan características que hacen al producto atractivo, rápido, confiable, entre otras. En muchos casos los requerimientos no funcionales son fundamentales en el éxito del producto. Normalmente están vinculados a requerimientos funcionales, es decir una vez que se conozca lo que el sistema debe hacer, se puede determinar cómo ha de comportarse y las cualidades que debe tener.

#### **Apariencia o interfaz externa:**

La interfaz debe tener un diseño sencillo con los componentes visuales necesarios, permitiendo la utilización del sistema por cualquier persona sin mucha experiencia. No se debe sobrecargar la aplicación con muchas imágenes y gráficos para acelerar la velocidad de respuesta del sistema, por lo que tendrá íconos sencillos que representen claramente la operación a realizar.

#### **Usabilidad:**

La aplicación contara con un instalador que guiara en pocos pasos el proceso de instalación del sistema, garantizando un acceso directo en el menú principal y en el escritorio si el usuario lo desea. El sistema podrá ser usado por cualquier usuario que posea conocimientos básicos en la administración de servicios telemático. La aplicación contará con un manual para su uso, donde se explicarán detalladamente las funcionalidades del software para garantizar la eficiente gestión de los servicios telemáticos.

**Soporte:**

Se requiere que el sistema cuente con la documentación apropiada para agilizar la instalación, configuración y administración de los servicios telemáticos eficientemente. Se brindarán cursos de capacitación al personal que utilizará el software en caso de ser necesario.

**Seguridad:**

El sistema garantiza la seguridad de la información utilizando el protocolo de conexión SSH, con el cual todas las transferencias se realizan de forma encriptada, usando llaves públicas y privadas.

**Políticos-culturales:**

El módulo debe contar con logotipos e imágenes que se encuentren en correspondencia con el carácter científico y profesional del tema.

**Legales:**

La aplicación se realizará utilizando herramientas de software libre, pero será un producto propio de la UCI que podrá comercializarse en un futuro.

**Confiabilidad:**

Validar la captación de datos en el caso de los administradores de red al inscribirse para evitar entradas inadecuadas.

**Software:**

Los ordenadores donde se vaya a ejecutar la aplicación necesitan tener instalado una distribución de GNU/Linux. Además debe poseer los paquetes, librerías y dependencias mínimas necesarias que van a hacer usadas por la aplicación.

**Hardware:**

Los requisitos mínimos de hardware para Módulo DHCP son, una PC donde se instale la aplicación con las siguientes características:

- Microprocesador Pentium IV.
- 256 MB de memoria RAM.
- 100 MB de espacio libre en el disco duro.

**Restricciones del diseño y la implementación:**

Este tipo de requerimiento especifica o restringe la codificación o construcción de nuestro sistema, constituyen restricciones que deben ser cumplidas estrictamente.

- Será empleado Python 2.6 como lenguaje de programación para la construcción del sistema.
- La codificación estará regida por la guía de estilo propuesta por Guido Van Rossum creador del lenguaje de programación Python.
- Las herramientas utilizadas están divididas fundamentalmente en dos grupos: herramientas de diseño y herramientas de codificación; en el primer caso se ha seleccionado *Qt Designer* en su versión 4.2.1 para elaborar las interfaces de usuario, en el segundo caso aunque Python permite extender su codificación a los más simples editores de texto, se ha decidido emplear *Eclipse con Pydev*.
- Como arquitectura general del sistema se ha concluido la utilización de una arquitectura en capas debido a las ventajas que ello proporciona en términos de claridad, facilidad en la corrección de errores producidos durante el desarrollo, rápido desarrollo al permitir el trabajo paralelo en cada una de las capas definidas, entre otros factores.
- Sería realmente ineficiente si en cada ocasión que los desarrolladores de sistemas informáticos se proponen una nueva meta deban comenzar la elaboración de su producto desde la base, sobretodo en casos donde se pueden reutilizar muchas funcionalidades. Con el objetivo de resolver la problemática anterior han sido creadas bibliotecas de clases o frameworks que faciliten el desarrollo y acorten los tiempos de producción, permitiendo a los desarrolladores concentrarse en aspectos de mayor nivel.

## 2.7 Modelo de Casos de Uso del Sistema

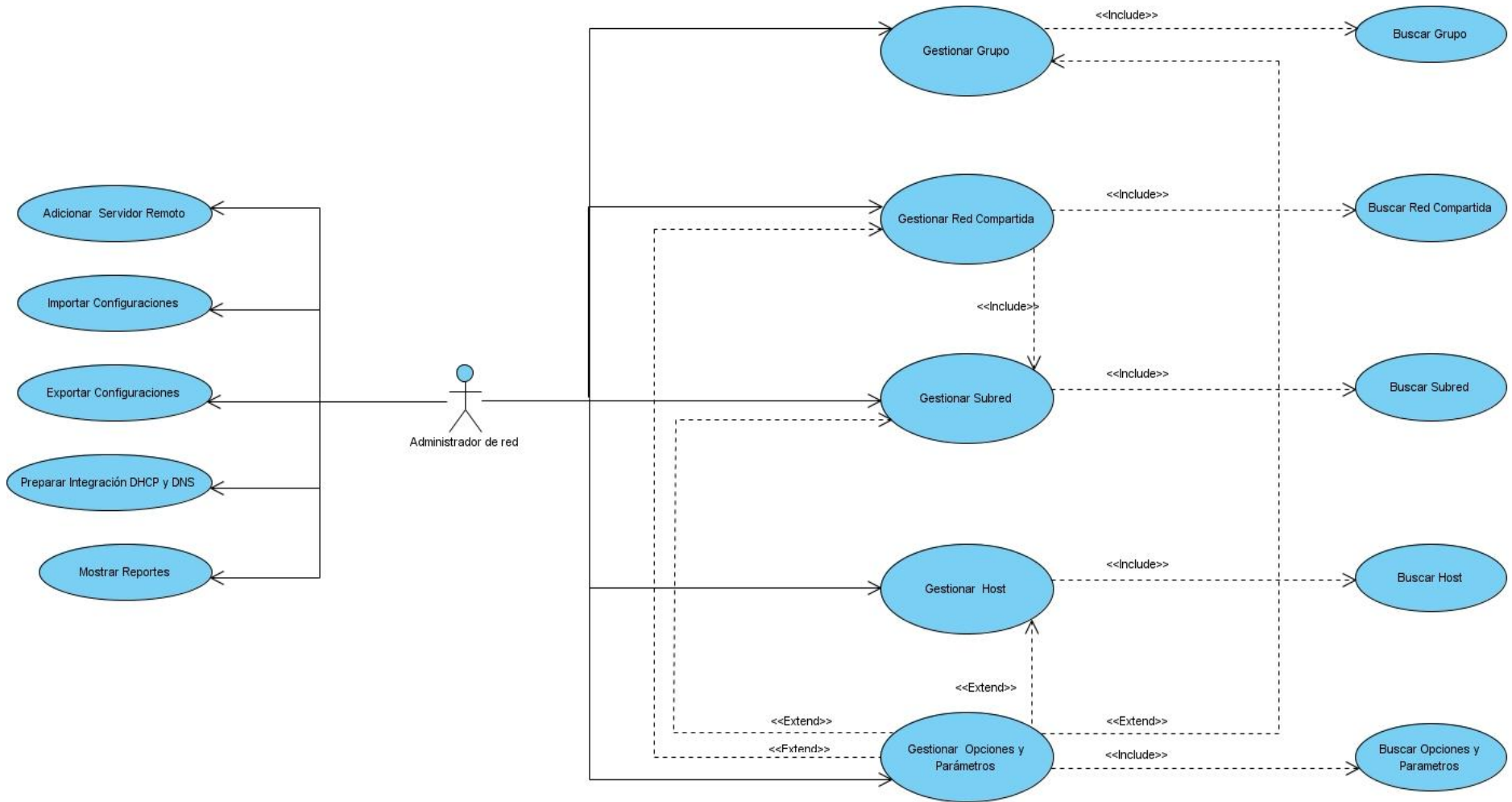
### 2.7.1 Definición de los actores del sistema a automatizar

En el módulo de DHCP interviene un actor: el administrador de red.

#### Descripción de los actores del sistema.

| Actor                | Descripción   |
|----------------------|---|
| Administrador de Red | Es el encargado de gestionar el servicio DHCP a través de la aplicación. Tiene permisos para instalar el servidor, realizarle modificaciones a las configuraciones y modificar su estado. |

## 2.7.2 Diagrama de Casos de Uso del Sistema.



### 2.7.3 Especificación de los Casos de Uso

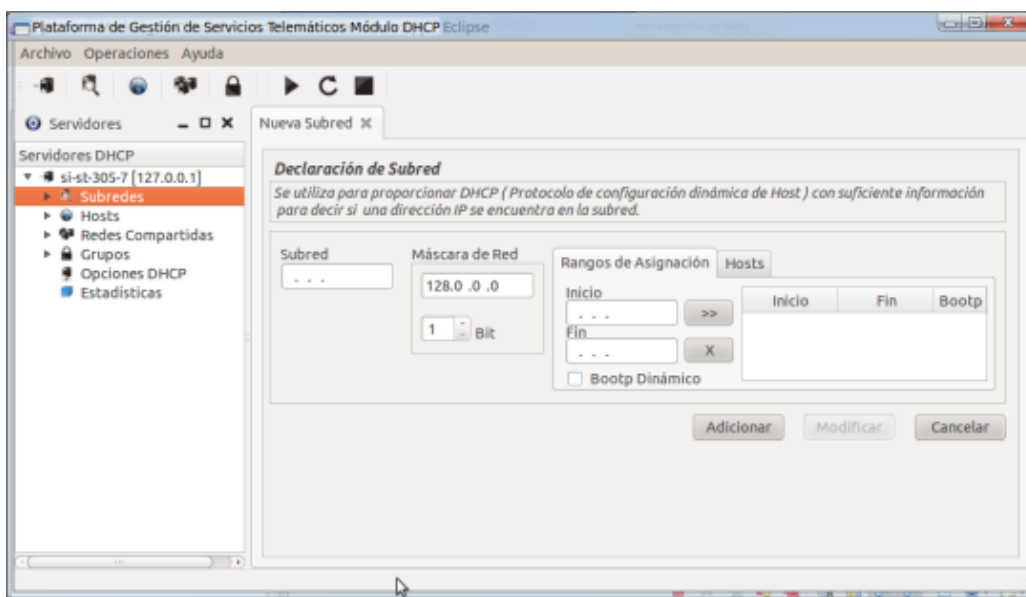
Luego de obtener el diagrama de Casos de Uso del sistema, y para lograr un mayor entendimiento de las funcionalidades asociadas a cada uno de estos, se realizará una descripción detallada de los CU críticos del Sistema, para los demás CU sólo se hará una descripción general.

#### 2.7.3.1 Descripción del Casos de Uso Gestionar Subred

|   |  |  |
|---|--|--|
| <b>Caso de Uso:</b>   | Gestionar Subred   |  |
| <b>Actores:</b>   | Administrador de red   |  |
| <b>Resumen:</b>   | El CU inicia cuando el actor necesite insertar, modificar, eliminar y mostrar datos de una subred y el caso de uso termina cuando el administrador ha realizado las operaciones satisfactoriamente.  |  |
| <b>Precondiciones:</b>  | El administrador debe estar autenticado y debe haber seleccionado el servidor DHCP o la declaración donde desea realizar la gestión de subred.   |  |
| <b>Referencias</b>  | RF2, RF12  |  |
| <b>Prioridad</b>  | <b>Alta</b>  |  |
| <b>Flujo Normal de Eventos</b>  |  |  |
| <b>Acción del Actor</b>   | <b>Respuesta del Sistema</b>   |  |
| 1- El administrador selecciona una subred en el árbol o realiza una de las siguientes acciones en el menú Operaciones submenú Subredes: <ul style="list-style-type: none"> <li>• Nueva subred</li> <li>• Eliminar subred</li> </ul> | 2- Ejecuta alguna de las siguientes acciones: <ol style="list-style-type: none"> <li>a. Si selecciona Nueva subred, ir a Sección Nueva subred.</li> <li>b. Si selecciona Eliminar subred ir a Sección Eliminar subred.</li> <li>c. Si selecciona una subred existente en el árbol ir a la Sección Mostrar Subred.</li> </ol> |  |

| Sección "Nueva subred"  |   |
|---|---|
| Acción del Actor  | Respuesta del Sistema   |
| <p>2. El administrador llena los campos correspondientes y llena la lista de rangos de la subred.</p> <p>3. El administrador presiona el botón Adicionar.</p> | <p>1- El sistema Muestra la interfaz Nueva Subred con los siguientes campos a llenar:</p> <ul style="list-style-type: none"> <li>• Subred</li> <li>• Máscara de red o Bit de longitud</li> <li>• Inicio</li> <li>• Fin</li> </ul> <p>Donde los campos de Inicio y Fin se utilizan para llenar la lista de rangos de la subred.</p> <p>4. El sistema comprueba en la lista de subredes que la subred insertada por el administrador no exista.</p> <p>5. El sistema comprueba que los datos hayan sido insertados en el formato correcto.</p> <p>6. El sistema comprueba que no existan campos vacíos.</p> <p>7. Adiciona la subred al servidor DHCP.</p> <p>8. El sistema muestra un mensaje de confirmación indicando que la red ha sido insertada correctamente y el caso de uso termina.</p> |
| <b>Prototipo de Interfaz</b>  |   |





### Flujos Alternos “4a Existencia de la subred”

| Acción del Actor | Respuesta del Sistema  |
|------------------|--|
|                  | <p>4a1.El sistema muestra un mensaje indicando que la subred que intenta insertar ya existe en el servidor y regresa a la acción 3 del flujo de eventos.</p> |

### Prototipo de Interfaz

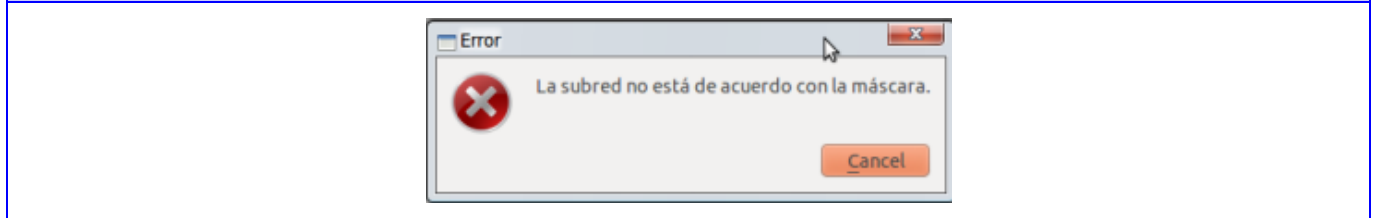


### Flujos Alternos “5a Formato incorrecto”

| Acción del Actor | Respuesta del Sistema                              |
|------------------|--|
|                  | <p>5a1.El sistema muestra un mensaje indicando</p> |

|  |   |
|--|---|
|  | <p>que el formato está incorrecto, mostrando alguno de los siguientes mensajes.</p> <ul style="list-style-type: none"> <li>• Existen rangos en conflicto.</li> <li>• La subred no está de acuerdo con la máscara.</li> <li>• Los rangos están incorrectos.</li> <li>• Subred Incorrecta.</li> </ul> <p>5a2. Regresa a la acción 3 del flujo de eventos.</p> |
|--|---|

**Prototipo de Interfaz**



**Flujos Alternos “6a Campos vacíos.”**

| Acción del Actor | Respuesta del Sistema  |
|------------------|--|
|                  | <p>6a1.El sistema muestra un mensaje indicando que existe un error en el IP de la Subred y regresa a la acción 3 del flujo de eventos.</p> |

**Prototipo de Interfaz**



**Flujos Alternos “3a Cancelar”**

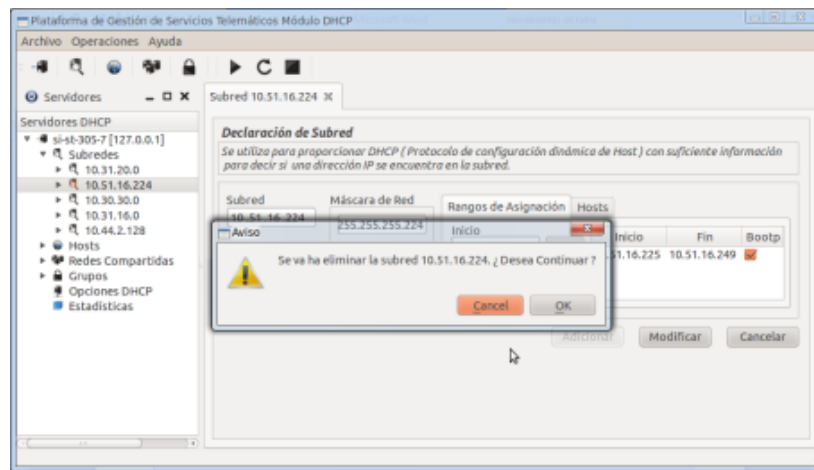
| Acción del Actor                                 | Respuesta del Sistema                                  |
|--|--|
| <p>4a 1. El administrador presiona Cancelar.</p> | <p>3a 1. El sistema limpia todos los campos que el</p> |

|  |   |
|--|---|
|  | administrador había insertado anteriormente y regresa a la acción 3 del flujo de eventos. |
|--|---|

**Sección “Eliminar una subred”**

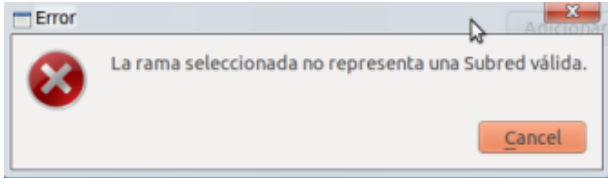
| Acción del Actor                                  | Respuesta del Sistema   |
|---|---|
| 3. El administrador selecciona la opción Aceptar. | <ol style="list-style-type: none"> <li>1. El sistema comprueba que el administrador haya seleccionado una subred.</li> <li>2. El sistema emite un mensaje de confirmación para saber si el administrador de red está seguro de la operación a realizar.</li> <li>4. El sistema elimina la red.</li> <li>5. El sistema muestra un mensaje indicando que la operación ha sido realizada satisfactoriamente y termina el caso de uso.</li> </ol> |

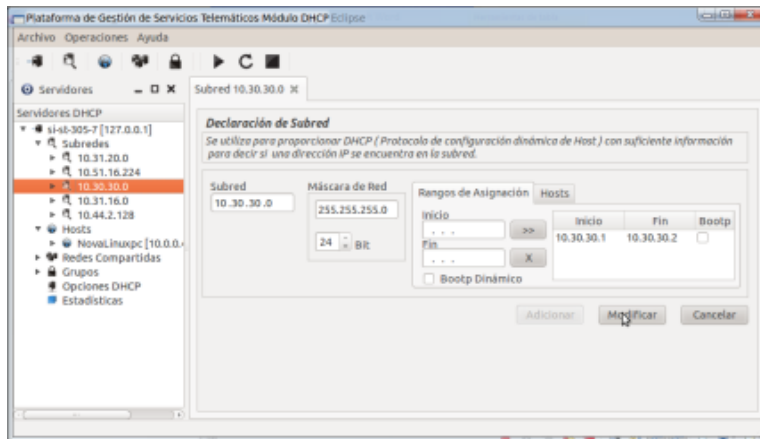
**Prototipo de Interfaz**



**Flujos Alternos “3a Cancelar”**

| Acción del Actor                            | Respuesta del Sistema         |
|---|-------------------------------|
| 3a 1. El administrador de red selecciona la | 3a 2. El caso de uso termina. |

|  |   |
|--|---|
| opción Cancelar.   |   |
| <b>Flujos Alternos “1a Subred no seleccionada”</b>   |   |
| <b>Acción del Actor</b>  | <b>Respuesta del Sistema</b>  |
|  | 1a 1. El sistema emite un mensaje indicando que la rama seleccionada no representa una subred válida y regresa a la acción 2 del flujo de eventos.  |
| <b>Prototipo de Interfaz</b>   |   |
|    |   |
| <b>Sección “Modificar una subred”</b>  |   |
| <b>Acción del Actor</b>  | <b>Respuesta del Sistema</b>  |
| <ol style="list-style-type: none"> <li>1. El administrador modifica los campos y la lista de rangos de la subred que desea.</li> <li>2. El administrador selecciona la opción Modificar en la interfaz.</li> </ol> | <ol style="list-style-type: none"> <li>3. El sistema comprueba en la lista de subredes que la subred insertada por el administrador no exista.</li> <li>4. El sistema comprueba que los datos hayan sido insertados en el formato correcto.</li> <li>5. El sistema comprueba que no existan campos vacíos.</li> <li>6. El sistema muestra un mensaje de confirmación indicando que la red ha sido modificada correctamente y el caso de uso termina.</li> </ol> |
| <b>Prototipo de Interfaz</b>   |   |



### Flujos Alternos “3a Existencia de la subred”

| Acción del Actor | Respuesta del Sistema   |
|------------------|---|
|                  | <p>3a1.El sistema muestra un mensaje indicando que la subred ya existe en el servidor y regresa a la acción 3 del flujo de eventos.</p> |

### Prototipo de la Interfaz

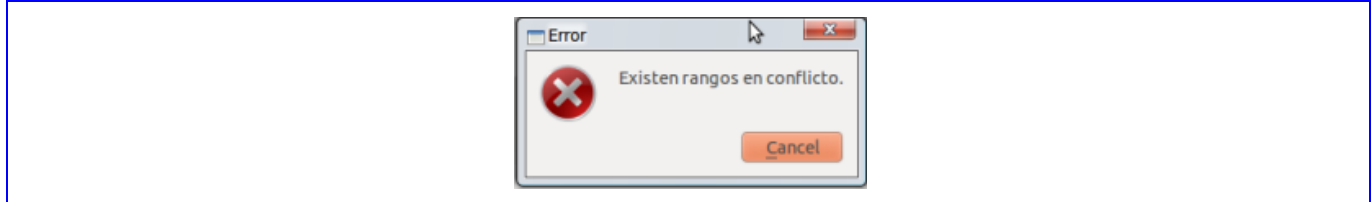


### Flujos Alternos “4a Formato incorrecto”

| Acción del Actor | Respuesta del Sistema  |
|------------------|--|
|                  | <p>4a1.El sistema muestra un mensaje indicando que el formato está incorrecto mostrando alguno de los siguientes mensajes.</p> <ul style="list-style-type: none"> <li>• Existen rangos en conflicto.</li> <li>• La subred no está de acuerdo con la</li> </ul> |

|  |   |
|--|---|
|  | <p>máscara.</p> <ul style="list-style-type: none"> <li>• Los rangos están incorrectos.</li> <li>• Subred incorrecta.</li> </ul> <p>4a2. Regresa a la acción 3 del flujo de eventos.</p> |
|--|---|

**Prototipo de la Interfaz**



**Flujos Alternos “5a Campos vacíos.”**

| Acción del Actor | Respuesta del Sistema  |
|------------------|--|
|                  | <p>5a1.El sistema muestra un mensaje indicando que existe un error en el IP de la Subred y regresa a la acción 3 del flujo de eventos.</p> |

**Prototipo de Interfaz**



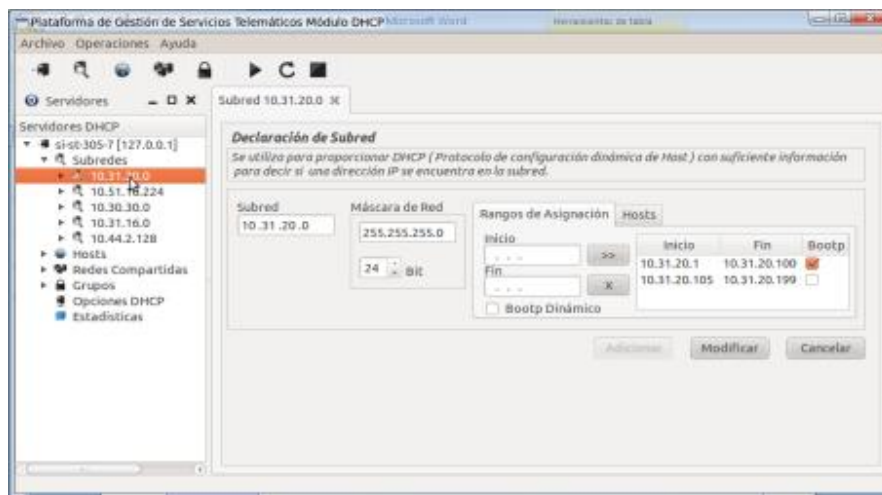
**Flujos Alternos “4a Cancelar”**

| Acción del Actor                                 | Respuesta del Sistema  |
|--|--|
| <p>4a 1. El administrador presiona Cancelar.</p> | <p>4a 1.1 El sistema restablece todos los campos con los valores que tenía antes de hacer cualquier cambio y regresa a la acción 3 del flujo de eventos.</p> |

**Sección “Mostrar una subred”**

| Acción del Actor | Respuesta del Sistema  |
|------------------|--|
|                  | <p>1. El sistema muestra la subred con los datos correspondientes:</p> <ul style="list-style-type: none"> <li>• Subred</li> <li>• Máscara de Red</li> <li>• Rangos de IP</li> </ul> <p>El caso de uso termina.</p> |

**Prototipo de Interfaz**



|                       |  |
|-----------------------|--|
| <b>Poscondiciones</b> |  |
|-----------------------|--|

**2.7.3.2 Descripción del Casos de Uso Gestionar Grupo**

|                    |   |
|--------------------|---|
| <b>CU 3</b>        | Gestionar Grupo   |
| <b>Actor</b>       | Administrador   |
| <b>Descripción</b> | Este CU permite al administrador agregar nuevos Grupos al servidor. Podrá además modificar o eliminar dichos Grupos en caso que lo necesite. El Caso de Uso termina con la gestión exitosa de los Grupos del servidor DHCP. |

|                   |           |
|-------------------|-----------|
| <b>Referencia</b> | RF3, RF14 |
| <b>Prioridad</b>  | Media     |
|                   |           |

### 2.7.3.3 Descripción del Casos de Uso Gestionar Red Compartida

|                    |  |
|--------------------|--|
| <b>CU 4</b>        | Gestionar Red Compartida   |
| <b>Actor</b>       | Administrador de Red   |
| <b>Descripción</b> | Este CU permite al administrador agregar nuevas Redes Compartidas al servidor. Podrá además modificar o eliminar dichas Redes Compartidas en caso que lo necesite. El Caso de Uso termina con la gestión exitosa de las Redes Compartidas del servidor DHCP. |
| <b>Referencia</b>  | RF4, RF13  |
| <b>Prioridad</b>   | Media  |
|                    |  |

### 2.7.3.4 Descripción del Casos de Uso Gestionar Host

|                    |   |
|--------------------|---|
| <b>CU 5</b>        | Gestionar Host  |
| <b>Actor</b>       | Administrador de Red  |
| <b>Descripción</b> | Este CU permite al administrador agregar nuevos Host al servidor. Podrá además modificar o eliminar dichos Host en caso que lo necesite. El Caso de Uso termina con la gestión exitosa de los Host del servidor DHCP. |
| <b>Referencia</b>  | RF5, RF15   |
| <b>Prioridad</b>   | Media   |
|                    |   |



### 2.7.3.5 Descripción del Casos de Uso Gestionar Opciones y Parámetros

|                    |   |
|--------------------|---|
| <b>CU 6</b>        | Gestionar Opciones y Parámetros   |
| <b>Actor</b>       | Administrador de Red  |
| <b>Descripción</b> | Este CU permite al administrador agregar nuevas Opciones y Parámetros al servidor, además los podrá modificar o eliminar en caso que lo necesite. El sistema ofrece la posibilidad de listar todas las Opciones y Parámetros que se encuentran activos. El Caso de uso termina con la gestión exitosa de las opciones y los parámetros. |
| <b>Referencia</b>  | RF6, RF17   |
| <b>Prioridad</b>   | Media   |

### 2.7.3.6 Descripción del Casos de Uso Nuevo Servidor Remoto

|                    |  |
|--------------------|--|
| <b>CU 7</b>        | Nuevo Servidor Remoto  |
| <b>Actor</b>       | Administrador de Red   |
| <b>Descripción</b> | Este CU permite al administrador agregar nuevos servidores DHCP a la aplicación para realizar su gestión de manera remota. Para adicionar un nuevo servidor a la aplicación se le debe especificar la dirección IP, usuario y contraseña para establecer una conexión con la PC donde se encuentra el servidor por el protocolo SSH. |
| <b>Referencia</b>  | RF7, RF16  |
| <b>Prioridad</b>   | Baja   |

### 2.7.3.7 Descripción del Casos de Uso Preparar integración DHCP con DNS

|                        |   |
|------------------------|---|
| <b>CU 8</b>            | Preparar integración DHCP con DNS   |
| <b>Actor</b>           | Administrador de Red  |
| <b>Descripción</b>     | El caso de uso permite preparar el escenario para una integración de los servicios DHCP y DNS, permitiendo que el servidor DHCP actualice la información de las zonas en el Servidor de nombres de dominio cuando el servidor DHCP realiza una nueva concesión. Dicho caso de uso brinda la posibilidad de adicionar las zonas necesarias para realizar la integración y modificar los parámetros necesarios del servidor DHCP. |
| <b>Precondiciones:</b> | El administrador debe estar autenticado. Que exista un servidor DNS con las zona que se van actualizar.   |
| <b>Referencia</b>      | RF8   |
| <b>Prioridad</b>       | Alta  |

### 2.7.3.8 Descripción del Casos de Uso Mostrar Reportes

|                        |  |
|------------------------|--|
| <b>CU 9</b>            | Mostrar Reportes   |
| <b>Actor</b>           | Administrador de Red   |
| <b>Descripción</b>     | El caso de uso se inicia cuando el administrador de red desea ver los reportes estadísticos de uso de un servidor DHCP y termina cuando el sistema muestra una interfaz con los reportes cargados. |
| <b>Precondiciones:</b> | El administrador debe estar autenticado y este seleccionado el servidor DHCP para mostrar sus reportes.  |
| <b>Referencia</b>      | RF11   |
| <b>Prioridad</b>       | Media  |

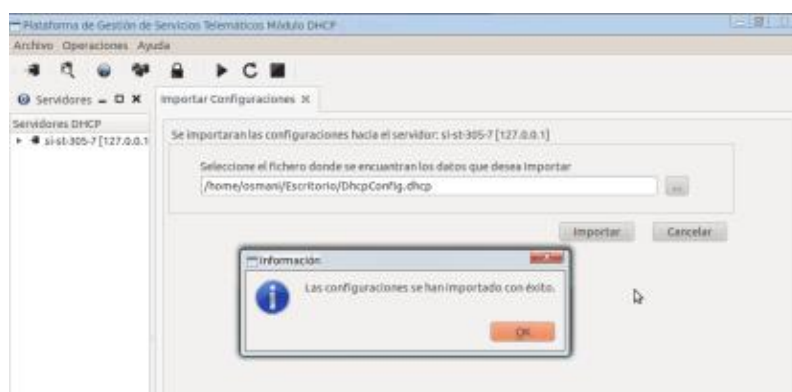
**2.7.3.9 Descripción del Casos de Uso Exportar Configuraciones**

|                        |  |
|------------------------|--|
| <b>CU 10</b>           | Exportar Configuraciones   |
| <b>Actor</b>           | Administrador de Red   |
| <b>Descripción</b>     | El caso de uso se inicia cuando el administrador de red decide exportar las configuraciones de un servidor hacia un fichero, donde el actor debe especificar la ruta donde se quiere guardar el fichero, el caso de uso termina cuando el sistema muestra un mensaje de confirmación indicando que la operación se realizó satisfactoriamente. |
| <b>Precondiciones:</b> | El administrador debe estar autenticado. Este seleccionado el servidor de el cual se quiere exportar las configuraciones.  |
| <b>Referencia</b>      | RF10   |
| <b>Prioridad</b>       | Media  |

**2.7.3.10 Descripción de Casos de Uso Importar Configuraciones**

|                        |  |
|------------------------|--|
| <b>CU 11</b>           | Importar configuraciones DHCP.   |
| <b>Actores:</b>        | Administrador de red   |
| <b>Resumen:</b>        | El caso de uso se inicia cuando el administrador de red decide Importar configuraciones DHCP ya sean configuraciones anteriormente exportadas de un servidor Windows Server 2003 o del propio servidor de GNU/Linux y el caso de uso finaliza cuando el sistema muestra un mensaje de confirmación indicando que la operación se realizó satisfactoriamente. |
| <b>Precondiciones:</b> | El administrador debe estar autenticado y debe seleccionar el servidor hacia el  |

|  |  |  |
|--|--|--|
|  | cual se va a importar las configuraciones, además deben existir las configuraciones en un fichero previamente exportadas.  |  |
| <b>Referencias</b>   | RF9, RF1   |  |
| <b>Prioridad</b>   | <b>Alta</b>  |  |
| <b>Flujo Normal de Eventos</b>   |  |  |
| <b>Sección “Importar configuraciones DHCP.”</b>  |  |  |
| <b>Acción del Actor</b>  | <b>Respuesta del Sistema</b>   |  |
| <p>1. El administrador selecciona la opción Importar Configuraciones en el menú Operaciones.</p> <p>3. El administrador especifica la ruta de las configuraciones o la selecciona del cuadro de diálogo Abrir.</p> <p>4. El administrador oprime el botón Importar para realizar la operación.</p> | <p>2. El sistema muestra una interfaz para que especifique la ruta donde se encuentran las configuraciones a Importar.</p> <p>5. El sistema comprueba que exista la ruta especificada.</p> <p>6. El sistema comprueba que las configuraciones sean correctas.</p> <p>7. El sistema importa los datos hacia el Servidor DHCP de GNU/Linux previamente seleccionado.</p> <p>8. El sistema muestra un mensaje de Importación satisfactoria.</p> |  |
| <b>Prototipo de Interfaz</b>   |  |  |



**Flujos Alternos 5a “Ruta Incorrecta”**

| Acción del Actor | Respuesta del Sistema  |
|------------------|--|
|                  | 5a.1 El sistema muestra un mensaje informando que la ruta no existe y regresa a la acción 2 del flujo normal de eventos. |

**Prototipo de Interfaz**



**Flujos Alternos 4a “Cancelar”**

| Acción del Actor   | Respuesta del Sistema                                   |
|--|---|
| 4a.1 El administrador Cancela la Importación de los datos. | 4a.2 Regresa a la acción 2 del flujo normal de eventos. |

**Flujos Alternos 6a “Datos Incorrectos”**

| Acción del Actor | Acción del Sistema |
|------------------|--------------------|
|------------------|--------------------|

6a.1 El sistema muestra un mensaje informando que existe un error en los datos importados y regresa a la acción 2 del flujo normal de eventos del sistema.

**Prototipo de Interfaz**



**2.8 Conclusiones:**

En el presente capítulo se presentaron los procesos del negocio y se realizó una descripción detallada de cada uno de dichos procesos, estos sirvieron de guía para identificar los requisitos funcionales del sistema, detallando cada uno de ellos, también se presentaron los requisitos no funcionales. Luego se obtuvo al Diagrama de Casos de Uso del Sistema junto a una descripción detallada de los CU críticos, para los demás CU se hizo una descripción general. Además se presentó un análisis de la situación problemática y de la información que se maneja para gestionar las configuraciones, y se presentó una propuesta del sistema a desarrollar.

## CAPÍTULO 3: DISEÑO DEL SISTEMA

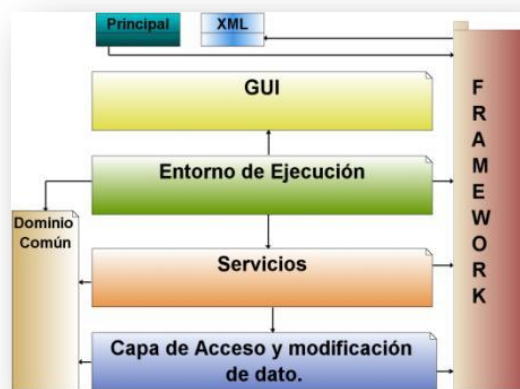
### 3.1 Introducción

En el presente capítulo luego de los resultados obtenidos anteriormente, se profundizará en los casos de usos detallándolos para lograr una vista de la estructura interna del sistema, donde se reflejan las clases necesaria para dar cumplimiento a las funcionalidades del sistema descritos en los casos de uso. Además constituye una actividad de suma importancia para descomponer los trabajos de implementación en partes más manejables. Algunos artefactos que fueron generados en el diseño del sistema se tuvieron en cuenta, a partir de los cuales se desarrolló la implementación del sistema en términos de componentes, es decir, ficheros de código fuente, scripts, ficheros de código binario, ejecutables y similares.

El presente capítulo tiene como objetivo la realización del Diagrama de Clases del Diseño de cada caso de uso y los diagramas de componentes, además se representa el diagrama de despliegue que contiene la topología hardware sobre la que se ejecuta el sistema.

### 3.2 Modelo Diseño

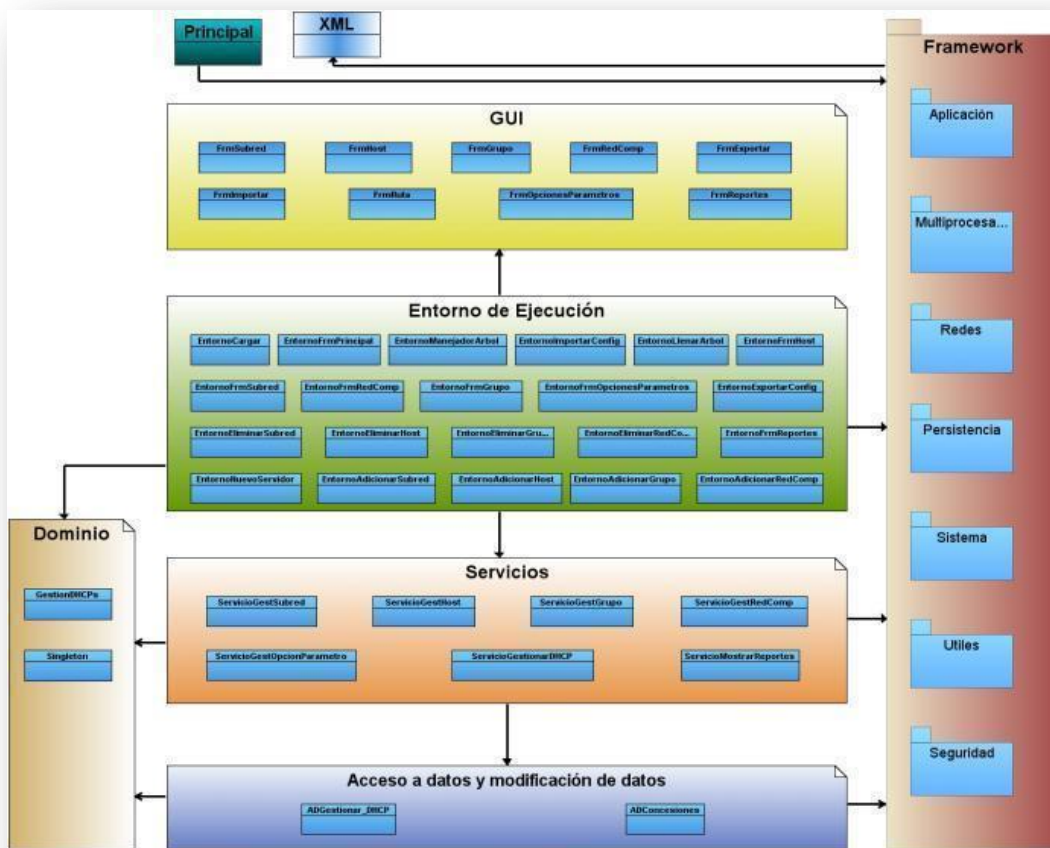
El diseño está estrechamente ligado a la arquitectura que se utiliza para el desarrollo. Siendo la arquitectura de n capas la seleccionada para la elaboración del Módulo DHCP, la cual posee varias ventajas ya que permiten la descentralización del sistema, organiza el modelo de diseño en capas y mejora la detección y corrección de errores, además de ofrecer una mayor transparencia y entendimiento del sistema. La arquitectura en capas reduce las dependencias, lo cual se refleja en que los componentes de una capa sólo pueden hacer referencia a componentes en capas inmediatamente inferior o superior. En los siguientes diagramas se puede apreciar la estructura general del sistema en términos de capas y componentes:



**Estructura general del sistema**

Se utilizó el patrón de arquitectura n-capas, por lo que se identificaron cuatro capas principales ellas son: GUI, Entorno de Ejecución, Servicios, y Acceso y Modificación de Datos; además existen varios elementos importantes los cuales se van a tratar a continuación. En primer lugar se decidió implementar un Framework para resolver problemas puntuales relacionados con la iniciación del sistema, la instalación desde fuentes, la ejecución de funciones en paralelo, las conexiones remotas, además de un grupo de librerías que se utilizan en el desarrollo del sistema, entre ellas se encuentra la librería pyparsing utilizada en el parseo de ficheros de configuración, esta cuenta con una capa de abstracción para facilitar el trabajo con la misma; con el objetivo de poder reutilizar estas funcionalidades en futuros desarrollos de software y proporcionar al desarrollador la posibilidad de concentrarse en los aspectos exclusivos de su negocio. La configuración de la estructura general del sistema estará reflejada en un archivo XML que será utilizado por el framework para conocer que clases serán empleadas y donde residirá su implementación particular. Otro aspecto importante es comprender el funcionamiento del componente Dominio Común, el mismo contiene entidades o clases que serán reutilizados por más de una clase en el sistema.

### 3.2.1 Estructura detallada del Sistema





### 3.3 Diagrama de clases del diseño.

Los diagramas a continuación muestran directamente cada uno de los CU reflejando el uso del patrón n capas y sus relaciones entre las clases del diseño, para los cual se definió un color para identificar a que capa pertenece cada clase del diseño. A continuación se describen la relación entre colores y capas:



Capa GUI donde se encuentran todo lo relacionado con las interfaces de usuario.



Capa Entornos es la intermediaria entre la capa GUI y Servicios

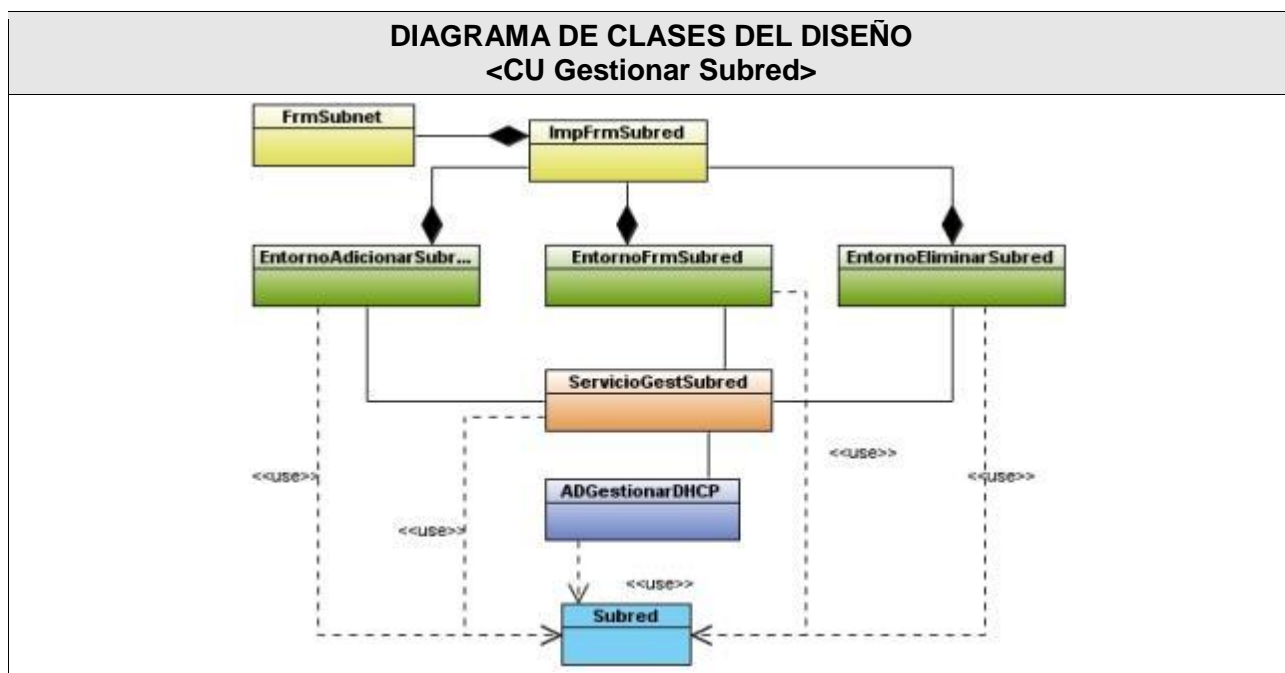


Capa Servicios se encuentra el negocio de la aplicación.

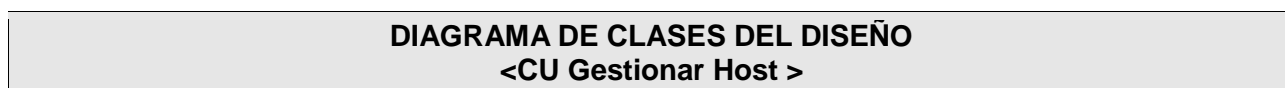


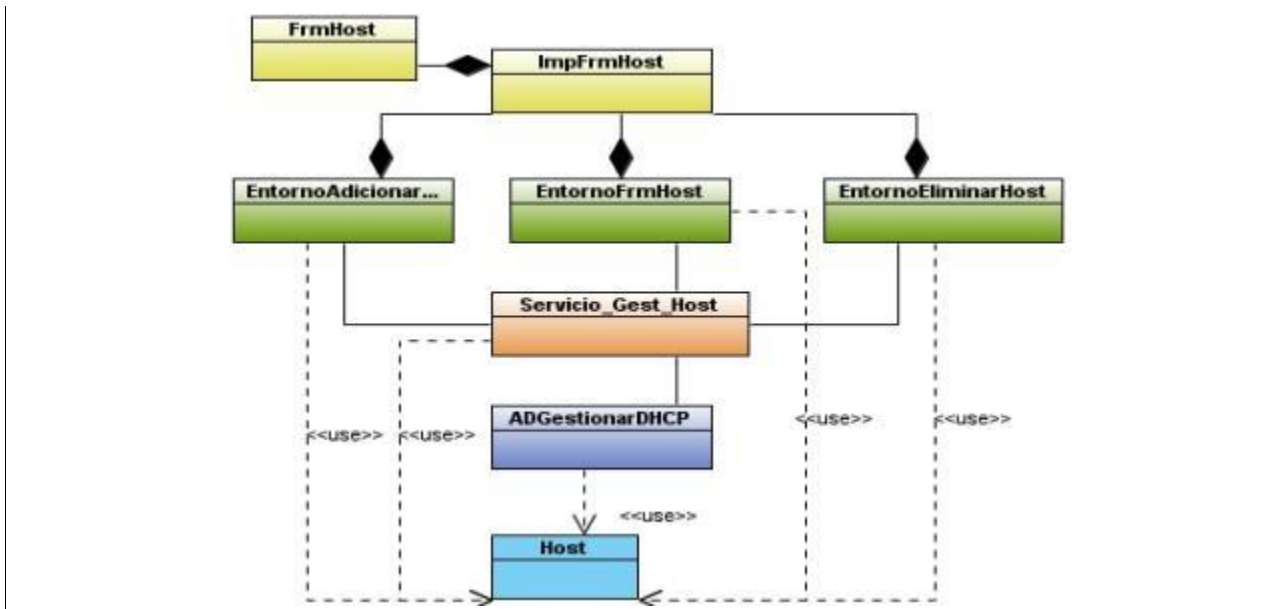
Capa Acceso y modificación de datos.

#### 3.3.1 Diagrama de Clases del diseño. CU Gestionar Subred

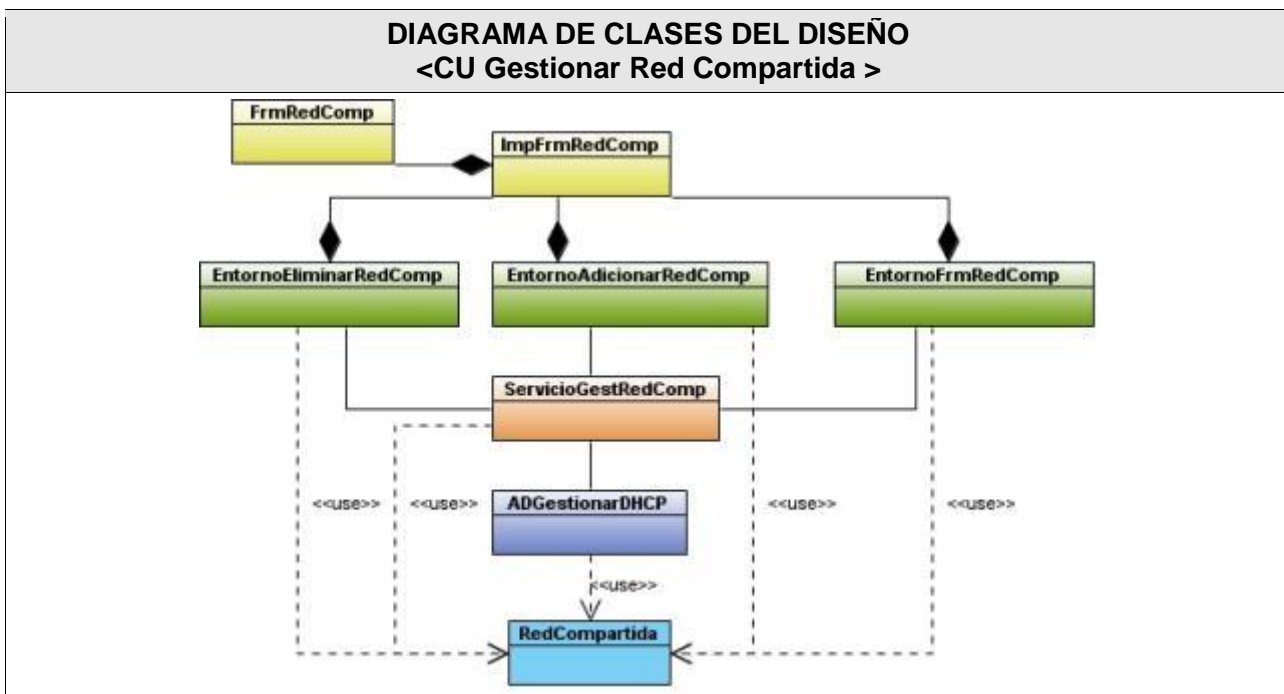


#### 3.3.2 Diagrama de Clases del diseño. CU Gestionar Host

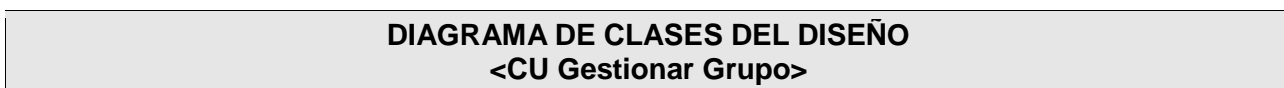


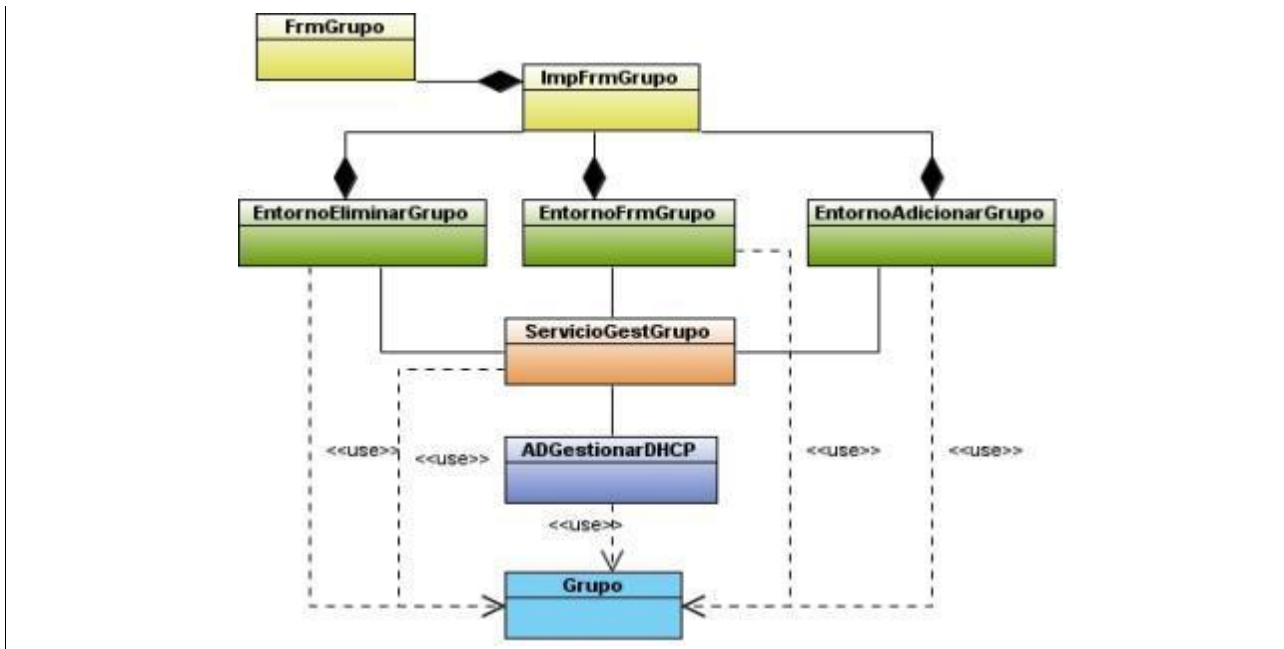


3.3.3 Diagrama de Clases del diseño. CU Gestionar Red Compartida

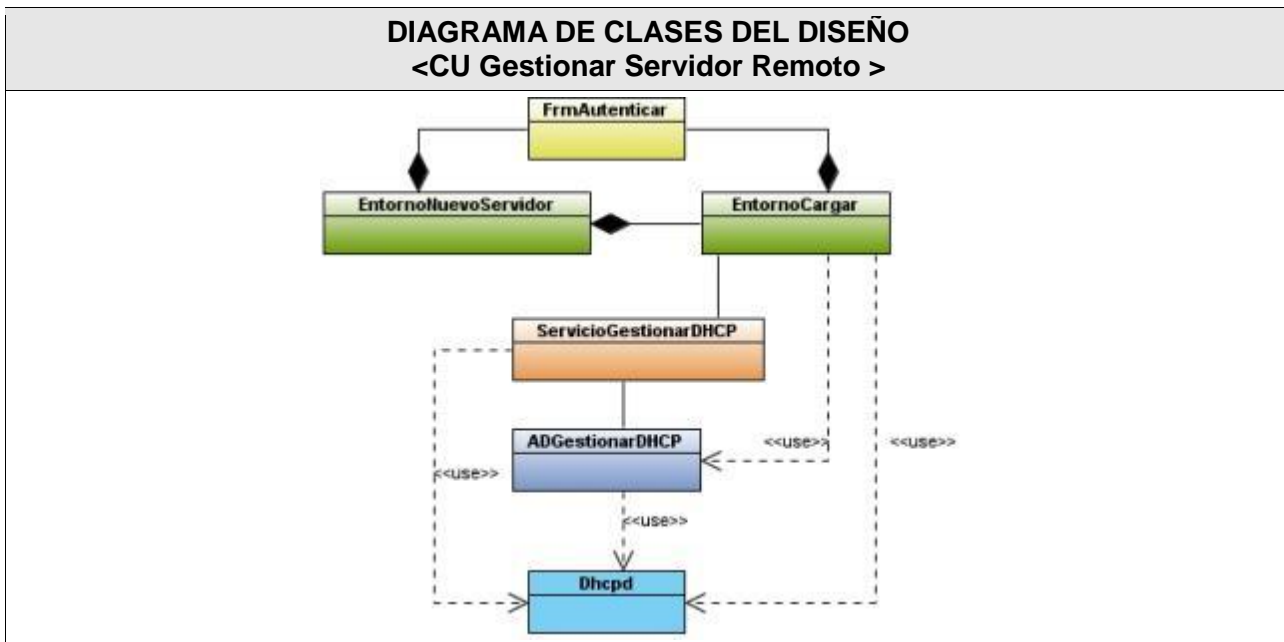


3.3.4 Diagrama de Clases del diseño. CU Gestionar Grupo

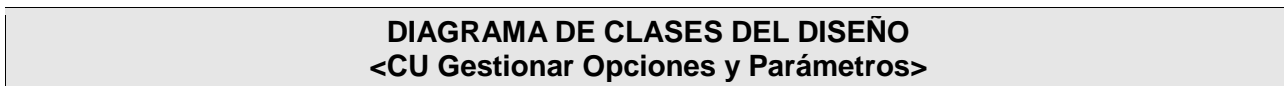


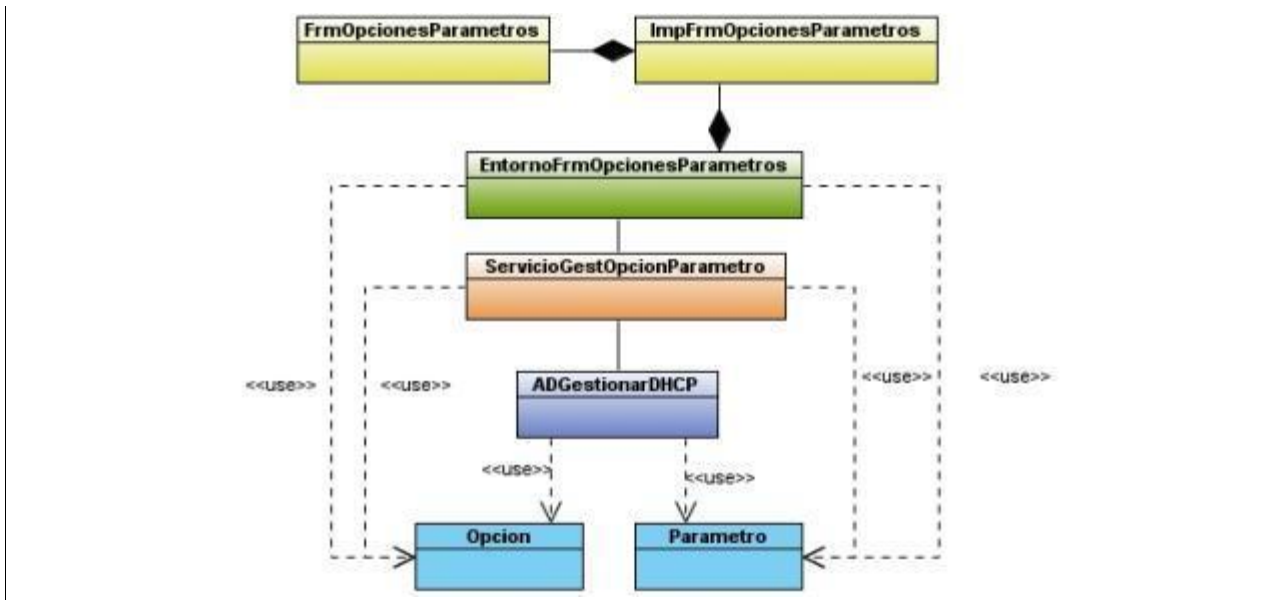


3.3.5 Diagrama de Clases del diseño. CU Gestionar Servidor

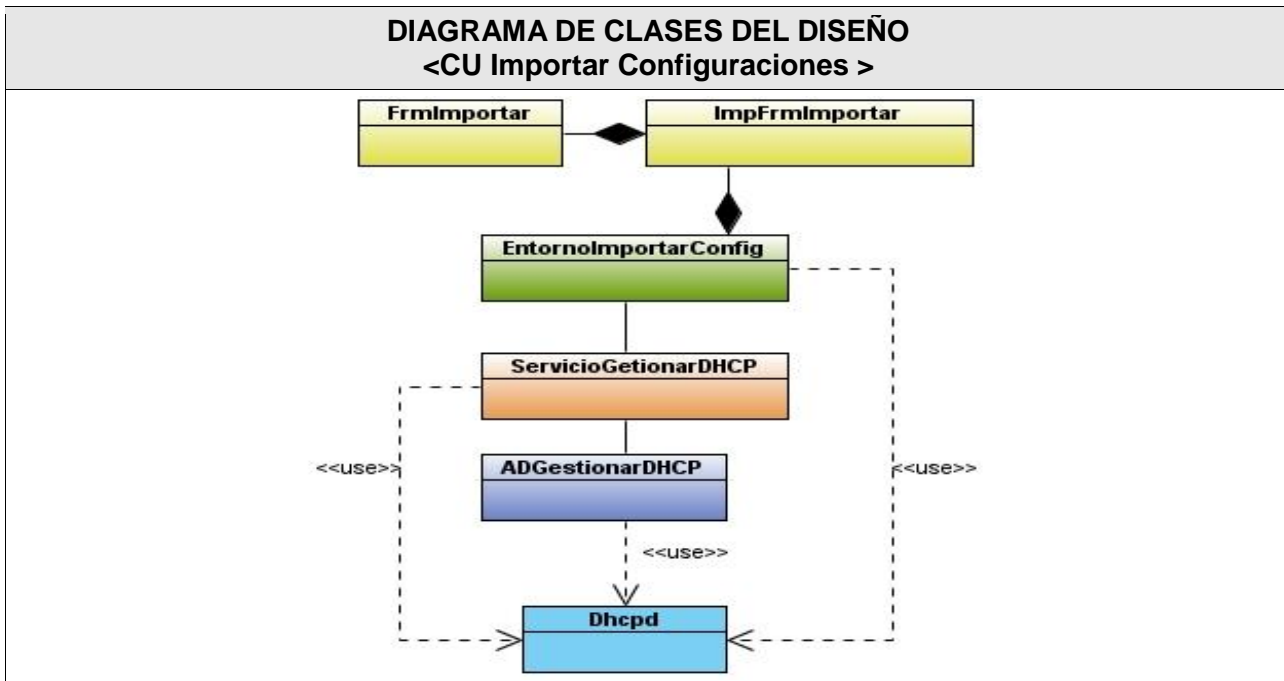


3.3.6 Diagrama de Clases del diseño. CU Gestionar Opciones y Parámetros

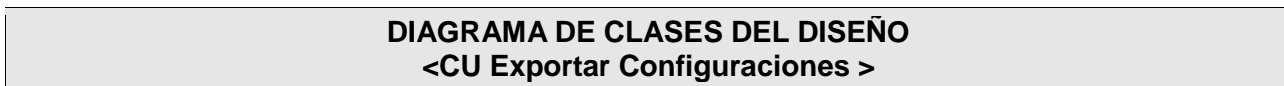


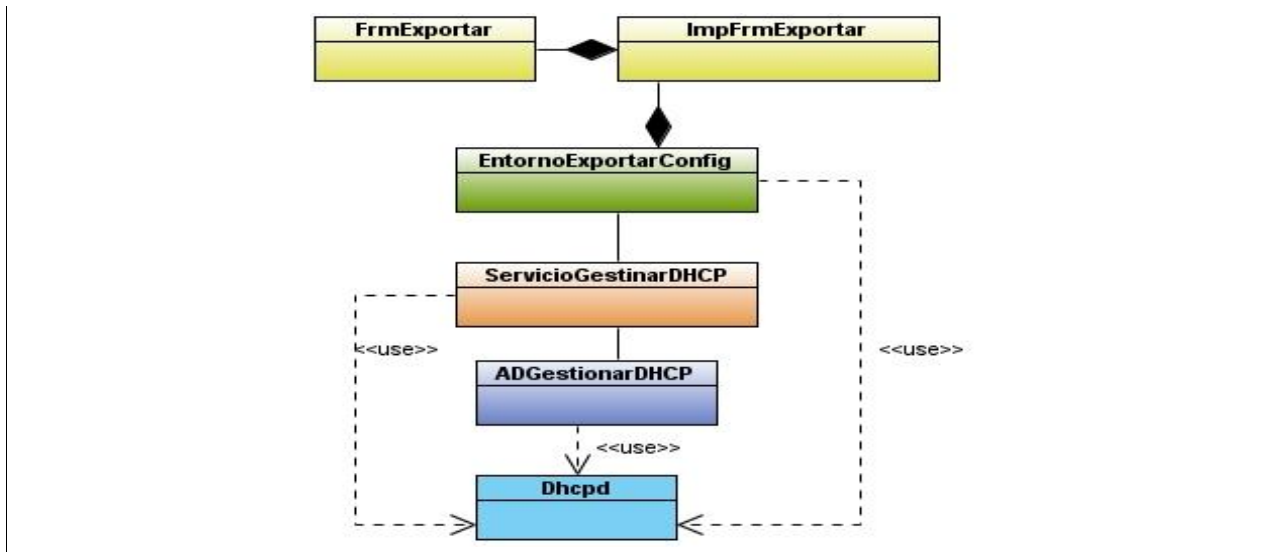


3.3.7 Diagrama de Clases del diseño. CU Importar Configuraciones

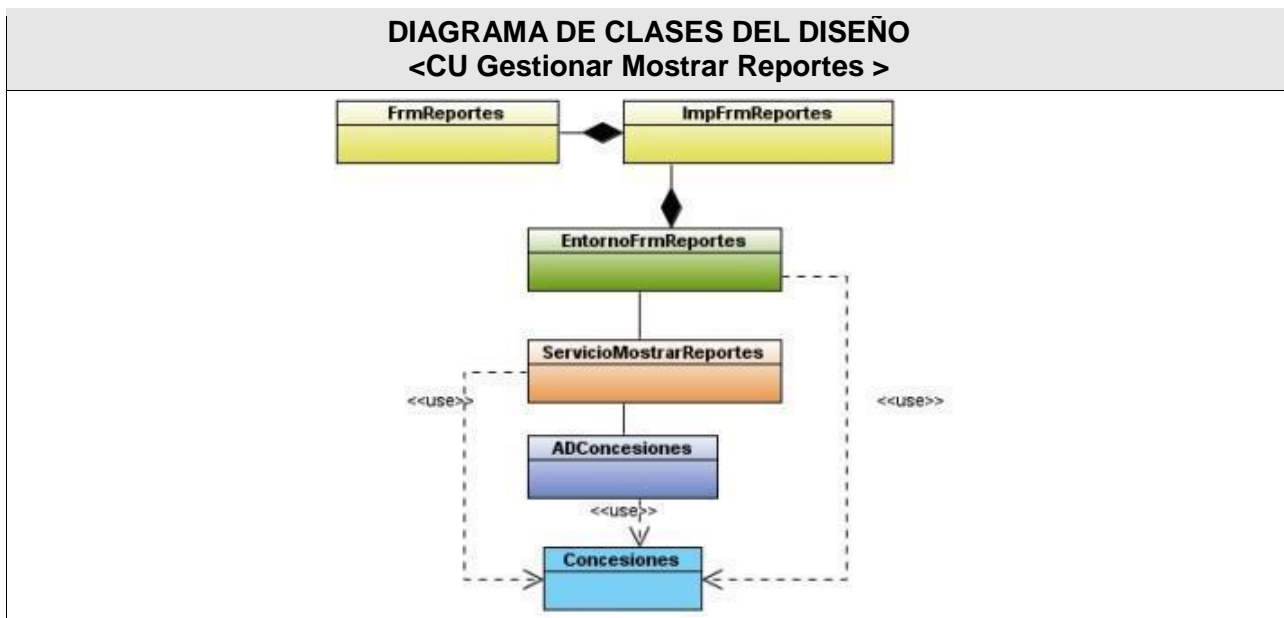


3.3.8 Diagrama de Clases del diseño. CU Exportar Configuraciones





3.3.9 Diagrama de Clases del diseño. CU Mostrar Reportes



### 3.4 Conclusiones

En el presente capítulo se obtuvieron los artefactos fundamentales correspondientes al Diseño del sistema, como son los modelos de diseño, los cuales constituyen la entrada principal para la futura implementación del sistema. Además se presentó la arquitectura utilizada en la construcción del sistema, en la cual se realizó una explicación detallada. Se explicó la nueva capa que fue creada, buscando que la aplicación sea lo más escalable y extensible posible.

## CAPÍTULO 4: IMPLMANTACIÓN Y PRUEBAS

### 4.1 Introducción

En el presente capítulo se realiza la fase de implementación en la cual se comienza el desarrollo del sistema en términos de componentes, es decir, ficheros de código fuente, ficheros de código binario y ejecutables; todo esto a partir de los resultados del capítulo anterior de Diseño del Sistema. Además se realiza el diagrama de despliegue para situar los componentes lógicos desarrollados sobre el hardware que lo va a contener.

### 4.2 Diagrama de Despliegue.

El diagrama de despliegue muestra los componentes lógicos de la aplicación desarrollada en su situación física, ubica el software desarrollado en el hardware que lo contiene. Este diagrama está formado por un grafo de nodos unidos por conexiones de comunicación. Estos nodos en general son una unidad de cómputo de algún tipo.

#### 4.2.1 Descripción de los nodos del diagrama de despliegue.

| Nombre del Nodo | Descripción   |
|-----------------|---|
| PC              | PC en la que se ejecutará la aplicación y se realizan las operaciones de gestión del servidor DHCP. |
| Servidor DHCP   | Máquina en la cual se encuentra instalado el servidor DHCP de GNU/Linux.                            |

#### 4.2.2 Diagrama de despliegue.



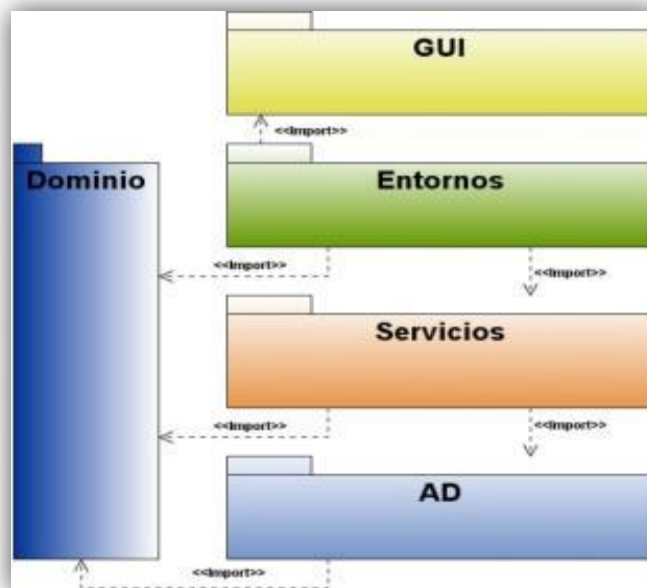
### 4.3 Diagramas de Componentes.

Luego de la Fase de Elaboración y de establecer la línea base de la arquitectura del sistema, la Fase de Construcción tiene como propósito completar el desarrollo del sistema basándose en dicha línea base. En

esta fase se comienza el desarrollo de los diagramas de componentes para mostrar las dependencias lógicas entre componentes del software. A continuación se hace una breve descripción de los paquetes existentes.

- El paquete “GUI” contiene la definición e implementación de los formularios que se utilizan, y dentro se encuentra el paquete UI con los respectivos formularios.
- El paquete “Entornos” contiene los entornos correspondientes a los formularios que constituyen intermediarios entre GUI y Servicios.
- El paquete “Servicios” contiene los componentes del negocio de la aplicación.
- El paquete “AD” contiene componentes encargados de acceder y modificar los datos del servidor.
- El paquete “Dominio” contiene un grupo de componentes que son reutilizados por la mayoría de las capas del sistema.

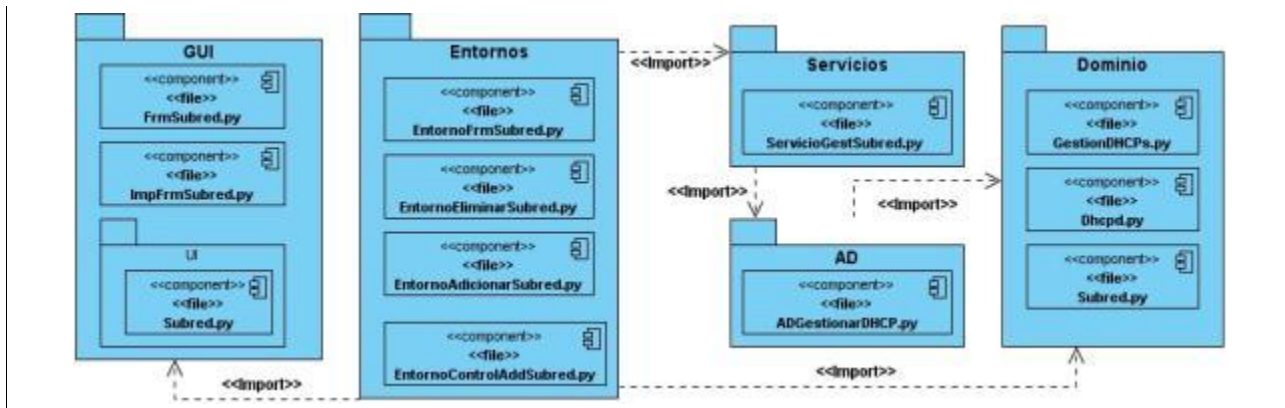
#### 4.3.1 Diagrama General de Componentes



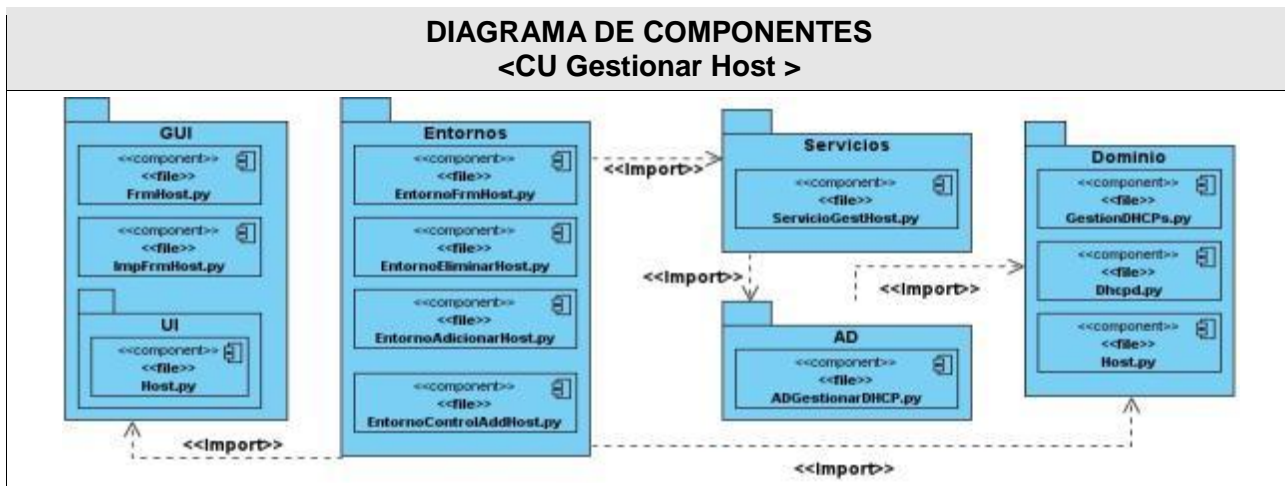
#### 4.3.2 Diagrama de Componentes. CU Gestionar Subred.

DIAGRAMA DE COMPONENTES  
<CU Gestionar Subred >

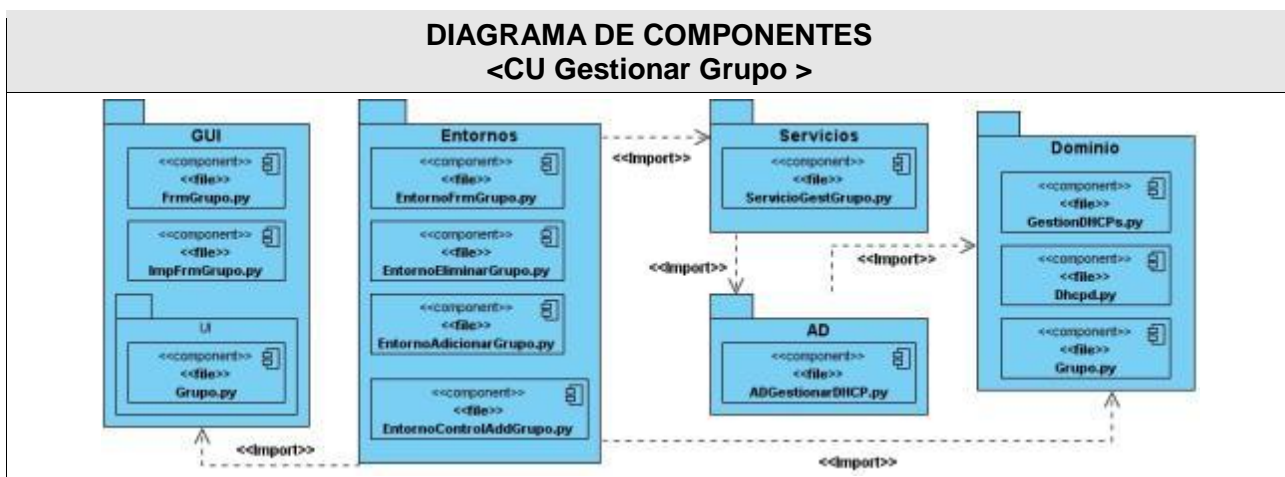




4.3.3 Diagrama de Componentes. CU Gestionar Host.

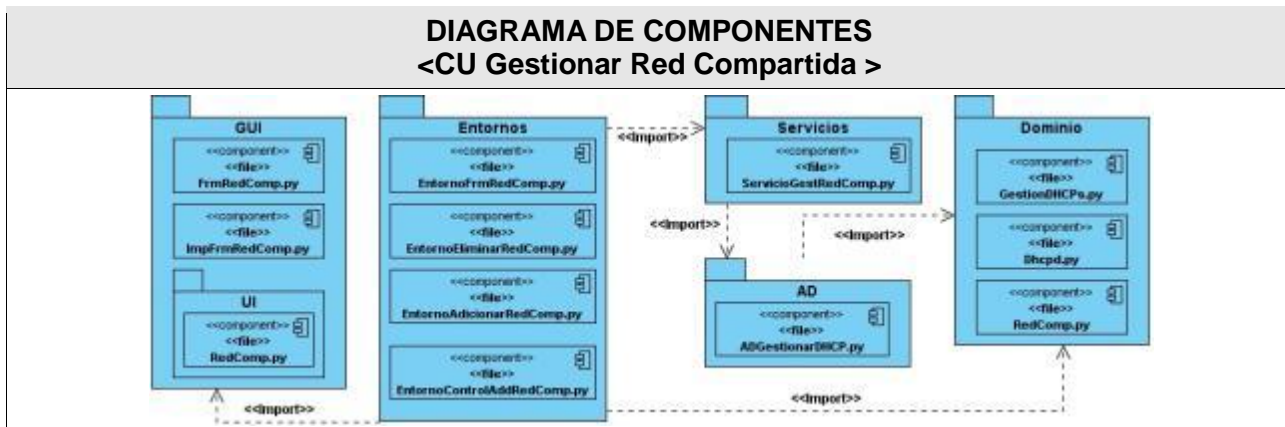


4.3.4 Diagrama de Componentes. CU Gestionar Grupo.

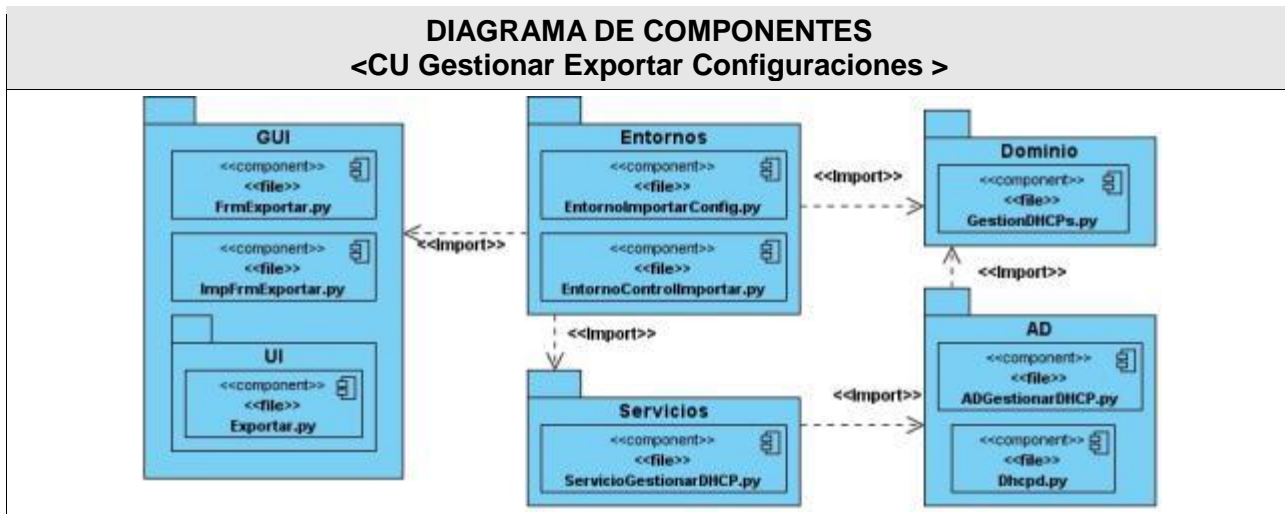


4.3.5 Diagrama de Componentes. CU Gestionar Red Compartida.

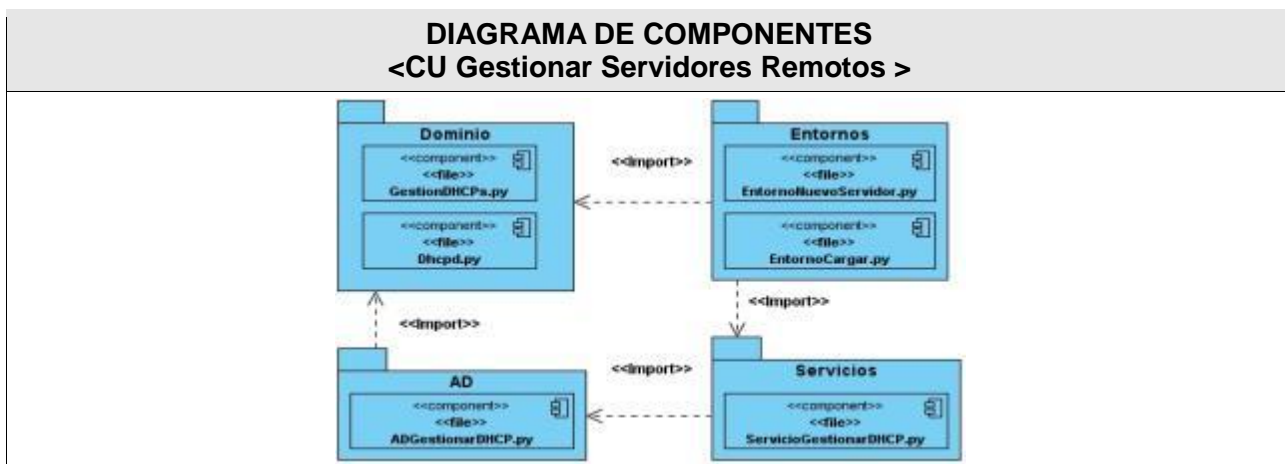




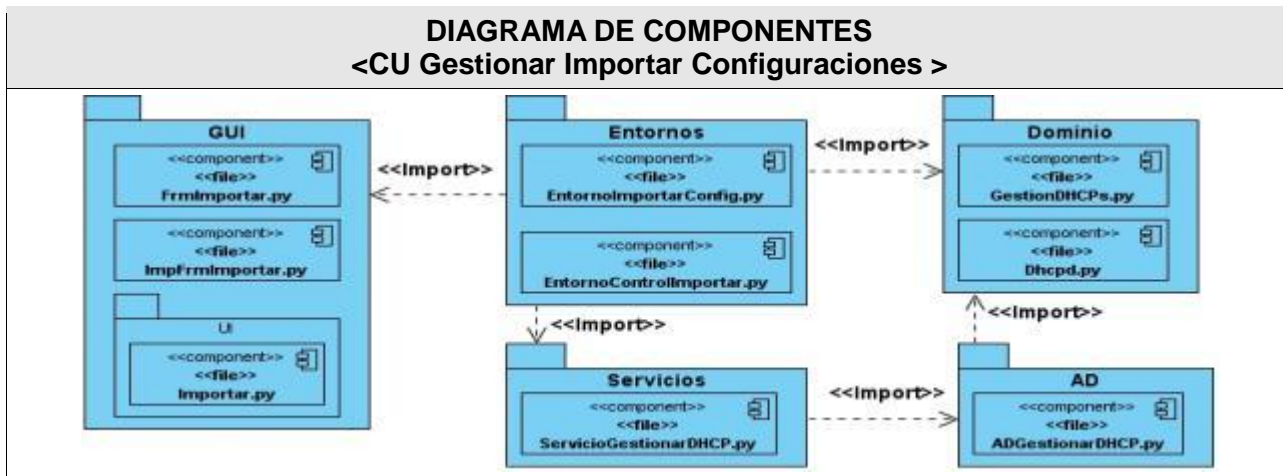
4.3.6 Diagrama de Componentes. CU Exportar Configuraciones.



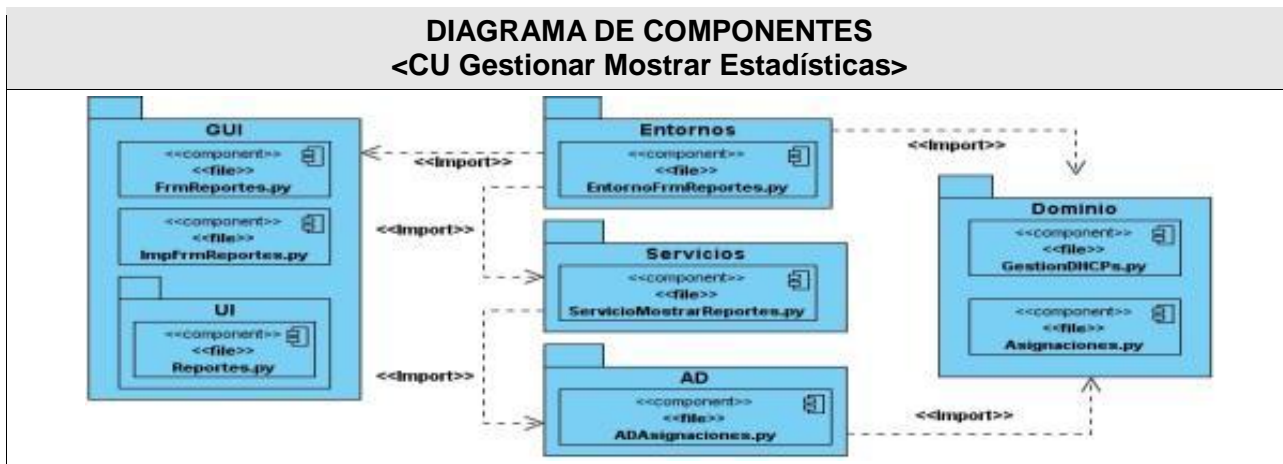
4.3.7 Diagrama de Componentes. CU Gestionar Servidores Remotos.



4.3.8 Diagrama de Componentes. CU Importar Configuraciones.



4.3.9 Diagrama de Componentes. CU Mostrar Estadísticas.



#### 4.4 Pruebas

Las pruebas se llevan a cabo sobre todo cuando una construcción (como un resultado de implementación) es sometida a prueba de integración y de sistema. Luego de la implementación del sistema y para la validación del mismo fueron realizadas las pruebas, definiendo inicialmente una estrategia para ejecución de las pruebas con: niveles, tipos de pruebas, métodos y técnicas correspondientes en busca de errores en la implementación y para perfeccionar el funcionamiento de la aplicación.

A continuación se muestra la estrategia definida a seguir para realizar las pruebas.

De los niveles generales de pruebas:

1. Pruebas Internas.
2. Pruebas Cruzadas.

3. Pruebas de Liberación.
4. Pruebas de Aceptación.
5. Pruebas Piloto.

Se utilizó el nivel de **pruebas internas** para la estrategia definida, donde se encuentran los niveles:

1. Prueba de Desarrollador.
2. Prueba Independiente.
3. Prueba de Unidad.
4. Prueba de Integración.
5. Pruebas de Sistema.

Haciendo uso del **nivel de integración** en la estrategia de prueba a seguir. Para este nivel existen tipos de pruebas basados en dimensiones de calidad, los cuales se mencionarán a continuación:

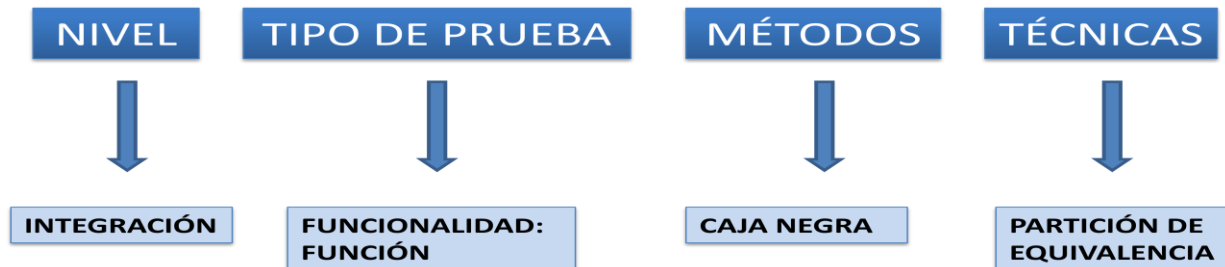
1. Funcionalidad
  - Función
  - Seguridad
2. Usabilidad
3. Fiabilidad
  - Integridad
  - Stress
  - Estructura
4. Rendimiento
  - Carga
  - Contención
  - Benchmark
  - Performance profile
5. Compatibilidad
  - Configuración
  - Instalación

De los tipos de pruebas anteriormente mencionados se utilizó **Funcionalidad Función** en la estrategia a seguir. De los siguientes Métodos de Prueba:

1. Caja Blanca

## 2. Caja Negra

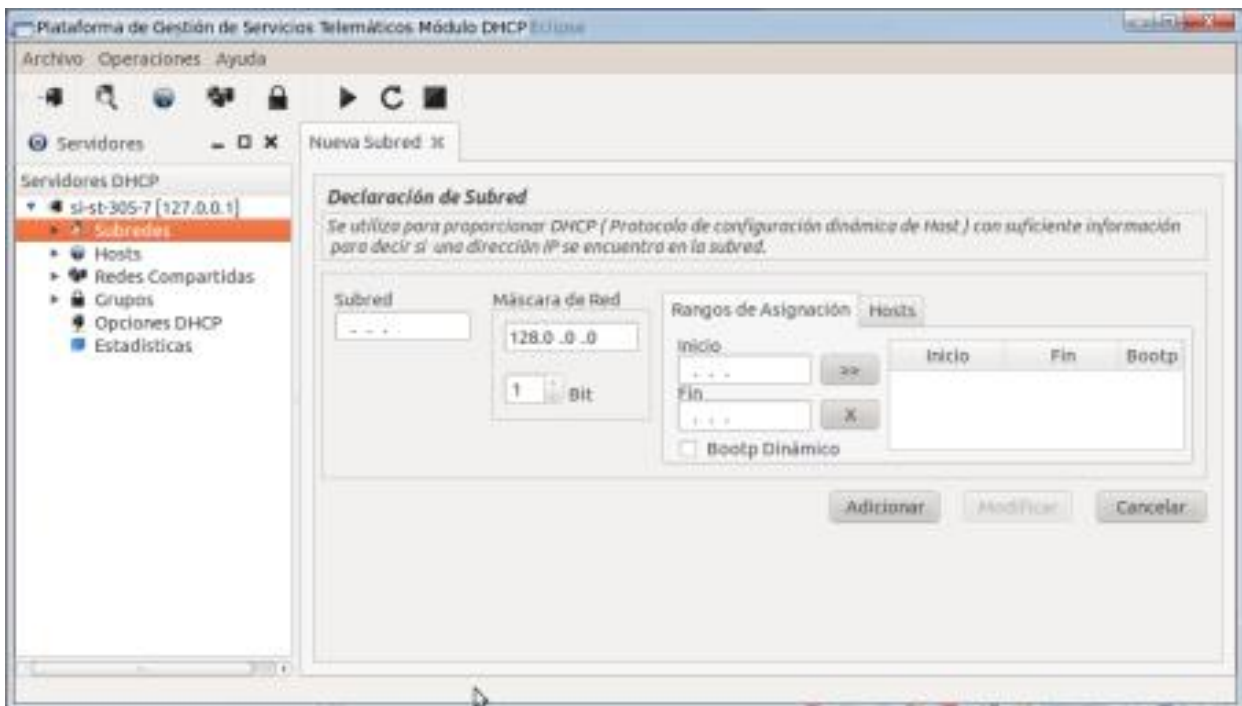
Se emplea **Caja Negra** con la técnica de Partición de Equivalencia dando como resultado final la siguiente estrategia.



### 4.4.1 Diseño del caso de prueba.

Se aplicará la prueba al Caso de Uso Gestionar Subred, el mismo tiene como condición para su ejecución que el Administrador de red debe estar autenticado y debe seleccionar el servidor o la declaración dentro de este a la cual se le van a realizar las pruebas. A continuación se muestra el prototipo de la interfaz del caso de uso en cuestión.

#### 4.4.1.1 Prototipo de interfaz gestionar Subred.



**4.4.1.2 Secciones y Escenarios del Caso de Uso Gestionar Subred.**

| Secciones           | Escenarios                       |
|---------------------|----------------------------------|
| 1. Eliminar subred  | 1.1 Eliminar satisfactoriamente  |
|                     | 1.2 Subred no seleccionada.      |
| 2. Adicionar subred | 2.1 Adicionar satisfactoriamente |
|                     | 2.2 Datos incorrecto             |
|                     | 2.3 Campos vacíos                |
| 3. Modificar subred | 3.1 Modificar satisfactoriamente |
|                     | 3.2 Datos incorrecto             |
|                     | 3.3 Campos vacíos                |

**4.4.1.3 Descripción de las variables.**

| No | Nombre de campo        | Clasificación                        | Valor Nulo | Descripción   |
|----|------------------------|--------------------------------------|------------|---|
| 1  | Subred                 | campo de texto                       | No         | Es una dirección IP que debe corresponderse con la máscara de red.  |
| 2  | Máscara de red         | campo de texto                       | No         | Es una dirección IP que define en la dirección cuanto es de red y cuanto de host, al modificarse este campo se actualiza el campo Bit con la longitud correspondiente a la máscara insertada. |
| 3  | Bit                    | Lista desplegable                    | No         | Esta lista define la longitud en bit de la máscara de red, por lo que si se modifica entonces se actualiza el campo máscara de red con la máscara correspondiente.                            |
| 4  | Rangos de asignaciones | Tabla con dos columnas (Inicio ,Fin) | Sí         | Las columnas Inicio (IP iniciales) y Fin (los IP finales,) definen los rangos de asignación, estos rangos deben de estar de acuerdo con la subred y no deben estar en conflicto entre ellos.  |

**4.4.1.4 Sección eliminar subred.**

| Escenario                             | Descripción  | Subred | Máscara | Bit | Rangos de asignaciones | Respuesta del sistema   | Resultado de la prueba | Flujo central  |
|---------------------------------------|--|--------|---------|-----|------------------------|---|------------------------|--|
| EC 1.1<br>Eliminar satisfactoriamente | Se elimina una Subred del DHCP luego de seleccionarla en el árbol. | NA     | NA      | NA  | NA                     | Emite un mensaje de confirmación.   | Satisfactorio          | El administrador selecciona la subred que desea eliminar del árbol, luego en el menú operaciones, submenú Subredes, selecciona Eliminar subred |
| EC 1.1<br>Subred no seleccionada      | Se intenta eliminar una subred del DHCP sin seleccionarla.         | NA     | NA      | NA  | NA                     | Emite un mensaje indicando que no ha seleccionado la subred a eliminar en el árbol. | Satisfactorio          | El administrador selecciona en menú operaciones, submenú Subredes, la opción Eliminar subred.  |

#### 4.4.1.5 Sección adicionar subred y modificar subred.

| Escenario                              | Descripción   | Subred          | Máscara              | Bit     | Rangos de asignaciones                             | Respuesta del sistema                           | Resultado de la prueba | Flujo central   |
|--|---|-----------------|----------------------|---------|--|---|------------------------|---|
| EC 1.1<br>Adicionar satisfactoriamente | Permite adicionar una Subred al DHCP, en caso que ya exista esta subred envía un error. | V<br>10.4.2.0   | V<br>255.255.255.0   | V<br>24 | V<br>10.4.2.15 - 10.4.2.200                        | La Subred se adicionó con éxito.                | Satisfactorio          | El administrador selecciona en el servidor donde desea adicionar la subred, luego en el menú operaciones, submenú Subredes, ejecuta la opción Nueva Subred. |
|  |   | V<br>10.4.2.0   | V<br>255.255.255.0   | V<br>24 | V<br>10.4.2.1 - 10.4.2.10                          | Ya existe la subred 10.4.2.0 en el servidor.    | Satisfactorio          |   |
| EC 1.2<br>Datos incorrectos            | Cuando se intenta adicionar una subred con los datos incorrectos.                       | I<br>10.8.4.8   | V<br>255.255.255.0   | V<br>24 | V<br>10.8.4.100 - 10.8.4.140                       | La subred no está de acuerdo con la máscara.    | Satisfactorio          | Inserta los datos como son:<br>1. Subred.<br>2. Máscara de Red.<br>3. Bit<br>4. Rangos de IP.<br><br>El administrador selecciona el botón Adicionar.        |
|  |   | V<br>10.8.4.128 | V<br>255.255.255.128 | V<br>25 | I<br>10.8.4.128 - 10.8.4.220                       | La Subred se adicionó con éxito.                | Insatisfactorio        |   |
|  |   | V<br>10.4.2.0   | V<br>255.255.255.0   | V<br>24 | I<br>10.8.4.10 - 10.8.4.2                          | El rango 10.8.4.10 - 10.8.4.2 no está correcto. | Satisfactorio          |   |
|  |   | V<br>10.8.3.0   | V<br>255.255.255.0   | V<br>24 | I<br>10.8.3.1 -10.8.3.120<br>10.8.3.10 -10.8.3.200 | Existen rangos en conflicto.                    | Satisfactorio          |   |
|  |   | V<br>10.8.3.0   | V<br>255.255.255.0   | V<br>24 | I<br>10.8.4.1 -10.8.3.120                          | El rango no está de acuerdo con la subred.      | Satisfactorio          |   |
| EC 1.3<br>Campos vacíos                | Al  | I<br>...        | V<br>255.255.0.0     | V<br>25 | V<br>10.8.3.1 -10.8.3.120                          | Existe un error en el IP de la subred.          | Satisfactorio          |   |

| Escenario                              | Descripción   | Subred          | Máscara              | Bit     | Rangos de asignaciones                             | Respuesta del sistema                           | Resultado de la prueba | Flujo central   |
|--|---|-----------------|----------------------|---------|--|---|------------------------|---|
| EC 2.1<br>Modificar satisfactoriamente | Permite modificar una Subred al DHCP, con los datos correctos.    | V<br>10.4.2.0   | V<br>255.255.255.0   | V<br>24 | V<br>10.4.2.15 - 10.4.2.200                        | La Subred se adicionó con éxito.                | Satisfactorio          | El administrador selecciona en el servidor la subred que desea modificar<br><br>Modifica los datos:<br>1. Subred. |
|  |   | V<br>10.4.2.0   | V<br>255.255.255.0   | V<br>24 | V<br>10.4.2.1 - 10.4.2.10                          | Ya existe la subred 10.4.2.0 en el servidor.    | Satisfactorio          |   |
| EC 2.2<br>Datos incorrectos            | Cuando se intenta modificar una subred con los datos incorrectos. | I<br>10.8.4.8   | V<br>255.255.255.0   | V<br>24 | V<br>10.8.4.100 - 10.8.4.140                       | La subred no está de acuerdo con la máscara.    | Satisfactorio          | El administrador selecciona el botón Modificar.   |
|  |   | V<br>10.8.4.128 | V<br>255.255.255.128 | V<br>25 | I<br>10.8.4.128 - 10.8.4.160                       | La Subred se adicionó con éxito.                | Insatisfactorio        |   |
|  |   | V<br>10.4.2.0   | V<br>255.255.255.0   | V<br>24 | I<br>10.8.4.10 - 10.8.4.2                          | El rango 10.8.4.10 - 10.8.4.2 no está correcto. | Satisfactorio          |   |
|  |   | V<br>10.8.3.0   | V<br>255.255.255.0   | V<br>24 | I<br>10.8.3.1 -10.8.3.120<br>10.8.3.10 -10.8.3.200 | Existen rangos en conflicto.                    | Satisfactorio          |   |
|  |   | V<br>10.8.3.0   | V<br>255.255.255.0   | V<br>24 | I<br>10.8.4.1 -10.8.3.120                          | El rango no está de acuerdo con la subred.      | Satisfactorio          |   |
|  |   | I<br>...        | V<br>255.255.0.0     | V<br>25 | V<br>10.8.3.1 -10.8.3.120                          | Existe un error en el IP de la subred.          | Satisfactorio          |   |



**4.4.1.6 Registro de defectos y dificultades detectados.**

| Elemento               | N<br>o | Ubicación de la No Conformidad    | Etapas de Detección | Significativa         | Estado NC                                   | Respuesta del Equipo de Desarrollo |
|------------------------|--------|-----------------------------------|---------------------|-----------------------|---|------------------------------------|
| Rangos de asignaciones | 1      | Gestionar Subred/Añadir subred    | F. Función          | Aplicación/Validación | -29/5/2011 pendiente<br>-30/5/2011 resuelta | -                                  |
| Rangos de asignaciones | 2      | Gestionar Subred/Modificar Subred | F. Función          | Aplicación/Validación | -29/5/2011 pendiente<br>-30/5/2011 resuelta | -                                  |

**4.5 Conclusiones**

En el presente capítulo se describieron los elementos correspondientes a los flujos de trabajo de implementación y prueba, donde en la implementación se desarrolló el sistema en términos de componentes, obteniendo los diagramas de componentes para cada caso de uso y el diagrama de despliegue. Además se trazó la estrategia de prueba a seguir para detectar errores en las funciones de la aplicación. De esta manera se completan los artefactos exigidos para la realización del presente trabajo de diploma.

## CONCLUSIONES

En la elaboración del presente trabajo se realizó una investigación sobre las tecnologías existentes que gestionan el servicio DHCP, de cada una se tomaron las mejores prácticas para lograr una aplicación con un alto nivel. Además se llevó a cabo una selección de las herramientas, metodologías y lenguajes que se utilizaron en la construcción del Módulo DHCP y sus funcionalidades, generando una serie de artefactos, seguido por la implementación del sistema y terminando con las pruebas realizadas al mismo, dando como resultado una aplicación capaz de gestionar las declaraciones del servidor DHCP en GNU/Linux, lo que puede ser de manera local o remota, además le permite al administrador de red tener control sobre las estadísticas de uso del servidor y de las asignaciones realizadas por este. Se logró mejorar el proceso de migración, incorporando una funcionalidad para importar los datos que se extraen previamente del servidor DHCP de Windows Server 2003 o de una exportación realizada desde el propio sistema; dados los resultados y artefactos obtenidos y las funcionalidades con que cuenta el Módulo se considera que se cumplió con el objetivo planteado en el presente trabajo. Se espera que éste sea de gran utilidad en el proceso de migración en que se encuentra enfrascada Cuba hacia el uso del software libre.

## RECOMENDACIONES

Se recomienda:

Desarrollar funcionalidades para la gestión de los failover (Conmutación por error) y las pool (piscinas) existentes en el DHCP de GNU/Linux

Realizar un estudio sobre gestión de las declaraciones básicas existente en el DHCP de GNU/Linux para IP en su versión 6 e incorporarle las funcionalidades respectivas la aplicación desarrollada.

Extender la administración a interfaces web y de consolas, reutilizando las funcionalidades, y dándole al administrador más de una forma de realizar su labor de acuerdo a sus preferencias o condiciones específicas del entorno en que se encuentre.

## BIBLIOGRAFÍA

1. GNU/Linux Operating System. [En línea] [Citado el: 08 de Diciembre de 2010.] <http://www.gnu.org/>.
2. Definición DHCP. *sitio Web Microsoft Technet*. [En línea] [Citado el: 9 de Febrero de 2011.] <http://technet.microsoft.com/es-es/library/cc780906%28WS.10%29.aspx>.
3. [www.buenastareas.com](http://www.buenastareas.com). [En línea] [Citado el: 10 de Diciembre de 2010.] <http://www.buenastareas.com/ensayos/Metodo-Empirico-Analitico/278643.html>.
4. **Hernández Meléndrez, Edelsys**. [www.sld.cu](http://www.sld.cu). [En línea] 2006. [Citado el: 12 de Diciembre de 2010.] <http://www.sld.cu/galerias/pdf/sitios/hematologia/tesis.pdf.pdf>.
5. **Gualdarrama Reyes, Jorge Luis**. *Plataforma de Gestión de Servicios Telemáticos en GNU/Linux. Módulo DNS v1.0*. Ciudad de la Habana : s.n., 2009.
6. Windows Server 2003. [En línea] [Citado el: 08 de Diciembre de 2010.] <http://www.microsoft.com/spain/windowsserver2003/download/sp1/overview.mspx>.
7. **Padilla Moya, Álvaro Luis**. *Plataforma de Gestión de Servicios Telemáticos en GNU/Linux. Módulo DNS v2.0*. Ciudad de la Habana : s.n., 2010.
8. [www.uml.org](http://www.uml.org). [En línea] [Citado el: 08 de Enero de 2011.] <http://www.uml.org/>.
9. **James Rumbaugh, Ivar Jacobson, Grady Booch**. El Lenguaje Unificado de Modelado. Manual de Referencia. [En línea] [Citado el: 14 de Diciembre de 2011.] <http://bibliodoc.uci.cu/pdf/reg03050.pdf>.
10. **James Rumbaugh, Ivar Jacobson, Grady Booc**. El Proceso Unificado de Desarrollo de Software. [En línea] [Citado el: 05 de Enero de 2011.] <http://bibliodoc.uci.cu/pdf/reg00060.pdf>.
11. IDEF0. *IDEF0*. [En línea] Knowledge Based Systems, Inc. [Citado el: 25 de Enero de 2011.] <http://www.idef.com/IDEF0.htm>.
12. **Guarda, Alfonso de la**. [dev.laptop.org](http://dev.laptop.org). [En línea] [Citado el: 08 de Enero de 2011.] [http://dev.laptop.org/~edsiper/byteofpython\\_spanish/ch01s02.html](http://dev.laptop.org/~edsiper/byteofpython_spanish/ch01s02.html).

13. di-mare.com. [En línea] [Citado el: 07 de Enero de 2011.] <http://www.di-mare.com/adolfo/cursos/2007-1/pp-GenGUI.pdf>.
14. easyeclipse.org. [En línea] [Citado el: 08 de Enero de 2011.] <http://www.easyeclipse.org/site/home/>.
15. visual-paradigm.com. [En línea] [Citado el: 07 de Enero de 2011.] <http://www.visual-paradigm.com/product/vpuml/>.
16. **Duque, Raúl González.** *Python para todos*.
17. **Pressman, Roger S.** *Ingeniería de software: Un enfoque práctico*. s.l. : Editorial McGraw-Hil, 2002.
18. **Cabrera, Sandy Noa.** *Plataforma de Gestión de Servicios Telemáticos en GNU/Linux. Módulo de Directorio*. Ciudad de la Habana : s.n., 2009.
19. Internet Systems Consortium. [En línea] [Citado el: 08 de Enero de 2011.] <http://www.isc.org/software/dhcp>.
20. SUSE Linux Enterprise Server. *www.novell.com*. [En línea] [Citado el: 08 de Enero de 2011.] <http://www.novell.com/es-es/solutions/enterprise-linux-servers/>.
21. *www.python.org*. [En línea] [Citado el: 08 de Noviembre de 2010.] <http://www.python.org/doc/>.
22. **Brittain, Jason.** *Python for Unix and Linux System Administration*. s.l. : O'Reilly Media, Inc., 2008. 978-0-596-51582-9.
23. Definición DHCP. *sitio Web Microsoft Technet*. [En línea] 22 de Mayo de 2011. <http://technet.microsoft.com/es-es/library/cc780906%28WS.10%29.aspx>.
24. elai.upm.es. [En línea] [Citado el: 27 de Enero de 2010.] <http://www.elai.upm.es/spain/Investiga/GCII/personal/vcorte/informeqt.PDF>.

## GLOSARIO

### A

Apache: Es un servidor web HTTP de código abierto para plataformas Unix (BSD, GNU/Linux, etc.), Microsoft Windows, Macintosh y otras, que implementa el protocolo HTTP.

### B

bytecodes: Código intermedio generalmente tratado como un fichero binario que contiene un programa ejecutable similar a un módulo objeto. Algunos sistemas, llamados traductores dinámicos traducen el bytecode a código máquina inmediatamente antes de su ejecución para mejorar la velocidad de ejecución.

### D

DHCP: Protocolo de configuración de host dinámico es un protocolo que permite configurar dinámicamente los parámetros de red necesarios para que un sistema pueda comunicarse.

DNS: Domain Name System es una base de datos distribuida, con información que se usa para traducir los nombres de dominio, fáciles de recordar y usar por las personas, en números de IP que es la forma en la que las máquinas pueden encontrarse en Internet.

### G

GPL: GNU General Public License, licencia creada por la Free Software Foundation a mediados de los años 80, orientada principalmente a proteger la libre distribución, modificación y uso del software. Su propósito es declarar que el software cubierto por esta licencia es Software Libre.

GTK: Biblioteca constituida por objetos y funciones que permiten crear interfaces gráficas de usuario. Maneja widgets como ventanas, botones, menús, etiquetas, deslizadores, pestañas, entre otros.

### O

Open Source: Término con el que se conoce al software distribuido y desarrollado libremente. Sin embargo, mucho de este software no da a sus usuarios la libertad de distribuir sus modificaciones, restringe el uso comercial, o en general restringe los derechos de los usuarios.

## P

**Perl:** Practical Extraction and Report Language es un **lenguaje de programación** para la administración de tareas propias de sistemas UNIX.

**Plugin:** Aplicación que se relaciona con otra para aportarle una función nueva y generalmente muy específica.

## Q

**Qt:** Biblioteca multiplataforma para desarrollar interfaces gráficas de usuario desarrollada por la compañía noruega Trolltech.

## S

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

**SSL:** Secure Sockets Layer o Protocolo de Capa de Conexión Segura, proporciona autenticación y privacidad de la información mediante comunicaciones seguras por una red, comúnmente Internet con el uso de criptografía.

## T

**TCP/IP:** Protocolo de Control de Transmisión (TCP) y Protocolo de Internet (IP) representan los dos protocolos más importantes que componen Internet. TCP/IP es la base de Internet, y sirve para enlazar computadoras que utilizan diferentes sistemas operativos.

## W

**Widgets:** es un componente gráfico, o control, con el cual el usuario interactúa, como por ejemplo, una ventana, una barra de tareas o una caja de texto.