

**Universidad de Cienfuegos “Carlos Rafael Rodríguez”
Facultad de Informática
Carrera de Ingeniería Informática**



SECL

Sistema de Envío de Correos por Lote

**Trabajo de diploma para optar por el título de
Ingeniería en informática**

Autor

Curtis Vidal Edwards.

Tutores

Ing. Alexis Gómez Domínguez.

Ing. Daimarelys Acevedo Cardoso.

Consultante

Dr. Roberto Fuentes Garí.

**Cienfuegos, Cuba
Curso 2005 – 2006**

Declaración de autoridad

Yo, Curtis Vidal Edwards, declaro que soy el único autor de este trabajo y autorizo al Departamento de Ingeniería Informática de la Universidad de Cienfuegos para que hagan el uso que estimen pertinente con este trabajo.

Para que así conste firmo la presente a los 15 días del mes de Junio del 2006.

Autor: Curtis Vidal Edwards

Tutor: Alexis Gómez Domínguez

Tutor: Daimarelys Acevedo Cardoso

Los abajo firmantes certificamos que el presente trabajo ha sido revisado según acuerdo de la dirección de nuestro centro y el mismo cumple los requisitos que debe tener un trabajo de esta envergadura referente a la temática señalada.

Firma Tutor

Firma ICT

Firma Tutor

Firma Vicedecano

*A mi familia, en especial a mi
mamá Antionette Charles Edwards
y hermanita Victoria Antoinette Edwards*

Agradecimientos.

Quisiera agradecer a todas las personas que de una forma u otra me han apoyado en la realización de este trabajo y en mi carrera, en especial:

- *A Dios por darme la fuerza y las habilidades necesarias para lograr lo mismo.*
- *A mi gobierno que me dio la oportunidad de estudiar afuera.*
- *Al Gobierno y pueblo Cubano por ofrecernos la posibilidad de estudiar en su país, y ayudarnos desde que llegamos.*
- *A mis tutores Ing. Alexis Gómez Domínguez e Ing. Daimarelys Acevedo Cardoso y mi consultante el Dr. Roberto Fuentes Gari, que tuvieron la paciencia y dedicación necesarias para que este trabajo fuera posible.*
- *Mis profesores de la Universidad de Cienfuegos que me transmitieron sus conocimientos y valores que ayudaron a formarme como profesional.*
- *A mis compañeros de grupo por soportarme todos estos años.*
- *A mi familia, especialmente a mi mamá quien fue mi primera maestra de la matemática y mi hermana por su ayuda en todos estos años.*
- *A mi novia Sasha S. Robin de todos estos años por su infinita paciencia y amor.*
- *A Lachy, la mamá de todos los extranjeros en la escuela.*
- *A mis mejores amigos Cubanos Yosvany Álvarez Bonilla y Julio Beltrán.*
- *A mi hermano Cubano Frank y su bella esposa Maryluz.*
- *Mis hermanos y hermanas aquí en Cuba Ursula Verdora Willock, Charles Hudson, Phillip Baker y Raphie-Ann Morgan.*
- *A mis compañeros de cuarto, Bigjoe, Pimpo, Ervin(salop), Indian y Andy*

A todos y por todo, Muchísimas Gracias.

*Hay una fuerza motriz más poderosa que el vapor,
la electricidad y la energía atómica: la voluntad.*

Albert Einstein

Resumen

Los sistemas de envío de correo electrónico a grupos de usuarios son ampliamente empleados en las redes telemáticas, su utilización resuelve eficientemente las necesidades de difusión de información de forma simultánea a grupos de usuarios. En la universidad de Cienfuegos este servicio se ha convertido en una herramienta de trabajo tanto para directivos como para usuarios finales, sin embargo su forma de implementación no cumple con las expectativas de los usuarios del servicio, viéndose afectado el mismo por las deficiencias asociadas al método tradicional basado en el empleo de alias de correo.

El presente trabajo investigativo titulado, "Sistema de Envío de Correos por Lote (SECL)" tiene como objetivo desarrollar un sistema que permita la administración, control y envío de correos a grupos de usuarios de manera eficiente y cumpliendo con los requerimientos deseados por la dirección del centro y por los usuarios finales.

Para ello, se efectuó una revisión de la bibliografía técnico-especializada y del estado del arte en la materia, a fin de determinar las tecnologías y métodos a emplear para desarrollar e implementar el sistema. Además se utilizó el lenguaje de modelado UML para el análisis, diseño e implementación de la solución propuesta, siguiendo lo establecido por el Proceso Unificado de Desarrollo de Software (RUP).

El trabajo culmina con las conclusiones y recomendaciones derivadas del cumplimiento de los objetivos trazados al inicio del trabajo y la implementación práctica del sistema desarrollado.

Índice

Introducción.....	1
Capítulo 1. Fundamentos teóricos.....	5
1.1 Introducción.	5
1.2 Principales conceptos asociados al dominio del problema.	5
1.2.1 Sistemas de correo electrónico.	5
1.2.2 Aplicaciones servidores de correo electrónico.	7
1.2.3 Aplicaciones clientes de correo electrónico.....	8
1.2.4 Protocolos utilizados en los sistemas de correo electrónicos.....	9
1.2.5 Alias de correo electrónico.	11
1.2.6 Listas de correos electrónicos.	12
1.3 Descripción del objeto de estudio.	14
1.3.1 Descripción general.....	14
1.3.2 Situación problemática.	15
1.3.3 Problema.	16
1.4 Sistemas existentes vinculados al campo de acción.	16
1.5 Lenguaje de Modelación Unificado (UML).	17
1.6 Proceso Unificado de Desarrollo (RUP).....	18
1.7 Tecnologías a utilizar.	19
1.7.1 Lenguajes.....	19
1.7.1.1 Hypertext Markup Language (HTML).	19
1.7.1.2 Professional Home Page Tools (PHP).....	20
1.7.1.3 Active Server Page (ASP).	22
1.7.1.4 ¿Por qué PHP?.....	23
1.7.1.5 Java Script.	23
1.6.3 Sistemas Gestores de Base de Datos (SGDB).	24
1.7.3.1 SQL (Structure Query Language).	25
1.7.3.2 MySQL.....	26
1.7.3.3 SQL Server.	26
1.7.3.4 ¿Por qué MySQL?	27
1.8 Conclusiones.	28

Capítulo 2. Descripción de la solución propuesta.	29
2.1 Introducción.	29
2.2 Descripción del Modelo del Dominio.....	29
2.2.1 Principales conceptos asociados al dominio del problema.	29
2.2.2 Representación del Modelo del Dominio.....	30
2.2.3 Reglas del negocio a considerar.	30
2.3 Descripción del sistema propuesto.	31
2.3.1 Concepción general.	31
2.3.2 Requerimientos funcionales.	32
2.3.3 Requerimientos no funcionales.	33
2.4 Modelo de Casos de Uso del Sistema.....	35
2.4.1 Actores del sistema.	35
2.4.2 Casos de Uso del sistema.....	37
2.4.3 Diagrama de Casos de Uso del sistema.	38
2.4.3 Descripción de los Casos de Uso del sistema.	39
2.4 Conclusiones.	49
Capítulo 3. Construcción de la solución propuesta.....	50
3.1 Introducción.	50
3.2 Diagrama de clases del diseño.....	50
3.3 Principios de diseño.....	60
3.3.1 Estándares en la interfaz de la aplicación.	60
3.3.2 Concepción general de la ayuda.....	61
3.3.3 Tratamiento de errores.....	61
3.4 Diseño de la base de datos.	61
3.4.1 Modelo lógico de datos.	62
3.4.2 Modelo físico de datos.	63
3.5 Diagrama de implementación.	64
3.6 Conclusiones.	65
Conclusiones.....	66
Recomendaciones.....	68
Referencias Bibliográficas.	69

Bibliografía.....	73
Glosario de Términos y Abreviaturas.....	75
Anexos	80
Anexo 1. Sistema de correo electrónico que soporta alias de correo y listas de expansión.	80
Anexo 2. Prototipos.	81

Introducción

Los sistemas de correo electrónico se han convertido en servicio indispensable de cualquier red telemática, su utilización resuelve innumerables problemas de intercambio de información y difusión de la misma. Hoy en día sus direcciones se han hecho comunes dentro de los datos personales de cualquier persona o entidad, permitiendo contar con un medio de comunicación fuera de línea rápido y eficiente. Estas características unidas al alcance de este servicio, su facilidad de utilización e implementación y su comúnmente, alto nivel de disponibilidad lo han hecho convertirse en herramienta de trabajo de muchas instituciones.

El correo electrónico se usa ampliamente hoy día en la industria para la comunicación intra-compañía. Permite que los empleados remotos cooperen en proyectos complejos, incluso estando en otros usos horarios. Algunas compañías han estimado que el correo electrónico ha mejorado su productividad hasta en 30% [1].

Una de las ventajas significativas de este sistema, es la posibilidad de enviar informaciones simultáneamente a grupos de usuarios, ofreciendo una vía muy efectiva para su difusión.

El envío de correos electrónicos a grupos de usuarios es indiscutiblemente una necesidad en las redes corporativas, redes de campus universitarios o intranets de cualquier organización, convirtiéndose en un servicio ampliamente solicitado y utilizado por los usuarios de centros con una infraestructura telemática mediana o grande. La universidad de Cienfuegos, es un ejemplo de ello, la utilización de este servicio producto de sus ventajas asociadas a la rapidez en la difusión de informaciones, hace que la demanda de utilización del mismo crezca en la misma medida que se expande la red telemática en la universidad. Su empleo a nivel de universidad, facultades, SUMs, áreas, departamentos e incluso grupos, brigadas y organizaciones, es una práctica común y necesaria en las actividades diarias del centro.

Sin embargo, los sistemas tradicionales utilizados para esto, basados generalmente en el empleo de alias de correo, no satisfacen completamente las necesidades de los usuarios de la red de la universidad de Cienfuegos, por el contrario le imponen deficiencias que afectan la fiabilidad, seguridad y estabilidad del sistema de correos empleado en la misma. Los problemas colaterales que genera el modelo actual, repercuten directamente en los niveles de seguridad informática deseados y exigidos en el Plan de Seguridad Informática aprobado en la universidad, además provocan molestias a los usuarios y administradores del servicio, resultando ser altamente sensible a los errores humanos y carece de un mecanismo de supervisión y control que permita evitar su uso indiscriminado.

El principal problema encontrado es precisamente la carencia de un sistema que permita ofrecer este servicio sin las deficiencias antes mencionadas y con niveles aceptables de seguridad, control y prestaciones. Para la solución de este problema surgen las siguientes interrogantes:

- ¿Cuáles son las diferentes variantes que pueden utilizarse para el envío de correo electrónico por lote, que permitan sustituir el sistema basado en alias?
- ¿Cómo desarrollar e implementar una variante eficiente para el envío de correos por lote en la universidad de Cienfuegos?

Debido a ello se decidió desarrollar una aplicación que solucionara y eliminara las deficiencias del modelo actual, capaz de garantizar la continuidad del servicio, adecuándolo a las necesidades y exigencias de la red telemática de la universidad de Cienfuegos, objeto del presente trabajo.

Objetivo general:

Analizar, diseñar e implementar un sistema que permita controlar y enviar correos electrónicos a grupos de usuarios en la universidad de Cienfuegos, evitando el empleo de alias de correo en su funcionamiento.

Objetivos específicos:

- Determinar los antecedentes y estado actual de los sistemas empleados para el envío de correos por lote.
- Analizar los principales problemas de los sistemas empleados para el envío de correos por lote basados en el empleo de alias.
- Realizar un estudio de las tendencias y tecnologías actuales del campo de la Informática, determinando cuales utilizar en la solución del problema.
- Realizar el análisis y diseño del sistema.
- Desarrollar e implementar un sistema para el envío de correos por lote en la red telemática de la universidad de Cienfuegos.

Para alcanzar los objetivos propuestos se desarrollarán las siguientes tareas:

- Revisión de la bibliografía técnico-especializada para la construcción del marco teórico que nos permitirá conocer el estado del arte referente a los sistemas de envío de correos por lote, principio de funcionamiento, ventajas y desventajas asociadas.
- Selección de las herramientas y lenguajes de programación más efectivos que cumplan con las exigencias planteadas.
- Analizar, diseñar y crear la base de datos necesaria para el sistema.
- Analizar, diseñar y crear la interfaz gráfica de la aplicación.
- Poner a prueba la aplicación para comprobar de forma práctica su correcto funcionamiento y el cumplimiento de los objetivos trazados.

El presente documento está estructurado en 3 capítulos, donde se presenta la siguiente información:

Capítulo 1: Fundamentos teóricos.

Se determinan los antecedentes y el estado actual de los sistemas de envío de correos electrónicos a grupos de usuarios, exponiendo los aspectos teóricos y

conceptos asociados con el tema a desarrollar y se describen las metodologías y tecnologías utilizadas.

Capítulo 2: Descripción de la solución propuesta.

Se describe la solución propuesta utilizando la Metodología RUP, empleando para ello el Modelo del Dominio, los Requerimientos Funcionales y No Funcionales, el Diagrama de Casos de Uso y la descripción de cada uno, así como una descripción general del sistema que se propone.

Capítulo 3: Construcción de la solución propuesta.

Se realiza una descripción de la construcción de la solución propuesta, para ello se han utilizado los diagramas de clases Web del diseño y el modelo lógico y físico de la base de datos. Además, se describen los principios de diseño utilizados, mostrando ejemplos de cómo son presentados al usuario y la concepción general de la ayuda.

Capítulo 1. Fundamentos teóricos.

1.1 Introducción.

Varios de los primeros encuentros de los usuarios con las redes de computadora se dan cuando envían o reciben correo electrónico (e-mail) desde o hacia una localidad remota siendo este el servicio de aplicación utilizado más ampliamente [2] en la red de redes¹. Este servicio presenta una gran cantidad de ventajas asociadas, destacándose entre ellas la posibilidad de enviar mensajes a grupos de usuarios simultáneamente. Este capítulo tiene como objetivo delinear los antecedentes y el estado actual de los sistemas de envío de correos electrónicos a grupos de usuarios, exponer los aspectos teóricos y conceptos asociados con el tema a desarrollar y describir las metodologías y tecnologías utilizadas.

1.2 Principales conceptos asociados al dominio del problema.

En realidad, muchos de los usuarios de computadoras acceden a las redes solo a través del correo electrónico [2]. Podemos decir que su popularidad radica en ofrecer un método rápido y conveniente de transferencia de información, adaptándose al envío de pequeñas notas o grandes y voluminosos documentos mediante un mecanismo sencillo que puede utilizarse tanto para realizar envíos a usuarios independientes o a pequeños, medianos y grandes grupos de usuarios de forma simultánea. Incluso, según estudios realizados, hay más usuarios que envían archivos por correo electrónico que por los protocolos de transferencia de archivos [2].

1.2.1 Sistemas de correo electrónico.

El correo electrónico o mensajería es un servicio que se utiliza para las comunicaciones de mensajes con ámbito interno, local, nacional o internacional soportado por cualquier red de comunicación de datos. Este servicio proporciona el envío electrónico de mensajes a una velocidad muy alta y a un coste en ocasiones

¹ La terminología “red de redes” es ampliamente utilizada para referirse a Internet por ser esta una red que interconecta la mayoría de las redes telemáticas mundiales.

inferior al del correo normal, por lo que cada día es más utilizado por las empresas. Un sistema de correo electrónico consta de cuatro elementos bien definidos [3].

Estos son los siguientes:

- Los Terminales emisor y receptor. Pueden ser computadoras personales o terminales de una computadora multipuesto.
- El Agente de Usuario (User Agent, UA). Es el encargado de la edición y recepción del mensaje, así como del control de prioridades, notificaciones de la recepción y seguridad de las operaciones realizadas en el terminal.
- El Agente de Transferencia de Mensajes (Message Transfer Agent, MTA). Es el encargado de proporcionar el encaminamiento de los mensajes a través de una ruta determinada.
- El Archivo o Almacén de Mensajes (Message Store, MS). Es el encargado del almacenamiento de los mensajes recibidos para facilitar su manipulación.

Además, un sistema de correo electrónico debe proporcionar los medios cuando la máquina remota o las conexiones de la red han fallado. El emisor no desea esperar a que la máquina remota esté disponible para continuar trabajado, ni el usuario quiere que se aborte la transmisión sólo porque las comunicaciones con la máquina remota no están disponibles temporalmente. Para manejar las entregas con retraso, el sistema de correo utiliza una técnica conocida como spooling². Cuando el usuario envía un mensaje de correo, el sistema coloca una copia en su área de almacenamiento privado (spool³) junto con la identificación del emisor, recipiente, máquina de destino y hora de depósito. El sistema, entonces, inicia la transferencia hacia la máquina remota como una actividad subordinada o secundaria, permitiendo al emisor que continúe con otras actividades computacionales.

² Se trata de una técnica de procesamiento simultáneo de tareas periféricas.

³ El área spool de correo se conoce a veces como “cola de correo”, aun cuando el término es técnicamente impreciso.

Para la implementación de los sistemas de correo electrónico se necesitan de aplicaciones servidores y aplicaciones clientes que desarrollen y cumplan con los requisitos y las funciones expuestas anteriormente.

1.2.2 Aplicaciones servidores de correo electrónico.

Existe una gran variedad de aplicaciones servidores de correo electrónico, tanto de sistemas propietarios como de código libre y para diferentes sistemas operativos. En general todos cumplen con los estándares internacionales que definen la arquitectura de los sistemas de correo electrónicos⁴ [4], permitiendo esto la compatibilidad de los sistemas mediante el uso de protocolos estandarizados, haciendo posible el intercambio de mensajes entre tecnologías diferentes.

Entre los más populares podemos mencionar los siguientes:

1. Microsoft Exchange 2003®, hoy en día es el servidor de correo que propone la compañía Microsoft y cuenta con gran popularidad entre los usuarios y administradores de sistemas de correo electrónico, es un sistema propietario que hay que pagar y se adquiere bajo licencia de Microsoft Corporation®. Se instala sobre sistemas operativos de la familia de servidores Windows 2003 Server® [5].
2. Mdaemon, es un sistema de amplio uso, sobre todo por su sencillez y fácil administración. Este sistema se instala sobre sistemas operativos de las familias de servidores Windows 2000 o Windows 2003, integrándose con el sistema de directorio activo de Microsoft (Active Directory®) [6].
3. Sendmail, es uno de los sistemas más famosos dentro de la comunidad *Open Source*⁵, diseñado para plataformas Unix y Linux. Es un potente sistema que aunque más complejo en su configuración y administración, ofrece elevadas prestaciones, fiabilidad y estabilidad a sus usuarios [7].

⁴ Las normalizaciones internacionales de correo electrónico están basadas en la serie de recomendaciones X.400 del ITU, equivalentes a los estándares llamados MOTIS de la ISO.

⁵ Open Source, se refiere a los sistemas de código libre.

4. Postfix, es una aplicación servidor de correo electrónico relativamente joven, pero desde su aparición ha ido ganando popularidad debido fundamentalmente a su simplicidad en la configuración y administración. Es una aplicación *Open Source* diseñada para ambientes Unix y Linux que elimina las complejidades de Sendmail y hereda sus niveles de prestaciones, fiabilidad y estabilidad [8].

1.2.3 Aplicaciones clientes de correo electrónico.

Si bien son importantes las aplicaciones servidores de correo, no menos lo son las aplicaciones clientes, estas proveen una interfase al usuario final que debe cumplir con las expectativas del mismo y ofrecer las funcionalidades necesarias que permitan un uso eficiente del sistema. Al igual que los servidores existe gran variedad de sistemas clientes, incluso más, debido a que muchos o la mayoría son independientes de las aplicaciones servidores gracias a la utilización de protocolos estandarizados, permitiéndoles a los desarrolladores de software facilidades e independencia en sus propuestas.

Entre los más utilizados podemos encontrar los siguientes:

1. Microsoft Outlook Express® de Microsoft Corporation®, es una aplicación para sistemas operativos Microsoft Windows®, disponible en diferentes versiones para las distintas distribuciones de ese sistema operativo, se destaca por su interfase amigable y facilidades de uso, permitiendo interactuar con aplicaciones servidores diferentes, que lo hacen uno de los sistemas clientes más populares [9].
2. Mozilla Thunderbird® de Mozilla Foundation®, esta aplicación *Open Source* está disponible para sistemas operativos Windows y Linux y al igual que otras presenta una interfase amigable y facilidades de uso, también es independiente del sistema servidor y desde su aparición ha ido despertando el interés de los usuarios que lo ven como una alternativa al hasta ahora casi dominante Outlook Express [10].
3. SquirrelMail es un cliente de correo con interfaz Web (*webmail*) desarrollado en el lenguaje de programación PHP que incluye soporte completo para los

protocolos IMAP y SMTP, generando todas las páginas Web en HTML, no requiriendo Javascripts para lograr la máxima compatibilidad en todos los navegadores. Cuenta con sencillos requerimientos de instalación y es fácilmente configurable. SquirrelMail posee todas las funcionalidades que pueda requerir de un cliente correo, incluyendo soporte MIME, libro de direcciones y manipulación de carpetas [11].

4. Open Web Mail, es una aplicación Web cliente de correo que está desarrollada en lenguaje de programación Perl, es *Open Source* y ofrece facilidades similares a los clientes no Web, tales como Outlook Express. Su interfase muy amigable lo hace ubicarse entre los preferidos de muchos usuarios, sin embargo es preciso detallar bien su instalación y configuración debido a que es más vulnerable que otros a problemas de seguridad [12].

1.2.4 Protocolos utilizados en los sistemas de correo electrónicos.

Tanto los sistemas servidores como los sistemas clientes se basan en la utilización de diferentes protocolos estandarizados en la arquitectura TCP/IP que permiten la compatibilidad entre sistemas diferentes que se ejecuten en plataformas de *software* y *hardware* diferentes.

- SMTP (Simple Mail Transfer Protocol), protocolo simple de transferencia de correos. El SMTP es un protocolo ASCII sencillo que opera sobre el protocolo de transporte TCP utilizando el puerto 25 en el cual espera las solicitudes de conexión de las máquinas clientes que desean interactuar con el servidor para entregarle correos [13]. Está definido en los RFC 2821 [14] y la estructura de los mensajes en el 2822 y 1893 [15].
- MIME (Multipurpose Internet Mail Extensions), extensiones multipropósito de correo de Internet. Está definido en el RFC 1521 [16]. La idea básica de MIME es continuar usando el formato RFC 822, pero agregar una estructura al cuerpo del mensaje y definir reglas de codificación para los mensajes no ASCII [1].
- POP3 (Post Office Protocol – version 3), protocolo de oficina postal (versión 3). Este protocolo está definido en el RFC 1939 [17]. Es un protocolo sencillo

para traer correo electrónico de un buzón en un servidor remoto que escucha en el puerto 110, su objetivo principal es obtener los correos y almacenarlos en la máquina local del usuario para su lectura posterior [1].

- IMAP (Internet Messages Access Protocol), protocolo interactivo de acceso a correo definido en la RFC 2060 [18]. Al igual que el POP3 es un protocolo utilizado por los clientes de correo para descargar los mensajes desde un servidor remoto que escucha en el puerto 143. La idea en que se basa IMAP es que el servidor mantenga un depósito central al que pueda accederse desde cualquier máquina [1].
- HTTP (HyperText Transfer Protocol), protocolo de transferencia de hipertexto. Este protocolo permite el intercambio de información entre clientes y servidores para la solicitud y envío de páginas Web, normalmente utiliza el puerto 80. Está dividido en dos elementos bastante diferentes, el grupo de solicitudes de los visualizadores a los servidores y el grupo de respuestas en el otro sentido [1].
- SSL (Secure Socket Layer). El protocolo SSL es un sistema diseñado y propuesto por Netscape Communications Corporation. Proporciona servicios de seguridad cifrando los datos intercambiados entre el servidor y el cliente con un algoritmo de cifrado simétrico, típicamente el RC4 o IDEA, y cifrando la clave de sesión de RC4 o IDEA mediante un algoritmo de cifrado de clave pública, típicamente el RSA. La clave de sesión es la que se utiliza para cifrar los datos que vienen de él y van al servidor seguro. Se genera una clave de sesión distinta para cada transacción, lo cual permite que aunque sea rota por un atacante en una transacción dada, no sirva para descifrar futuras transacciones. MD5 es utilizado como algoritmo de hash [19].
- POP3S, se refiere a la utilización del protocolo POP3 en combinación con el protocolo SSL, cuando se utiliza POP3S el servidor pasa a escuchar las solicitudes por el puerto 995 [20].

- IMAPS, se refiere a la utilización del protocolo IMAP en combinación con el protocolo SSL, cuando se utiliza IMAPS el servidor pasa a escuchar las solicitudes por el puerto 993 [21].
- HTTPS, se refiere a la utilización del protocolo HTTP en combinación con el protocolo SSL, cuando se utiliza HTTPS el servidor pasa a escuchar las solicitudes por el puerto 443 [22].

1.2.5 Alias de correo electrónico.

La mayor parte de los sistemas de correo electrónico proporcionan un *software* de envío de correo que incluye un mecanismo de expansión de alias de correo. Un emisor de correo permite a una localidad de zona transformar identificadores por medio de una dirección de correo para conjuntos de una o más direcciones de correo. Por lo regular, luego de que el usuario compone un mensaje y nombra un recipiente, el programa de interfaz de correo consulta los alias locales para remplazar el recipiente con la versión transformada antes de pasar el mensaje al sistema de entrega. Los recipientes para los que no se han especificado transformaciones se mantienen sin cambios. De la misma forma, el sistema de correo subyacente utiliza los alias de correo para transformar direcciones entrantes de recipientes [2].

Los alias incrementan sustancialmente la funcionalidad del sistema de correo. En términos matemáticos, las transformaciones de alias pueden ser de muchos a uno o de uno a muchos. Por ejemplo, el sistema de alias permite a un solo usuario tener varios identificadores de correo, incluyendo sobrenombres y posiciones, transformando un conjunto de identificadores hacia una sola persona. El sistema también permite que una localidad asocie grupos de recipientes con un solo identificador [2].

El uso de alias que transforman un identificador en una lista de identificadores hace posible establecer un distribuidor que acepte un mensaje entrante y lo envíe a un amplio conjunto de recipientes. El conjunto de recipientes asociados con un identificador se conoce como lista de correo electrónico [2].

No todos los recipientes en una lista necesitan estar en el mismo local. Aun cuando es poco común, es posible tener una lista en una localidad, Q , sin ninguno de los

recipientes de la lista localizados en Q . Extender un alias de correo en un conjunto amplio de recipientes es una técnica popular y muy utilizada [2]. El Anexo 1 ilustra los componentes de un sistema de correo que soporta alias de correo y listas de expansión.

La expansión de alias de correo puede ser peligrosa. Supongamos que dos localidades establecen alias conflictivos. Por ejemplo, supongamos que la localidad **A** transforma la dirección de correo x en la dirección de correo y en la localidad **B**, mientras que la localidad **B** transforma la dirección de correo y en la dirección x en la localidad **A**. Un mensaje de correo enviado hacia la dirección x en la localidad **A** puede rebotar indefinidamente entre las dos localidades. De la misma forma, si el administrador en la localidad **A** transforma accidentalmente el nombre de identificación en el sistema de un usuario en dicha localidad, en una dirección de otra localidad, el usuario será incapaz de recibir correo. El correo puede ir hacia el otro usuario o, si el alias especifica una dirección ilegal, el emisor recibirá mensajes de error [2].

1.2.6 Listas de correos electrónicos.

Una lista de distribución de correo es una dirección de correo a la que se pueden enviar mensajes a través del correo electrónico, y estos correos son recibidos por todas las personas que están suscritas a la misma lista. De la misma manera llegan a cada una de las cuentas de correo electrónico todos los mensajes que envíen los integrantes de la lista. Podemos decir que las listas de correo son un uso especial del correo electrónico que permite la distribución masiva de información entre múltiples usuarios de Internet simultáneamente.

Las listas de correos electrónicos distribuyen mensajes a grupos de personas que comparten un interés común, almacenando una serie de direcciones en una base de datos.

Las listas de correo se pueden utilizar con múltiples propósitos. Los más habituales son:

- **Crear foros de discusión sobre temas específicos**

Las personas interesadas en un tema en concreto se suscriben a una lista que trata dicho tema. A partir de ahí reciben mensajes con los comentarios que escriben otros suscriptores y las respuestas correspondientes.

Esto es lo mismo que las news o noticias de Internet. La única diferencia es que en las news los mensajes se almacenan en un servidor de libre consulta, mientras que en la lista se distribuyen al buzón de correo electrónico de los suscriptores.

- **Recibir información periódica sobre un tema**

Así como en los foros de discusión participan todas las personas suscritas, en este tipo de listas suele ser una única persona la que envía los mensajes y los integrantes de la lista quienes los reciben. Por ejemplo, una lista para recibir información sobre actualizaciones y nuevas versiones de un programa.

Funcionamiento y características de las listas.

El sistema se basa en un servidor de listas al que los usuarios envían mensajes para suscribirse, darse baja, pedir información, etc. De esa forma se mantiene un foro de discusión auto-gestionable: no es necesario suscribir o dar de baja las direcciones de correo ya que la operación es efectuada directamente por quienes desean apuntarse a la lista.

Cada lista se define por un nombre y el dominio al que pertenece. El nombre de la lista se asocia a la dirección de correo a la que hay que enviar los mensajes. Por ejemplo, si el dominio **ucf.edu.cu** tiene una lista de nombre **milista**, significa que para enviar mensajes a la lista hay que enviarlos a **milista@ucf.edu.cu**. Los mensajes enviados a esa dirección son los que recibirán todos los suscritos.

Cuando se pretende implementar un servicio a gran escala de listas de correos es muy recomendable utilizar programas especializados en el manejo y administración de listas de discusión. La mayoría de estos programas están desarrollados para máquinas Unix, aunque existen algunas versiones que se pueden instalar en computadoras personales. Entre los más conocidos están:

- Listserv o Listproc (Unix) [23]
- Majordomo (Unix) [24]
- AutoShare (Macintosh) [25]

Estos programas además de ayudar bastante en la administración, implementan servicios automáticos de suscripción, de forma que enviando un mensaje con cierta información a un usuario especial sobre esa máquina, la persona es suscrita automáticamente. Igual procedimiento para des-suscripción. Además de la ventaja de la suscripción automática, estos programas especializados implementan una serie de funcionalidades que permiten entre otras cosas:

- Petición y transferencia automática de archivos.
- Reenvío de mensajes a otros servicios como News.
- Listas distribuidas en varios servidores (peer lists).
- Revisión de listas de suscritos en determinadas listas.

1.3 Descripción del objeto de estudio.

1.3.1 Descripción general.

La red telemática de la Universidad de Cienfuegos (Red UCf) presenta una topología de backbone⁶ concentrado, interconectando 10 subredes que pertenecen a las principales áreas del campus universitario, para ello cuenta con una infraestructura de cableado de fibra óptica y enlaces a 100 Mbps, además cuenta con una infraestructura WAN para la interconexión de las SUMs mediante enlaces Frame Relay a 32 Kbps. También existe un nodo central de la red⁷ donde se concentran los principales recursos de *software* y *hardware* permitiendo implementar las condiciones óptimas necesarias para los servidores que soportan los principales servicios de la

⁶ Backbone, del inglés "espina dorsal" se refiere a la infraestructura para interconectar todas las subredes o redes de una red telemática.

⁷ Nodo central de la red, lugar donde se ubican los principales servidores y equipamiento de red, debe reunir determinados requisitos de climatización, aterramiento, alimentación eléctrica, respaldo eléctrico, respaldo de datos, cableado de datos, gestión y monitoreo, seguridad física y seguridad informática.

Red UCf. De esta forma se centraliza el equipamiento, pero, están creados los mecanismos y los sistemas para una administración descentralizada que se realiza en los nodos locales de cada subred. La Red UCf cuenta con un sistema de correo electrónico basado en servidores con sistema operativo Linux que ejecutan la aplicación servidor de correos Postfix. Existe un servidor principal y un servidor espejo que se actualiza periódicamente de forma automática mediante el protocolo *rsync*. Para la administración descentralizada se utiliza el sistema de administración webadmin [26] permitiendo que cada administrador local pueda administrar cuentas de usuarios remotamente y sin necesidad de cuenta de administración del sistema operativo del servidor. Entre las funciones de los administradores locales podemos mencionar:

- Creación de cuentas de usuarios.
- Cambio de contraseñas de usuarios.
- Habilitar servicios de salida de correo nacional o internacional.
- Creación de alias de correos electrónicos.

1.3.2 Situación problemática.

El envío de correos electrónicos a grupos de usuarios es sin dudas una necesidad en las intranets de cualquier empresa con una infraestructura informática enlazada mediante una red telemática, la Universidad de Cienfuegos es un ejemplo de ello, la utilización de este servicio producto de sus ventajas asociadas a la rápida difusión de informaciones hace que la demanda de utilización del mismo crezca en la misma medida que se expande la red telemática en la universidad.

Este servicio se ha estado implementando mediante el empleo de alias de correos electrónicos en los servidores de correo, sin embargo esta forma de implementación de envío de correos por lote tiene un gran número de problemas que afectan la fiabilidad, seguridad y estabilidad de la red telemática de la universidad.

Entre los problemas más relevantes podemos mencionar la imposibilidad de controlar el uso indiscriminado de las alias, empleadas en ocasiones para la difusión de correos basura (SPAM) y correos con mensajes no acorde a los reglamentos de la

red y el código de ética de la Red UCf, su vulnerabilidad ante los errores humanos cometidos por usuarios inexpertos que pueden provocar grandes congestiones en el servicio, el aprovechamiento de este sistema por parte de virus o códigos maliciosos que se replican vía correo electrónico, la incomodidad para el mantenimiento y administración del sistema y la necesidad de un usuario con privilegios de administración en el sistema operativo del servidor para ello. Todo lo antes mencionado hace que el empleo de alias de correos electrónicos no sea una solución viable para el envío de correos a grupos de usuarios en la Red UCf.

1.3.3 Problema.

Analizando lo planteado anteriormente determinamos el siguiente problema; la no existencia en la Red UCf de un sistema eficiente y seguro para el envío de correos electrónicos a grupos de usuarios.

1.4 Sistemas existentes vinculados al campo de acción.

En este epígrafe se analizan sistemas que pueden ser empleados para enviar correos electrónicos a grupos de usuarios.

Email Marketing Software [27].

Es un paquete de programas destinado al envío masivo de correo, su funcionamiento se basa en la utilización de listas de correo y una de sus principales ventajas es la posibilidad de manejar listas extensas, facilitándole su utilización a los usuarios finales, además incluye utilidades para la obtención automática de direcciones de correo desde Internet, actualizaciones de las listas creadas e interacción con los grupos de noticias de Internet.

Sin embargo no permite una administración descentralizada del sistema y no está orientado a interactuar directamente con los usuarios del servidor de correo y sus direcciones, además de no ser gratis, por lo que no resulta una solución para el problema planteado.

Multi-Mail [28].

Esta aplicación permite enviar mensajes masivos altamente modificados para requisitos particulares, incluyendo cuerpos de mensaje y asuntos personalizados,

archivos adjuntos y documentos HTML. Está orientada para los clientes o usuarios finales, no cuenta con funciones administrativas encaminadas a controlar su utilización y mantenimiento automático de los grupos de direcciones por lo que resuelve los problemas de envío de correo a grupos de usuarios en la red UCf.

Mail Them Pro [28].

Diseñado para el envío de correos masivos y para la administración de listas de correo. Incluye su propio servidor SMTP de alto rendimiento que lo hace independiente del sistema de correo que se utilice. Al igual que otros analizados, sus niveles de administración, control y mantenimiento automático no cumplen con los requisitos esperados, no presentando las funcionalidades necesarias para ser utilizado como sistema de control y envío de correos por lote en la red UCf.

Otros sistemas analizados.

En la búsqueda realizada se encontraron otros sistemas que permiten el envío de correo a grupos de usuarios, pero su funcionamiento se basa en el empleo de alias de correo, por lo que presentan los mismos problemas que fueron analizados anteriormente y que dieron lugar a iniciar este trabajo.

1.5 Lenguaje de Modelación Unificado (UML).

UML es un lenguaje para visualizar, especificar, construir y documentar los artefactos de un sistema que involucra una gran cantidad de software [29]. Es el resultado, en principio, de la unión de los métodos de Booch (Object Oriented Analysis and Design with Application) y Rumabugh (OMT- Object Modeling Technique) para producir lo que se conoció como el Método Unificado, pero que con la unión de Jacobson (OOSE- Object Oriented software Engineering: A use case driven approach) dio paso al Lenguaje Unificado de Modelación. En Noviembre de 1997 este lenguaje (en su versión 1.1) fue adoptado como el estándar por el OMG (Object Modeling Group).

Es importante recalcar que UML no es una guía para realizar el análisis y diseño orientado a objetos, es decir, no es un proceso. UML es un lenguaje que permite la modelación de sistemas con tecnología orientada a objetos.

¿Por qué UML?

La decisión de utilizar UML como notación para el desarrollo del software se debe a que se ha convertido en un estándar que tiene las siguientes características:

- Permite modelar sistemas utilizando técnicas orientadas a objetos.
- Permite especificar todas las decisiones de análisis y diseño, construyéndose así modelos precisos, no ambiguos y completos.
- Puede conectarse con lenguajes de programación (Ingeniería directa e inversa).
- Permite documentar todos los artefactos de un proceso de desarrollo (requisitos, arquitectura, pruebas, versiones, etc.).
- Es un lenguaje muy expresivo que cubre todas las vistas necesarias para desarrollar y luego desplegar los sistemas.
- Existe un equilibrio entre expresividad y simplicidad, pues no es difícil de aprender ni de utilizar.
- UML es independiente del proceso, aunque para utilizarlo óptimamente se debería usar en un proceso que fuese dirigido por los casos de uso, centrado en la arquitectura, iterativo e incremental.

1.6 Proceso Unificado de Desarrollo (RUP).

La metodología que se empleó para el desarrollo de este trabajo fue RUP (Rational Unified Process) [30]. En ella se utiliza UML (Unified Model Language), como lenguaje representativo, específicamente en el caso de este trabajo, se utilizan las extensiones para aplicaciones Web.

Este lenguaje fue creado por un grupo de estudiosos de la Ingeniería de Software formado por: Ivar Jacobson, Grady Booch y James Rumbaugh en el año 1995. Esta metodología es la que mejor se ajusta a las necesidades que existen actualmente en el desarrollo de software, pues propone un Modelo iterativo e incremental, muy acorde con la naturaleza cambiante de los requisitos en muchos proyectos. Es una

metodología que está totalmente respaldada por una excelente herramienta CASE: Rational Rose [31].

Inicialmente, la metodología no era compatible con aplicaciones Web, pero más tarde le fueron añadidos nuevos artefactos con los que quedaban satisfechas las necesidades para este tipo de aplicaciones. Su característica principal, la de ser guiada por los casos de usos, facilita una comprensión natural del proceso de desarrollo del proyecto por parte de los clientes y del equipo de desarrollo.

Esta metodología además, brinda un mejor enfoque del sistema al concebir su distribución y organización en capas y subsistemas con asignaciones de responsabilidades. Concibe también una arquitectura base, donde se priorizan el desarrollo de los casos de usos más importantes y con mayor repercusión para el sistema.

1.7 Tecnologías a utilizar.

Para el desarrollo del sistema se realizó un estudio sobre las posibles herramientas a utilizar en su construcción. Teniéndose en cuenta la tendencia actual y las novedades de cada una de ellas.

1.7.1 Lenguajes.

A continuación se comparan varios lenguajes que se utilizan para la construcción de sitios Web.

1.7.1.1 Hypertext Markup Language (HTML).

Cuando se busca información en la red Internet, se obtienen documentos hipertextos, es decir, documentos con texto que contienen imágenes, sonido o incluso animaciones y, además, enlaces a otros documentos. Además, al publicarse una página Web en un servidor WWW se debe tener en cuenta que se accederá a ella desde diferentes tipos de computadoras, por lo que la información debe presentarse en un formato adecuado, para que pueda ser interpretado correctamente por cualquier computadora conectada a la red.

El HTML (HyperText Markup Language) es un formato de datos para crear documentos de hipertexto que puedan ser entendidos desde computadoras con

diferentes sistemas operativos. Consiste en un sencillo sistema de marcas, que insertadas en un texto, permiten el enlace con otras páginas o recursos (imágenes, sonidos, etc.) y establecen ciertos atributos del documento (dimensiones, alineamiento del texto, estilo, etc.). La flexibilidad que posee es tal que sus características pueden aprovecharse para generar documentos multimedia, aunque no vayan a colocarse en un servidor [32].

HTML es un subconjunto de la especificación SGML (Standard Generalized Markup Language) que es mucho más amplia. Los documentos HTML son SGML con una semántica determinada que es apropiada para representar información de diversos tipos en un único documento.

En definitiva, HTML es un lenguaje de marcas que sigue una definición formal de acuerdo con un estándar. Su potencia se basa en su independencia del sistema operativo y que identifica cada parte de un documento por la finalidad que tiene [33].

HTML no tiene la complejidad de un lenguaje de programación, pues es bastante más sencillo de utilizar y no precisa de ningún compilador. Más bien, es semejante a un documento de texto. Todo documento HTML es almacenado como texto normal en formato ASCII, de manera que cualquier procesador o editor de textos puede ser utilizado para su creación. El navegador es quien se encarga de leer ese documento de texto e interpretar las etiquetas HTML que aparecen en él [34].

1.7.1.2 Professional Home Page Tools (PHP).

PHP (Profesional Home Page Tools) es un lenguaje de programación el cual se ejecuta en los servidores Web y que permite crear contenido dinámico en las páginas HTML, con un lenguaje propietario derivado del Perl.

PHP fue creado por Rasmus Lerdorf a finales de 1994, aunque no hubo una versión utilizable por otros usuarios hasta principios de 1995. Esta primera versión se llamó, Personal Home Page Tools.

Al principio, PHP sólo estaba compuesto por algunas macros que facilitaban el trabajo a la hora de crear una página Web. Hacia mediados de 1995 se creó el analizador sintáctico y se llamó PHP/F1 Versión 2, y sólo reconocía el texto *HTML* y

algunas directivas de MySQL. A partir de este momento, la contribución al código fue pública. El crecimiento de PHP desde entonces ha sido exponencial, y han surgido versiones nuevas como las actuales, PHP4 y PHP5.

Dispone de múltiples herramientas que permiten acceder a bases de datos de forma sencilla, por lo que es ideal para crear aplicaciones para Internet.

Es multiplataforma, funciona tanto para Linux (con Apache) como para Windows (con Microsoft Internet Information Server) de forma que el código que se haya creado para una de ellas no tiene por qué modificarse al pasar a la otra.

El lenguaje PHP es un lenguaje de programación de estilo clásico, con variables, sentencias condicionales, bucles, funciones, entre otras. La sintaxis que utiliza, la toma de otros lenguajes muy extendidos como C y Perl. El código de PHP está incluido en tags especiales "<?, ?>" [35].

El funcionamiento del PHP se puede describir a través de los pasos siguientes:

- Escribir en las páginas HTML pero con el código PHP dentro
- Guardar la página en el servidor Web
- Un navegador solicita una página al servidor
- El servidor interpreta el código PHP
- El servidor envía el resultado del conjunto de código HTML y el resultado del código PHP que también es HTML

En ningún caso se envía código PHP al navegador, por lo que todas las operaciones realizadas son transparentes al usuario, el código PHP es ejecutado en el servidor y el resultado enviado al navegador. El resultado es normalmente una página HTML. Por lo que al usuario le parecerá que está visitando una página HTML que cualquier navegador puede interpretar.

Al ser PHP un lenguaje que se ejecuta en el servidor no es necesario que el navegador lo soporte, es independiente del navegador, pero sin embargo para que sus páginas PHP funcionen, el servidor donde están alojadas debe soportar PHP [35].

PHP se encuentra libre en el mercado y se puede acceder a él por medio de Internet.

1.7.1.3 Active Server Page (ASP).

Microsoft introdujo la tecnología llamada Active Server Pages (ASP) en diciembre de 1996, como parte de la versión 3.0 de la aplicación Servidor Web Internet Information Server (IIS). Es una tecnología de páginas activas que permite el uso de diferentes scripts y componentes Active X en conjunto con el tradicional HTML para mostrar páginas generadas dinámicas.

Traduciendo la definición de Microsoft se puede entender que es ASP: "Las Active Server Pages son un ambiente de aplicación abierto y gratuito en el que se puede combinar código HTML, scripts y componentes ActiveX del servidor para crear soluciones dinámicas y poderosas para el Web" [36]. ASP es una tecnología dinámica funcionando del lado del servidor, lo que significa que cuando el usuario solicita un documento ASP, las instrucciones de programación dentro del script son ejecutadas para enviar al navegador únicamente el código HTML resultante.

Las aplicaciones ASP han sido bastante amplias. Entre las funciones principales están: el acceso a base de datos, envío de correo electrónico, creación dinámica de gráficos y otros. Básicamente, muchas cosas que se pueden realizar por medio de CGI, explicada en el próximo subepígrafe, pueden ser realizadas con esta tecnología.

Como se ha dicho, permite combinar HTML y código Script para crear páginas Web altamente interactivas. Soporta los mismos lenguajes Script popularmente conocidos para el uso del lado del cliente: JavaScript y VBScript, convirtiendo esto en su primera ventaja.

Una potencialidad importante de ASP está en incluir controles ActiveX especialmente contruidos para ser ejecutados en el servidor. Esta integración de ASP con los componentes activos del servidor permite el procesamiento robusto de aplicaciones en el servidor [37], [38]. De esta forma, ASP puede crear estos componentes con lenguajes de alto nivel como el Visual C++, con el objetivo de procesar datos y luego retornar los resultados al usuario a través del ASP.

El paradigma de desarrollo de ASP difiere en gran medida de la programación script del lado del cliente. En el lado del cliente, el script se incrusta dentro de la página que es enviada al usuario y es ejecutado por el Browser. Si el navegador no reconoce el lenguaje del script, entonces ignorará el código. Por el contrario, con ASP, todos los scripts son procesados en el servidor y los resultados son retornados al cliente en formato HTML estándar reconocido por cualquier navegador [39].

1.7.1.4 ¿Por qué PHP?

Luego de hacer el análisis entre el PHP y el ASP, se decide utilizar el PHP embebido en el código HTML ya que:

Está soportado en la mayoría de las plataformas de Sistemas Operativos, mientras que ASP por ser propiedad de Microsoft no es multiplataforma.

El PHP no tiene costo oculto, o sea que cuando se adquiere incluye un sinnúmero de bibliotecas que proporcionan el soporte para la mayoría de las aplicaciones Web, por ejemplo e-mail, generación de ficheros PDF y otros. En caso de que no se tengan las bibliotecas estas se pueden encontrar gratis en Internet. En el caso de ASP forma parte del Internet Information Server que viene integrado en Windows NT-2000 Server con su elevado costo de adquisición.

PHP y ASP son parecidos en cuanto a la forma de utilización, pero PHP es más rápido, gratuito y multiplataforma.

1.7.1.5 Java Script.

JavaScript originariamente denominado LiveScript, es un lenguaje de scripts compacto basado en objetos (y no orientado a objetos). Fue desarrollado por la compañía Netscape para su navegador Netscape Navigator 2.0 [40]. Actualmente Javascript corre en la mayoría de los sistemas operativos, incluyendo los entornos de Microsoft. Este lenguaje permite la realización de aplicaciones de propósito general a través de la WWW, aunque no está diseñado para el desarrollo de grandes aplicaciones, es suficiente para la implementación de aplicaciones WWW completas o interfaces WWW hacia otras más complejas.

Por ejemplo, una aplicación escrita en JavaScript puede ser incrustada en un documento HTML proporcionando un mecanismo para la detección y tratamiento de eventos, como clicks del ratón o validación de entradas realizadas en los formularios para la entrada de datos.

Javascript está diseñado para controlar la apariencia y manipular los eventos dentro de la ventana del navegador Web y aunque tiene algunas similitudes con Java, considerado su hermano mayor, es un lenguaje con sus propias características [41].

Siendo mucho más modesto y sencillo que Java, se basa en un modelo de instanciación de objetos muy simple para el que no es necesario tener conocimiento de conceptos tales como herencia y jerarquías. Además, soporta un sistema en tiempo de ejecución basado en un pequeño número de tipos de datos (numérico, boolean, y string) en el que ni siquiera es necesario declarar el tipo de variables. Para programar en JavaScript sólo es necesario un editor. No requiere de un ambiente de desarrollo ni de un compilador, como en la generalidad de los lenguajes, pues al ser un código que se interpreta en tiempo de ejecución por el navegador Web, es fácil de implementar y mantener.

1.6.3 Sistemas Gestores de Base de Datos (SGDB).

Una Base de Datos (BD) es un conjunto de datos interrelacionados, almacenados con carácter más o menos permanente en la computadora, puede ser considerada una colección de datos variables en el tiempo.

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

El objetivo fundamental de un SGBD consiste en suministrar al usuario las herramientas que le permitan manipular, en términos abstractos, los datos, o sea, de forma que no le sea necesario conocer el modo de almacenamiento de los datos en la computadora, ni el método de acceso empleado.

Un SGBD tiene los siguientes objetivos específicos:

- Independencia de los datos y los programas de aplicación

- Minimización de la redundancia
- Integración y sincronización de las bases de datos
- Integridad de los datos
- Seguridad y protección de los datos
- Facilidad de manipulación de la información
- Control centralizado

La información es representada a través de tuplas, las cuales describen al fenómeno, proceso o ente de la realidad objetiva que se está analizando y se representan a través de tablas.

1.7.3.1 SQL (Structure Query Language).

Definido como un lenguaje de base de datos normalizado, de propósito específico y no procedural, SQL (Structured Query Language), traducido al español como “Lenguaje de Consulta Estructurado” soporta la definición, manipulación y el control de los datos en sistemas de bases de datos relacionales. Es un lenguaje de propósito especial porque sólo puede ser utilizado para manipular bases de datos y es por eso, es conocido también como un sublenguaje de datos, que puede ser usado con lenguajes de aplicación. SQL no es procedural, pues el mismo no especifica como algo debe hacerse, sino qué debe hacerse, en otras palabras, a SQL sólo le interesan los resultados en vez de cómo obtenerlos [42].

Casi siempre se cita a SQL como un idioma neutral para los sistemas de gestión de bases de datos relacionales. Efectivamente, ningún otro lenguaje ha ganado tan amplia aceptación entre un rango tan amplio de productos [42]. Desde que se estandarizó por primera vez en 1986, SQL ha sido adoptado universalmente. Además, a diferencia de otros lenguajes de programación como C, que son de dominio exclusivo de los programadores, SQL es empleado por gran variedad de profesionales; programadores, administradores de Bases de Datos y analistas de negocios entre otros, que utilizan SQL para acceder a la información.

Para utilizar el lenguaje SQL, no es necesario ser un experto. Un principiante, teniendo un conocimiento básico del lenguaje, puede realizar gran cantidad de tareas. La única forma de que pueda llevarse a cabo alguna acción dentro de SQL es mediante la ejecución de instrucciones. Estas caen en tres categorías [42]:

- Instrucciones del Lenguaje de Definición de Datos (DDL),
- Instrucciones del Lenguaje de Manipulación de Datos(DML)
- Instrucciones del Lenguaje de Control de Datos (DCL)

Lo más importantes de este lenguaje es que muchos fabricantes de Bases de Datos lo han adoptado, convirtiéndose en el lenguaje estándar de consulta para interactuar con Sistemas de Bases de Datos Relacionales. Muchos sistemas de BD populares, desde PCs hasta Mainframes, lo soportan. SQL no es solamente un lenguaje de consulta pues mediante su uso, también pueden crearse tablas, así como añadir, actualizar y eliminar la información contenida en estas.

1.7.3.2 MySQL.

MySQL es un sistema de administración de Base de Datos. Opera en una arquitectura cliente/servidor. Es un proyecto “Open Source”. Permite la fácil conectividad, alta velocidad de respuesta a solicitudes, y gran seguridad, por ello se utiliza para acceder a Bases de Datos desde Internet. [43] [44]

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

El lenguaje PHP es altamente compatible con MySQL, por el amplio conjunto de comandos definidos para el tratamiento de este.

1.7.3.3 SQL Server.

Microsoft SQL Server, propietario de Microsoft, pertenece a la familia de los sistemas de administración de base de datos, operando en una arquitectura cliente/servidor de gran rendimiento. Su desarrollo fue orientado para hacer posible manejar grandes volúmenes de información, y un elevado número de transacciones. SQL Server es

una aplicación completa que realiza toda la gestión relacionada con los datos. El servidor sólo tiene que enviarle una cadena de caracteres (la sentencia SQL) y esperar a que le devuelvan los datos. [45]

SQL Server permite la creación de procedimientos almacenados, los cuales consisten en instrucciones SQL que se almacenan dentro de una base de datos de SQL Server, realizados en lenguaje SQL, se trata de procedimientos que se guardan semicompilados en el servidor y que pueden ser invocados desde el cliente. Se ejecutan más rápido que instrucciones SQL independientes. [46]

SQL Server puede manejar perfectamente bases de datos de TeraBytes con millones de registros y funciona sin problemas con miles de conexiones simultáneas a los datos, sólo depende de la potencia del hardware del equipo en el que esté instalado y solamente está disponible para sistemas operativos de Microsoft.

1.7.3.4 ¿Por qué MySQL?

Tanto el SQL Server como el MySQL operan en una arquitectura cliente/servidor, de tal manera que este sólo tiene que enviarle una cadena de caracteres (la sentencia SQL) y esperar la devolución de los datos.

Luego de analizadas las características y facilidades de los SGBD presentados, y de la herramienta a desarrollar, se decide usar el MySQL como SGBD, por las siguientes razones:

- No se necesitará de un manejo complejo de la información.
- El lenguaje PHP es altamente compatible con MySQL, por el amplio conjunto de comandos definidos para el tratamiento de este.
- MySQL es multiplataforma.
- MYSQL es una tecnología *Open Source*, se adquiere libremente.

1.8 Conclusiones.

El estudio realizado en este capítulo expone la importancia que tiene la utilización de sistemas de envío de correos electrónicos a grupos de usuarios, demostrando las deficiencias asociadas a los métodos tradicionales utilizados para este fin. Además se analizan críticamente las soluciones utilizadas en la red de la Universidad de Cienfuegos basadas en el empleo de alias de correos, determinándose la necesidad de desarrollar e implementar un sistema que solucione los problemas actuales, exponiéndose finalmente las metodologías, herramientas y tecnologías a emplear en el desarrollo de la solución propuesta.

Capítulo 2. Descripción de la solución propuesta.

2.1 Introducción.

En el presente capítulo se describe la solución propuesta utilizando la Metodología RUP, empleando para ello el Modelo del Dominio, los Requerimientos Funcionales y No Funcionales, el Diagrama de Casos de Uso y la descripción de cada uno, así como una descripción general del sistema que se propone.

2.2 Descripción del Modelo del Dominio.

Un modelo del domino captura los tipos más importantes de objetos en el contexto del sistema. Los objetos del dominio representan las "cosas" que existen o los eventos que suceden en el entorno en el que trabaja el sistema. Muchos de los objetos del dominio o clases pueden obtenerse de una especificación de requisitos o mediante la entrevista con los expertos del dominio [30].

La modelación del dominio tiene como objetivo fundamental la comprensión y descripción de las clases más importantes en el sistema.

Para poder entender el contexto en que se desarrolla el sistema se definen los principales conceptos relacionados con el entorno del problema.

2.2.1 Principales conceptos asociados al dominio del problema.

Usuario: Se refiere a los usuarios creados en el sistema de envío de correos por lote y los mismos pueden ser estudiantes o trabajadores de la universidad.

Grupos: Se refiere a conjuntos de direcciones de correo electrónico que se agrupan según las necesidades de los usuarios. Los grupos pueden ser asociados a uno o muchos usuarios.

Direcciones: Se refiere a las direcciones de correo electrónico de todos los usuarios de la universidad y las mismas pueden pertenecer a uno o muchos grupos.

Áreas: Se refiere a las áreas a las que pertenecen los usuarios y son determinadas por la dirección del centro.

Direcciones IP: Se refiere a las direcciones IP de las computadoras que acceden al sistema.

2.2.2 Representación del Modelo del Dominio.

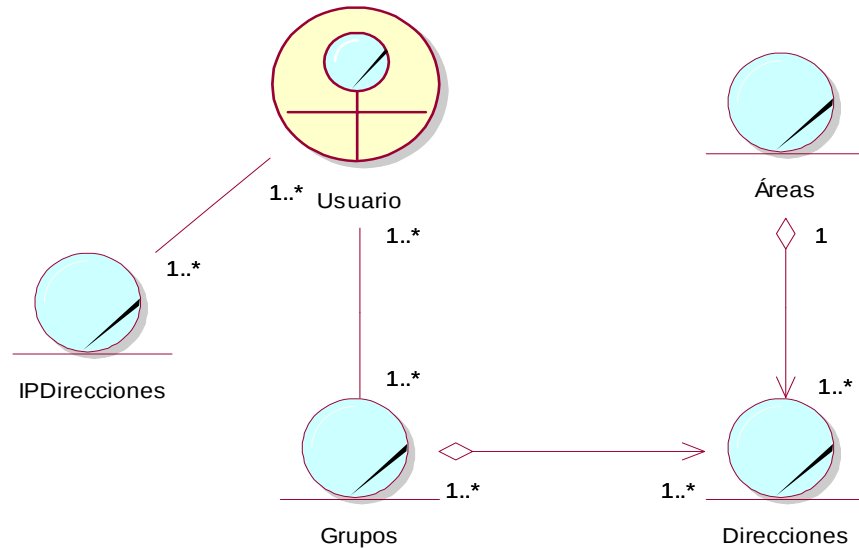


Figura 2.1. Modelo del dominio.

2.2.3 Reglas del negocio a considerar.

- Todos los usuarios del servicio de correo electrónico de la universidad no tienen derecho a enviar correos por lote.
- El rector de la universidad determina los autorizados a enviar correos múltiples a todos los usuarios de la red UCf.
- Los jefes de áreas determinan quiénes pueden enviar correos a todos los miembros de la misma.
- Los administradores de redes locales crean los grupos de direcciones de correo según las indicaciones de los jefes de áreas y las necesidades de la administración de la red.
- Los correos enviados a grupos de direcciones deben cumplir con los requisitos del reglamento de la UCf, el código de ética de la red y las indicaciones del rector para el uso de la misma.

- Los grupos de direcciones de correo sólo deben ser utilizados por los usuarios de la red autorizados y estos no deberán permitir el uso de los mismos por otros usuarios no autorizados, siendo responsables de su adecuada utilización.
- No utilizar correos múltiples con mensajes adjuntos, para circular grandes volúmenes de información utilizar los servidores de las redes locales y la intranet de la UCf.
- Se debe evitar el reenvío de los mensajes recibidos por los usuarios finales a todos los usuarios de los grupos de direcciones de correo.
- La dirección de correo electrónico del administrador de la red forma parte de todos los grupos de direcciones de correos, permitiendo así un control del uso correcto de este servicio.
- Usar correos múltiples para enviar informaciones no acordes con el reglamento de la red, el código de ética y las indicaciones del rector de la universidad llevará a la cancelación del servicio por parte de la administración de la red UCf.

2.3 Descripción del sistema propuesto.

SECL (Sistema de Envío de Correos por Lote), surge por la necesidad de eliminar el empleo de alias de correo electrónico para el envío de correos por lote, debido a las desventajas asociadas a su utilización.

2.3.1 Concepción general.

El resultado que se pretende alcanzar en esta investigación es la obtención de un producto de software que permita el envío de correos electrónicos a grupos de usuarios de forma controlada y a la vez permita la creación, modificación y eliminación de grupos de direcciones de correo, implementando para ello diferentes opciones de administración que posibilitan gestionar usuarios y grupos de direcciones. Con este sistema se eliminarán los problemas existentes en los métodos utilizados actualmente en la universidad para realizar estas funciones y se le dará solución al problema planteado en el Capítulo 1: Fundamentos Teóricos.

El software permitirá a los administradores configurar grupos de direcciones de correo, asociar los grupos a usuarios, descentralizar la administración del servicio y mantener actualizado el sistema con todas las direcciones de correos existente en la universidad. Además se implementarán mecanismos de control y protección de los datos encaminados a garantizar los niveles de seguridad informática necesarios.

2.3.2 Requerimientos funcionales.

Los requerimientos funcionales permiten expresar una especificación más detallada de las responsabilidades del sistema que se propone. Ellos permiten determinar, de una manera clara, lo que debe hacer el mismo [30]

1. Crear usuarios.
2. Eliminar usuarios.
3. Visualizar usuarios.
4. Asignar privilegios a usuarios.
5. Modificar privilegios de usuarios.
6. Configurar acceso desde IP.
7. Importar direcciones de correo.
8. Crear áreas.
9. Modificar áreas.
10. Eliminar áreas.
11. Visualizar información de las áreas.
12. Crear grupos de direcciones.
13. Modificar grupos de direcciones.
14. Eliminar grupos de direcciones.
15. Visualizar grupos de direcciones.
16. Añadir grupos de direcciones a usuarios.
17. Eliminar grupos de direcciones asociados a un usuario.

18. Añadir direcciones a grupos.
19. Eliminar direcciones de un grupo.
20. Buscar usuarios.
21. Buscar grupos de direcciones.
22. Buscar direcciones.
23. Cambiar contraseña de administradores de áreas.
24. Cambiar contraseña de usuarios.
25. Visualizar direcciones de un grupo.
26. Enviar correo a grupos de direcciones.
27. Mostrar ayuda general.
28. Mostrar ayuda específica.
29. Autenticar.
30. Cerrar sesión.
31. Cambiar contraseña del administrador general.

2.3.3 Requerimientos no funcionales.

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener [30].

Apariencia o interfaz externa

El sistema contará con una interfaz Web sencilla, con un diseño que le permita al usuario tener en todo momento el control de la aplicación, lo que le permitirá ir de un punto a otro dentro de ella con gran facilidad mostrando todas las opciones disponibles según los privilegios asignados.

Los mensajes de error deberán ser reportados por la propia aplicación y no por el Sistema Operativo.

Usabilidad

Los usuarios que harán uso del sistema serán creados por el mismo y forman parte del conjunto de usuarios del servicio de correo electrónico de la universidad, que son: profesores, trabajadores y estudiantes, permitiéndoles enviar correos a grupos de usuarios de una forma simple y segura, brindando la posibilidad de acceder a la ayuda del sistema en todo momento.

Soporte

La instalación y mantenimiento de la aplicación serán responsabilidad de los administradores de la red de la universidad. Las pruebas del sistema se realizarán con anterioridad a la puesta en funcionamiento del mismo.

El sistema deberá permitir futuras mejoras y nuevas opciones que se le quieran incorporar, además de poder interactuar con otros sistemas existentes adaptándose a las características de la red donde se desee implantar.

Portabilidad

El sistema podrá ser utilizado sobre el sistema operativo Linux.

Seguridad

El sistema debe garantizar confidencialidad, integridad, disponibilidad, fiabilidad, estabilidad y no repudio en el manejo de los datos almacenados y enviados a través de la red. Para esto se debe tener en cuenta:

- Establecer niveles de privilegio para los diferentes tipos de usuarios.
- El control de acceso que permita la identificación y validación de cada usuario, garantizando que tenga disponible solamente las opciones asociadas a su nivel de privilegio.
- Cifrado de los datos almacenados en la base de datos, utilizando para ellos la función MD5.
- Seguridad de los datos enviados por la red, utilizando el protocolo SSL.

- Garantizar el no repudio incorporando la dirección del usuario en los correos enviados por él y una dirección de control en todos los grupos de direcciones creados.

Ayuda y documentación en línea

La aplicación contará con una ayuda en línea a la que se podrá acceder en cualquier momento. Existirán opciones de ayuda específica para aclarar determinadas funciones. En el *Capítulo 3: Construcción de la Solución Propuesta*, se realiza una descripción más detallada de este tema.

Software

El sistema se desarrollará con tecnología PHP y gestor de base de datos MySQL para servidores Web Apache, permitiendo su instalación en plataformas Linux. Los clientes podrán acceder desde diferentes navegadores Web, en diferentes sistemas operativos.

Hardware

Los requerimientos de hardware son impuestos por el servidor de correo debido a que el sistema se instalará en el mismo, los usuarios que deseen acceder al sistema lo podrán hacer desde estaciones de trabajo clientes con los siguientes requerimientos mínimos:

- Microprocesador, Pentium a 175 MHz de velocidad o similares.
- Memoria Física, 32 MB de memoria RAM.

2.4 Modelo de Casos de Uso del Sistema.

El modelo de casos de uso permite que los desarrolladores del software y los clientes lleguen a un acuerdo sobre los requisitos, es decir, sobre las condiciones y posibilidades que debe cumplir el sistema. Describe lo que hace el sistema para cada tipo de usuario [30].

2.4.1 Actores del sistema.

Un actor no es más que un conjunto de roles que los usuarios de Casos de Uso desempeñan cuando interaccionan con estos Casos de Uso. Los actores

representan a terceros fuera del sistema que colaboran con el mismo. Una vez que hemos identificado los actores del sistema, tenemos identificado el entorno externo del sistema [30].

Actor	Justificación
Usuario	Es el que utiliza el sistema para enviar correos a grupos de direcciones. Puede visualizar las direcciones incluidas en los grupos que tiene asociados, cambiar su contraseña y visualizar la ayuda en cualquier momento. Requerimientos asociados: 24, 25, 26, 27, 28, 29, 30.
Administrador de Área	Realiza las mismas funciones del usuario; además de administrar el sistema según su nivel de privilegio, gestiona los usuarios y los grupos de direcciones, configura las relaciones entre ellos y realiza búsquedas bajo diferentes criterios. Requerimientos asociados: 1, 2, 3, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30.
Administrador General	Realiza las mismas funciones del Administrador de Área; adicionándole la posibilidad de asignar y modificar los privilegios de los usuarios, gestionar todo lo referente a las áreas, actualizar la información de la tabla de direcciones y configurar las direcciones IP desde donde se podrá acceder al sistema. Requerimientos asociados: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31.

Tabla 2.1 Actores del sistema.

2.4.2 Casos de Uso del sistema.

Cada forma en que los actores usan el sistema se representa con un caso de uso. Los casos de uso son “fragmentos” de funcionalidad que el sistema ofrece para aportar un resultado de valor para sus actores. De manera más precisa, un caso de uso especifica una secuencia de acciones que el sistema puede llevar a cabo interactuando con sus actores, incluyendo alternativas dentro de la secuencia [30].

Se definieron los siguientes casos de uso para el sistema propuesto:

1. Gestionar usuarios.
2. Actualizar privilegios.
3. Configurar acceso desde IP.
4. Importar direcciones de correo.
5. Gestionar áreas.
6. Gestionar grupos de direcciones.
7. Gestionar grupos de direcciones asociados a un usuario.
8. Gestionar direcciones de un grupo.
9. Buscar información.
10. Cambiar contraseña de administradores de áreas.
11. Cambiar contraseña de usuarios.
12. Visualizar direcciones de un grupo.
13. Enviar correo a grupos de direcciones.
14. Mostrar ayuda.
15. Autenticar.
16. Cerrar sesión.
17. Cambiar contraseña del administrador general.

2.4.3 Diagrama de Casos de Uso del sistema.

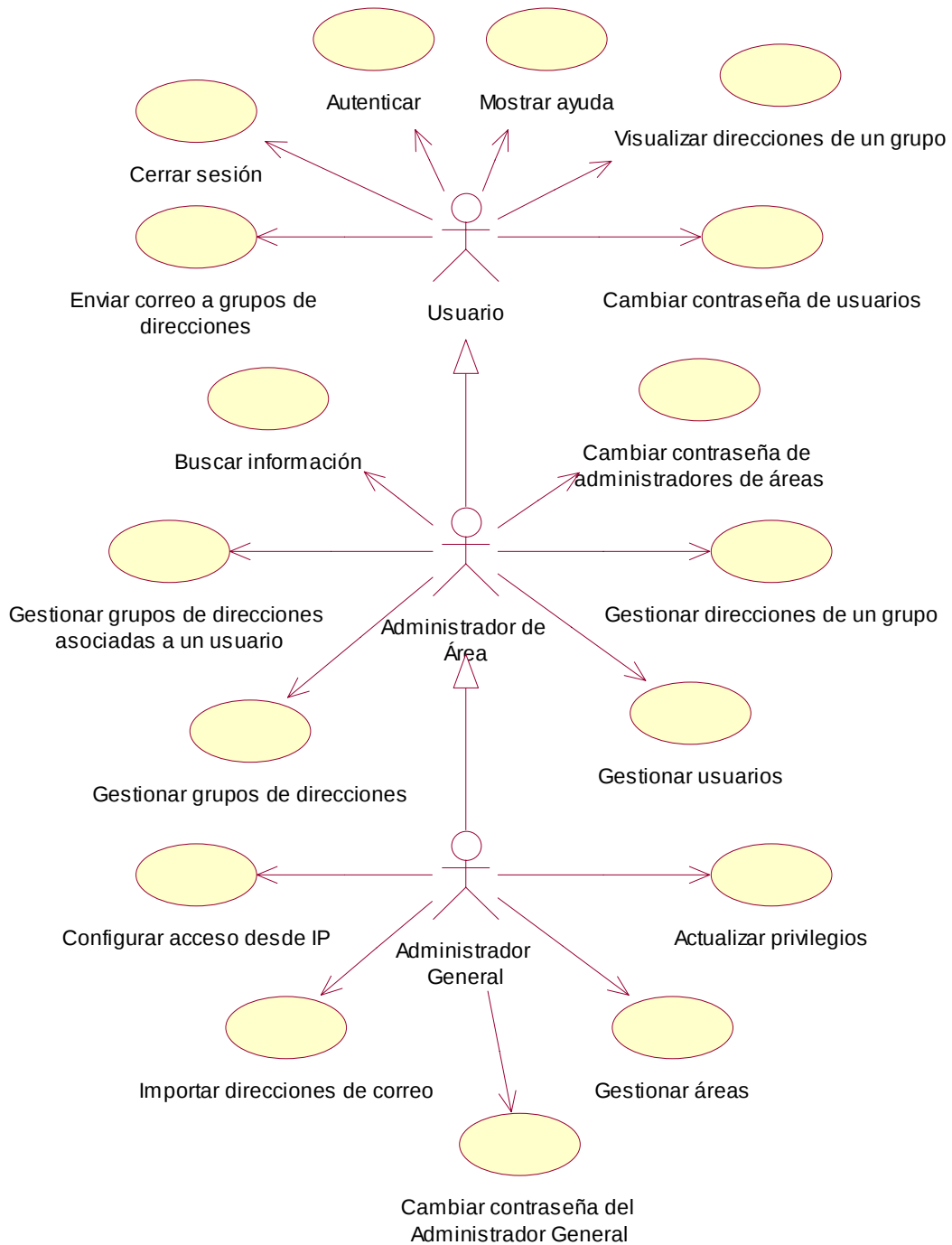


Figura 2.2 Diagrama de Casos de Uso del sistema.

2.4.3 Descripción de los Casos de Uso del sistema.

Caso de uso	Gestionar usuarios
Actores	Administrador de Área (inicia).
Propósito	Gestionar los usuarios del sistema.
Resumen	El caso de uso se inicia cuando el Administrador de Área desea gestionar usuarios, para ello puede crear, eliminar o visualizar un usuario. Si desea crearlo selecciona los datos necesarios: login y nombre. Para eliminarlo se selecciona el mismo de una lista mostrada por el sistema donde se visualizan todos los usuarios existentes. El caso de uso culmina con la actualización de los datos en el sistema.
Referencias	R1, R2, R3.
Precondiciones	Debe estar almacenado en la base de datos el usuario que se desea eliminar. Debe estar almacenada en la base de datos la información a partir de la cual se crea el usuario deseado.
Post-condiciones	Se actualizan los cambios en la base de datos.
Requisitos Especiales	No pueden existir usuarios repetidos.
Prototipo	Ver anexo 2 Prototipo 2.1

Caso de uso	Actualizar privilegios
Actores	Administrador General (inicia)
Propósito	Asignar y modificar privilegios a los usuarios.
Resumen	El caso de uso se inicia cuando el Administrador General desea asignar o modificar los privilegios de usuarios. Para ello asigna el tipo de usuario (administrador de área o usuario) en el proceso

	de creación del mismo o lo modifica en la opción de actualizar privilegios de usuarios. El caso de uso culmina con la actualización de los privilegios en el sistema.
Referencias	R4, R5.
Precondiciones	Debe existir en la base de datos el usuario al que se le quieren actualizar los privilegios.
Post-condiciones	Se actualizan los datos modificados.
Requisitos Especiales	Al usuario Administrador General no se le puede cambiar el privilegio.
Prototipo	Ver anexo 2 Prototipo 2.2

Caso de uso	Configurar acceso desde IP
Actores	Administrador General (inicia)
Propósito	Permitir o denegar el acceso al sistema desde máquinas específicas.
Resumen	El caso de uso se inicia cuando el Administrador General desea permitir o denegar el acceso al sistema desde determinadas máquinas, para ello debe insertar las direcciones IP de las máquinas a las que quiere bloquear o autorizar y seleccionar la opción deseada (denegar o permitir), finalizando así el caso de uso.
Referencias	R6.
Precondiciones	-
Post-condiciones	Se actualiza la tabla de direcciones IP de la base de datos.
Requisitos Especiales	El sistema debe permitir por defecto el acceso desde todas las máquinas.
Prototipo	Ver anexo 2 Prototipo 2.3

Caso de uso	Importar direcciones de correo
Actores	Administrador General (inicia)
Propósito	Importar los datos necesarios desde el sistema operativo.
Resumen	El caso de uso se inicia cuando el Administrador General desea importar los datos necesarios para llenar la tabla de direcciones de la base de datos a partir de la información disponible en el servidor de correo, permitiendo esto incorporar o actualizar la información que contiene la misma, culminando así el caso de uso.
Referencias	R7
Precondiciones	-
Post-condiciones	Se actualiza la tabla direcciones de la base de datos.
Requisitos Especiales	Debe sincronizar la información existente en la tabla direcciones según los datos existentes en el servidor.
Prototipo	Ver anexo 2 Prototipo 2.4

Caso de uso	Gestionar áreas
Actores	Administrador General (inicia)
Propósito	Gestionar las áreas en el sistema.
Resumen	El caso de uso se inicia cuando el Administrador General desea crear, modificar, eliminar o visualizar un área, si lo deseado es adicionar un área debe proporcionar los datos necesarios para ello y si lo que desea es modificar o eliminar, se visualizan las áreas con sus datos seleccionando el área deseada, culminando el caso de uso con la actualización de los datos.

Referencias	R8, R9, R10, R11.
Precondiciones	Debe estar almacenada en la base de datos el área que se desea modificar o eliminar.
Post-condiciones	Se actualizan los cambios en la base de datos.
Requisitos Especiales	-
Prototipo	Ver anexo 2 Prototipo 2.5

Caso de uso	Gestionar grupos de direcciones
Actores	Administrador de Área (inicia)
Propósito	Gestionar los grupos del sistema.
Resumen	El caso de uso se inicia cuando el Administrador de Área desea gestionar los grupos de direcciones, para ello puede crear, modificar o eliminar un grupo de direcciones. En caso de crearlo se deben proporcionar los datos necesarios y en caso de modificar o eliminar se visualizan los grupos para que se especifique a cual se le quiere realizar la acción. El caso de uso finaliza con la creación o actualización de los grupos.
Referencias	R12, R13, R14, R15.
Precondiciones	Debe estar almacenado en la base de datos el grupo que se desea modificar o eliminar.
Post-condiciones	Se actualizan los cambios en la base de datos.
Requisitos Especiales	-
Prototipo	Ver anexo 2 Prototipo 2.6

Caso de uso	Gestionar grupos de direcciones asociadas a un usuario.
Actores	Administrador de Área (inicia)

Propósito	Añadir o eliminar grupos de direcciones asociadas a un usuario.
Resumen	El caso de uso se inicia cuando el Administrador de Área desea añadir o eliminar algún grupo asociado a un determinado usuario, si lo que desea es adicionar un grupo se selecciona el usuario al que se le quiere añadir el grupo y si lo que desea es eliminar un grupo se visualizan todos los grupos asociados a un usuario para seleccionar el grupo o los grupos a eliminar, culminando el caso de uso con la actualización de los datos.
Referencias	R16, R17.
Precondiciones	Tienen que estar almacenados en la base de datos los usuarios y los grupos a utilizar.
Post-condiciones	Se actualizan los grupos de direcciones asociados al usuario deseado.
Requisitos Especiales	-
Prototipo	Ver anexo 2 Prototipo 2.7

Caso de uso	Gestionar direcciones de un grupo.
Actores	Administrador de Área (inicia)
Propósito	Añadir o eliminar las direcciones de un grupo.
Resumen	El caso de uso se inicia cuando el Administrador de Área desea añadir o eliminar las direcciones asociadas a un grupo, si lo que desea es adicionar una o varias direcciones se seleccionan de una lista y se incorporan al grupo deseado, si lo deseado es eliminar una o varias direcciones se selecciona el grupo y se visualizan las direcciones que contiene permitiendo seleccionar

	las que se desean eliminar, culminando el caso de uso con la actualización de los datos.
Referencias	R18, R19.
Precondiciones	Tienen que estar almacenados en la base de datos las direcciones y los grupos que se desean utilizar.
Post-condiciones	Se actualizan los cambios en la base de datos.
Requisitos Especiales	-
Prototipo	Ver anexo 2 Prototipo 2.8

Caso de uso	Buscar información
Actores	Administrador de Área (inicia)
Propósito	Buscar información de los datos almacenados.
Resumen	El caso de uso se inicia cuando el Administrador de Área desea buscar alguna información relacionada con usuarios, grupos o direcciones, seleccionando previamente los criterios de búsqueda. El caso de uso finaliza con la visualización de los datos buscados.
Referencias	R20, R21, R22.
Precondiciones	Tiene que estar almacenada en la base de datos la información de los usuarios, grupos o direcciones que se deseen buscar.
Post-condiciones	-
Requisitos Especiales	-
Prototipo	Ver anexo 2 Prototipo 2.9

Caso de uso	Cambiar contraseña de administradores de áreas
Actores	Administrador de Área (inicia)

Propósito	Permitir cambiar la contraseña de los administradores.
Resumen	El caso de uso se inicia cuando el Administrador de Área desea cambiar su contraseña que se encuentra almacenada en el sistema, para ello debe insertar su cuenta de usuario, la contraseña anterior y la nueva, si los datos son correctos la contraseña queda cambiada, de lo contrario se muestra un mensaje de error, finalizando así el caso de uso.
Referencias	R23.
Precondiciones	La contraseña anterior insertada tiene que coincidir con la almacenada en la base de datos.
Post-condiciones	Se actualiza la contraseña en la base de datos.
Requisitos Especiales	-
Prototipo	Ver anexo 2 Prototipo 2.10

Caso de uso	Cambiar contraseña de usuarios
Actores	Usuario (inicia)
Propósito	Permitir cambiar la contraseña a los usuarios.
Resumen	El caso de uso se inicia cuando el Usuario desea cambiar su contraseña que es la misma utilizada para el acceso al servicio de correo electrónico, para ello el sistema le da la posibilidad de poner su cuenta de usuario, la contraseña anterior y la nueva, si los datos son correctos la contraseña queda cambiada, de lo contrario se muestra un mensaje de error, finalizando así el caso de uso.
Referencias	R24.
Precondiciones	-
Post-condiciones	-

Requisitos Especiales	El sistema tiene que interactuar con el servidor de correo para validar la contraseña anterior.
Prototipo	Ver anexo 2 Prototipo 2.11

Caso de uso	Visualizar direcciones de un grupo
Actores	Usuario (inicia)
Propósito	Permitir a los usuarios visualizar las direcciones de un grupo.
Resumen	El caso de uso se inicia cuando el usuario desea conocer las direcciones pertenecientes a un determinado grupo, para ello selecciona el grupo del cual desea que se muestren las direcciones que contiene, culminando el caso de uso con la visualización de la lista de direcciones.
Referencias	R25.
Precondiciones	El usuario tiene que tener algún grupo asociado, este grupo tiene que tener direcciones incorporadas, existiendo estas en la base de datos.
Post-condiciones	-
Requisitos Especiales	
Prototipo	Ver anexo 2 Prototipo 2.12

Caso de uso	Enviar correo a grupos de direcciones
Actores	Usuario (inicia)
Propósito	Enviar correos a lotes de usuarios.
Resumen	El caso de uso se inicia cuando el usuario desea enviar un correo a un grupo de direcciones de correo electrónico pertenecientes a diferentes usuarios del servicio de correo electrónico. Para

	ello selecciona el grupo o los grupos a los que desea enviarle información. Culminando el caso de uso con el envío del mensaje.
Referencias	R26.
Precondiciones	El usuario debe tener al menos un grupo de direcciones asociado.
Post-condiciones	-
Requisitos Especiales	Debe enviarse una copia del correo generado a una cuenta de supervisión, siendo esto transparente al usuario.
Prototipo	Ver anexo 2 Prototipo 2.13

Caso de uso	Mostrar ayuda
Actores	Usuario (inicia)
Propósito	Proporcionar al usuario la ayuda necesaria.
Resumen	El caso de uso se inicia cuando el Usuario necesita obtener información referente a alguna función del sistema. Para ello tiene la posibilidad de acceder a una ayuda general o específica en dependencia sus necesidades. Culminando el caso de uso con la visualización de la información deseada.
Referencias	R27, R28.
Precondiciones	Debe existir la ayuda correspondiente.
Post-condiciones	-
Requisitos Especiales	-
Prototipo	Ver anexo 2 Prototipo 2.14

Caso de uso	Autenticar
Actores	Usuario (inicia)

Propósito	Restringir el nivel de acceso a la información registrada en el sistema.
Resumen	El caso de uso se inicia cuando el Usuario desea entrar al sistema, para ello escribe su nombre de usuario y contraseña, si el usuario pertenece al sistema se valida su tipo y se chequea la contraseña según el tipo de usuario, si los datos proporcionados son correctos se le permite iniciar sesión; terminando así el caso de uso.
Referencias	R29.
Precondiciones	El usuario tiene que existir en la base de datos.
Post-condiciones	Se inicia una sesión de usuario.
Requisitos Especiales	Si es un usuario que intenta iniciar sesión es del tipo Usuario se chequea la contraseña contra el servidor de correos.
Prototipo	Ver anexo 2 Prototipo 2.15

Caso de uso	Cerrar sesión
Actores	Usuario (inicia)
Propósito	Cerrar una sesión iniciada por un usuario.
Resumen	El caso de uso se inicia cuando el usuario desea terminar su sesión, para ello debe seleccionar la opción desconectar del sistema, terminando así el caso de uso.
Referencias	R30.
Precondiciones	Debe existir una sesión abierta.
Post-condiciones	Se cierra la sesión abierta.
Requisitos Especiales	La sesión que esté inactiva por más de 10 minutos se debe cerrar automáticamente.
Prototipo	Ver anexo 2 Prototipo 2.16

Caso de uso	Cambiar contraseña del administrador general
Actores	Administrador General (inicia)
Propósito	Permitir cambiar la contraseña del administrador general.
Resumen	El caso de uso se inicia cuando el Administrador General desea cambiar su contraseña que se encuentra almacenada en el sistema, para ello debe insertar la contraseña nueva, quedando cambiada la contraseña anterior, finalizando así el caso de uso.
Referencias	R31.
Precondiciones	-
Post-condiciones	Se actualiza la contraseña en la base de datos.
Requisitos Especiales	-
Prototipo	Ver anexo 2 Prototipo 2.17

2.4 Conclusiones.

En este capítulo se ha descrito la solución propuesta utilizando el Modelo del Dominio, en el que se definieron los conceptos asociados al dominio del problema y sus relaciones. Se definieron también los requerimientos funcionales y no funcionales y los Casos de Uso con su descripción de alto nivel.

Capítulo 3. Construcción de la solución propuesta.

3.1 Introducción.

En el presente capítulo se realiza una descripción de la construcción de la solución propuesta, para ello se han utilizado los diagramas de clases web del diseño y el modelo lógico y físico de la base de datos. Se describen los principios de diseño utilizados, mostrando ejemplos de cómo son presentados al usuario y la concepción general de la ayuda. Para describir los elementos fundamentales de la implementación se muestra el diagrama de implementación.

3.2 Diagrama de clases del diseño.

En las aplicaciones Web, el diagrama de clases representa las colaboraciones que ocurren entre las páginas, donde cada página lógica puede ser representada como una clase, además permite representar el nivel de abstracción adecuado y la relación con los restantes artefactos de UML.

A continuación se muestran los diagramas de clases web desarrollados a partir de los casos de uso del sistema.

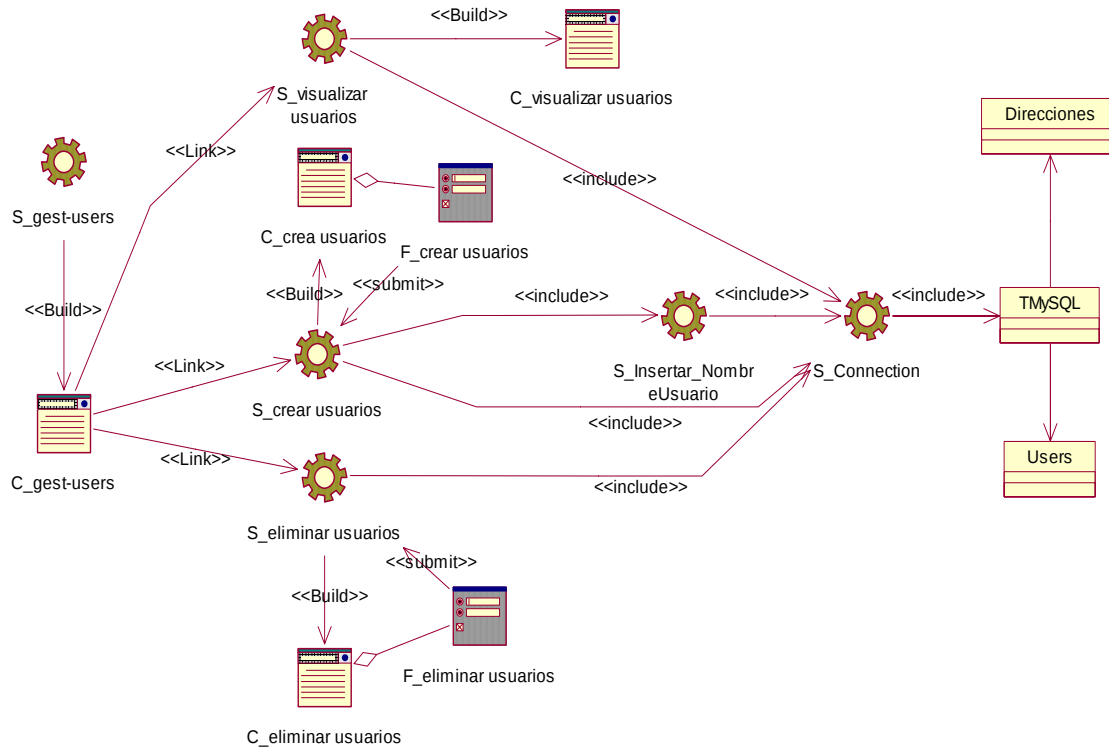


Figura 3.1 Gestionar usuarios

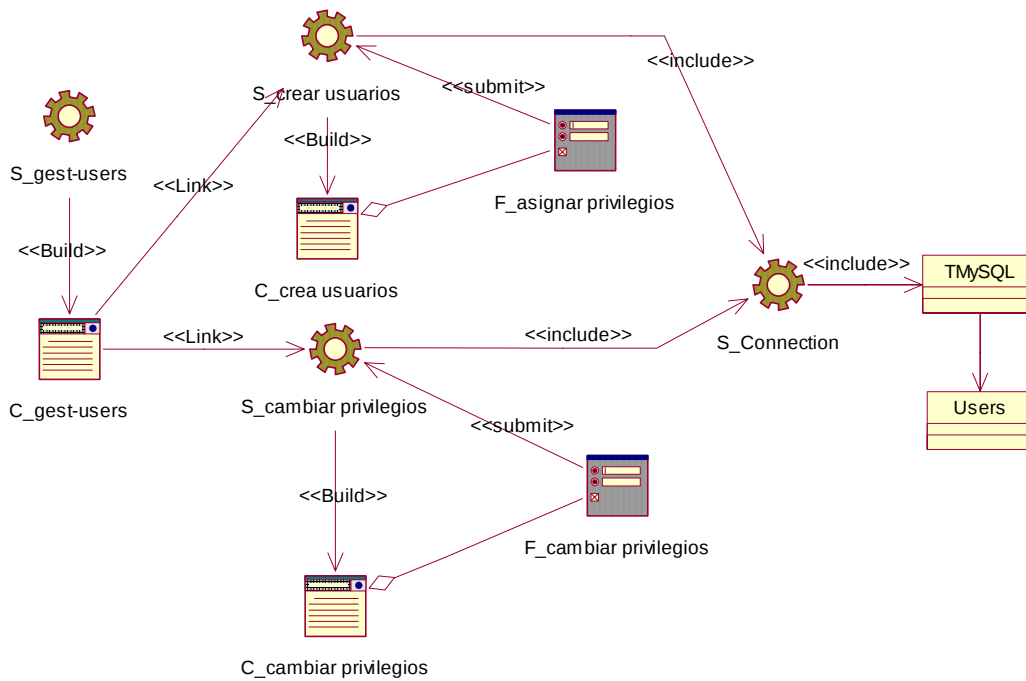


Figura 3.2 Actualizar privilegios

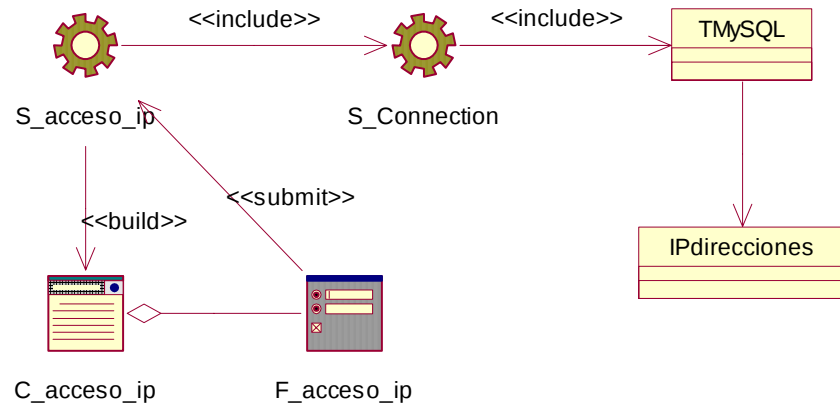


Figura 3.3 Configurar acceso desde IP

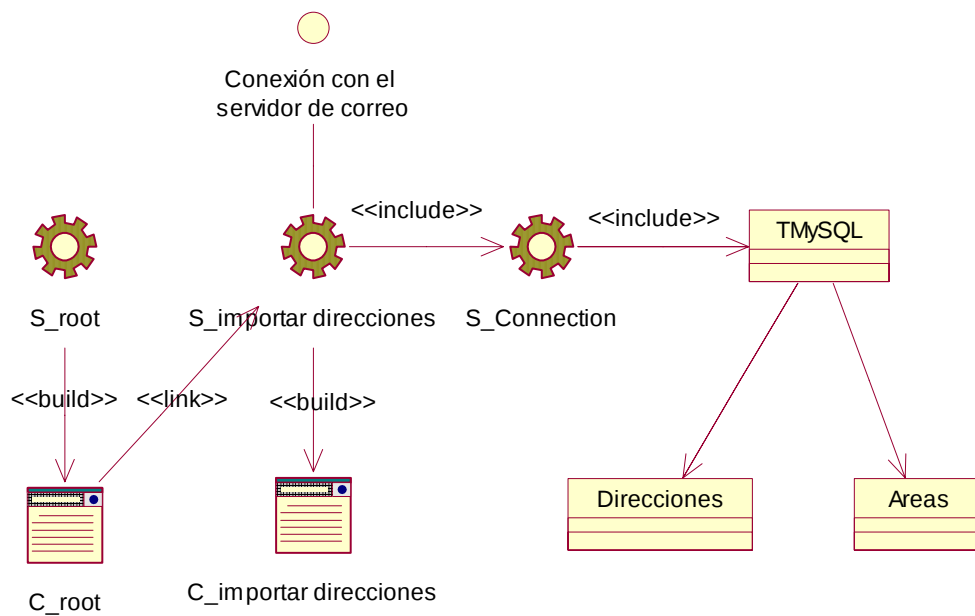


Figura 3.4 Importar direcciones de correo

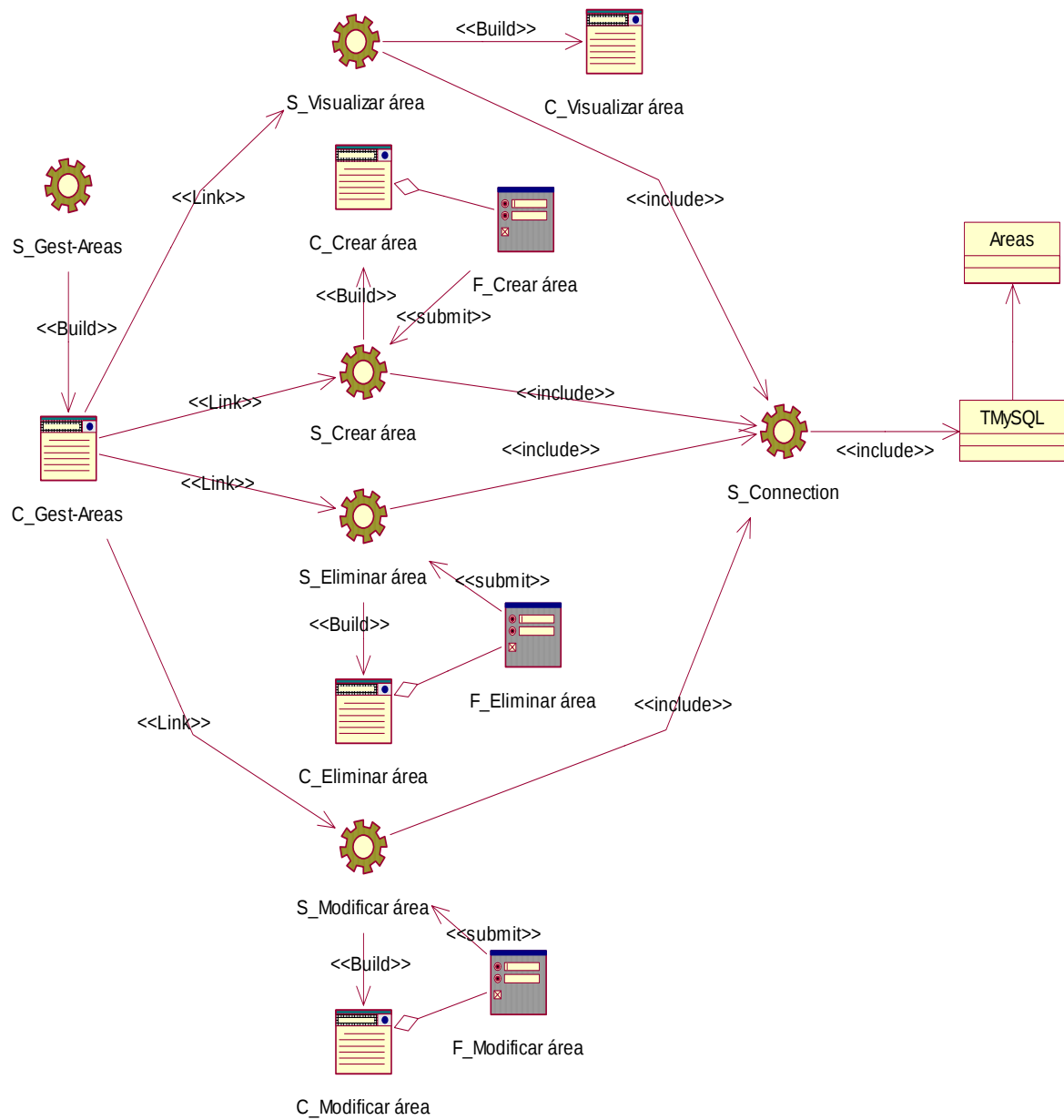


Figura 3.5 Gestionar áreas

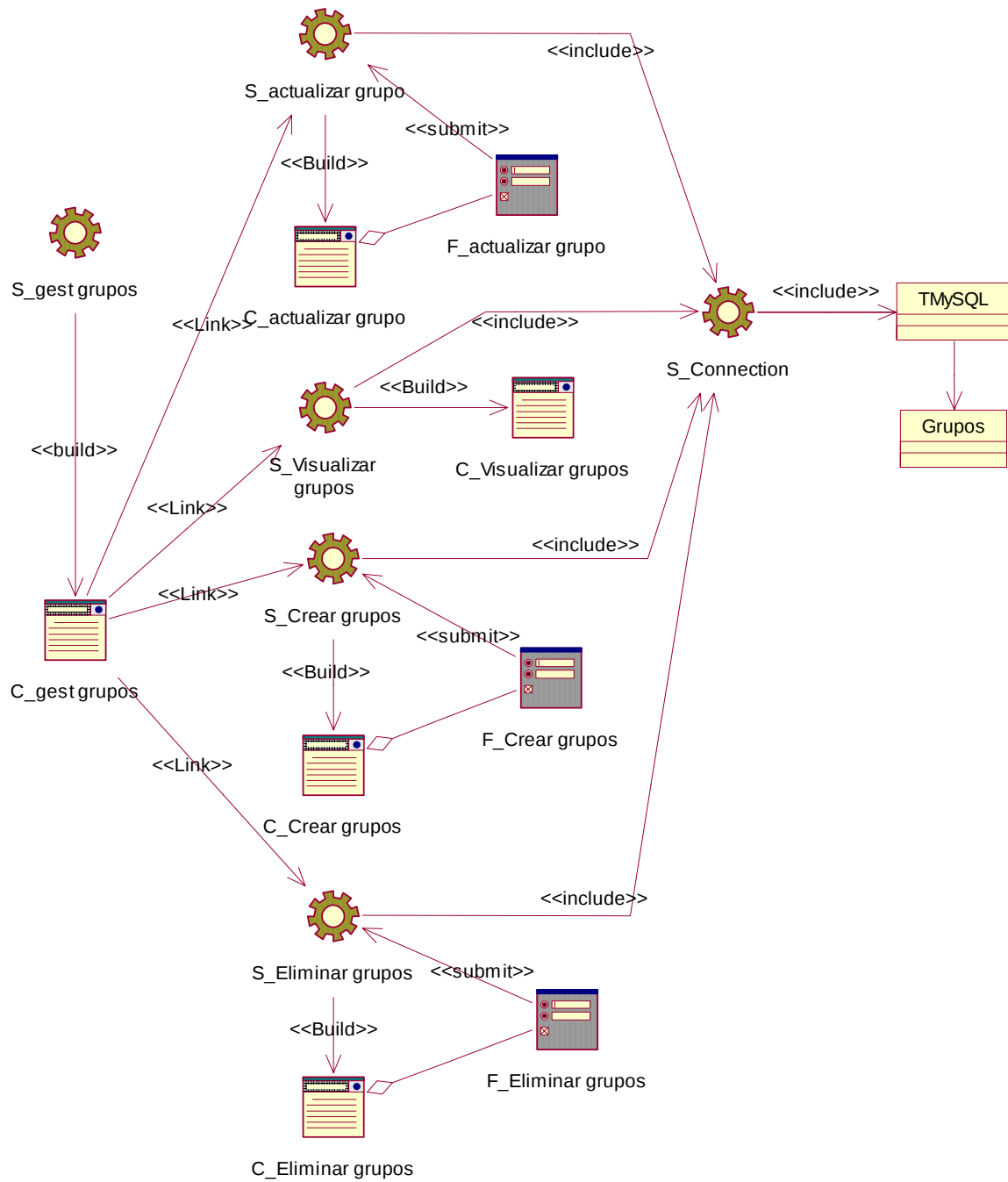


Figura 3.6 Gestionar grupos de direcciones

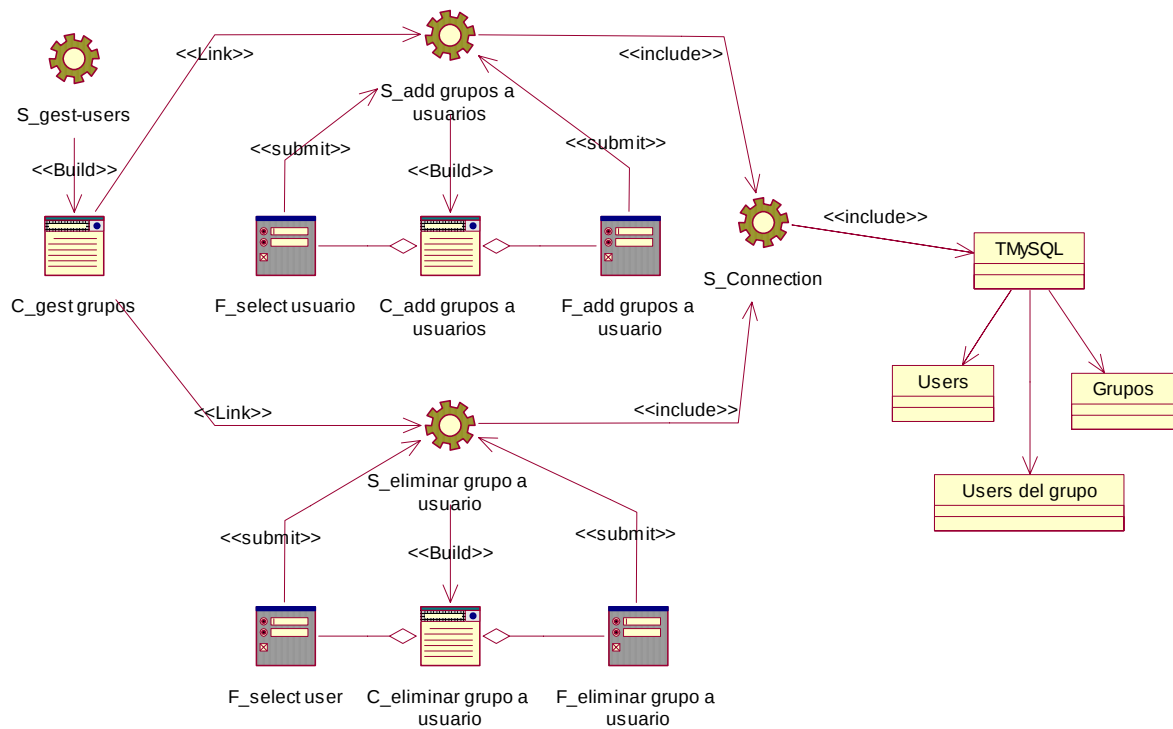


Figura 3.7 Gestionar grupos de direcciones asociados a un usuario

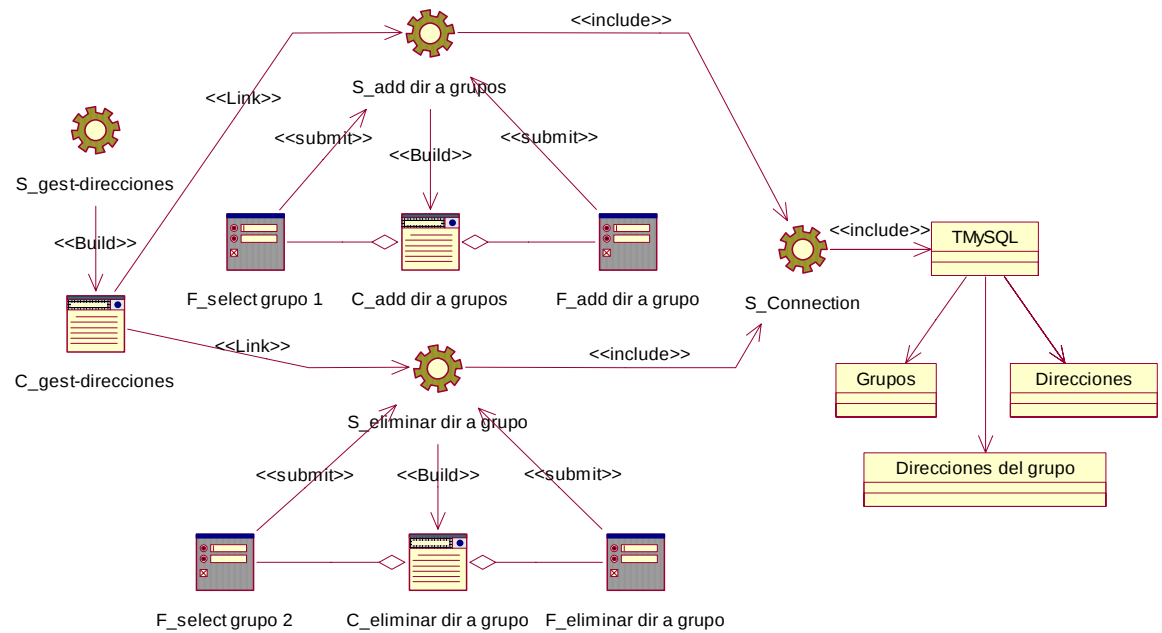


Figura 3.8 Gestionar direcciones de un grupo

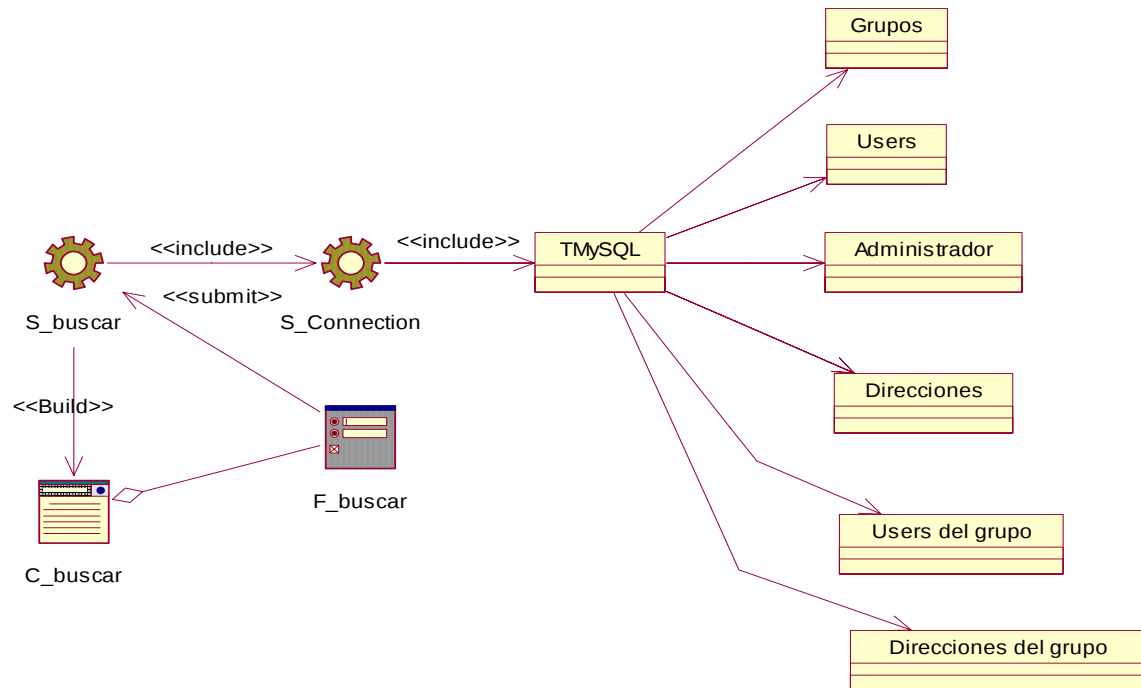


Figura 3.9 Buscar información

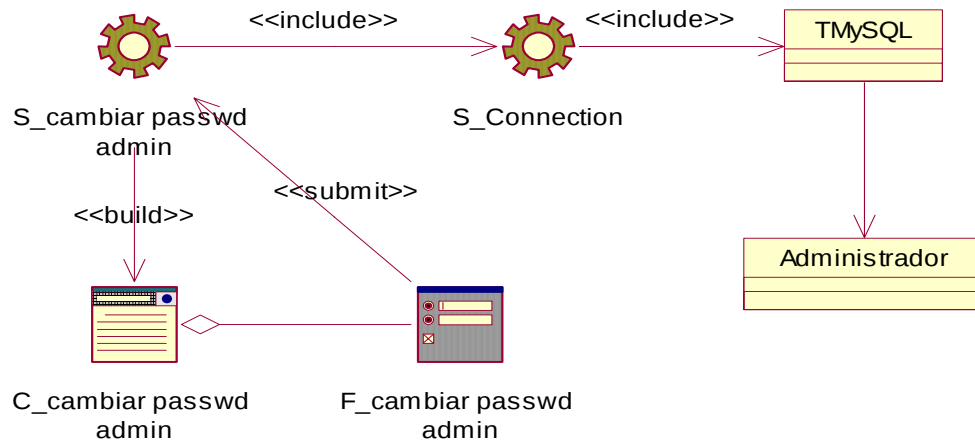


Figura 3.10 Cambiar contraseña de administradores de áreas

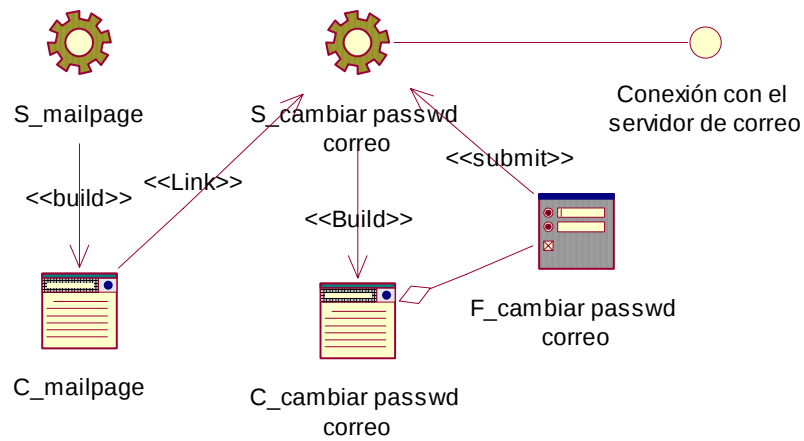


Figura 3.11 Cambiar contraseña de usuarios

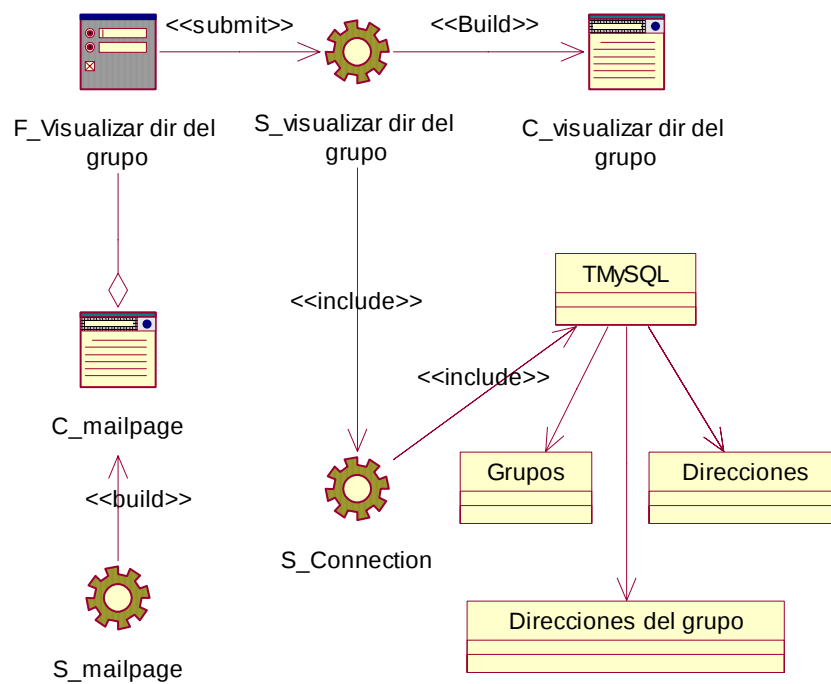


Figura 3.12 Visualizar direcciones de un grupo

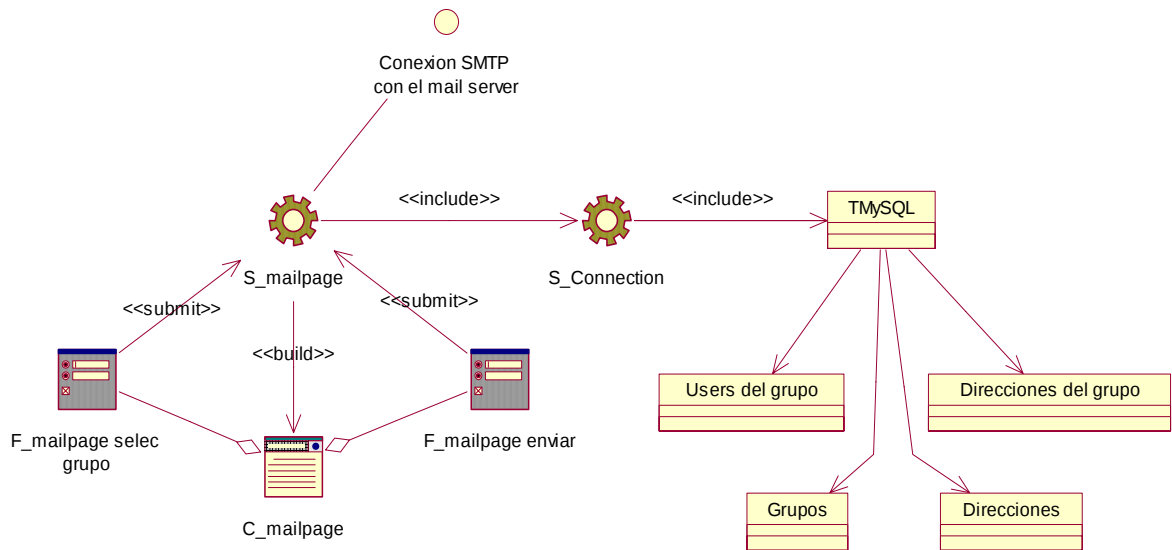


Figura 3.13 Enviar correo a grupos de direcciones

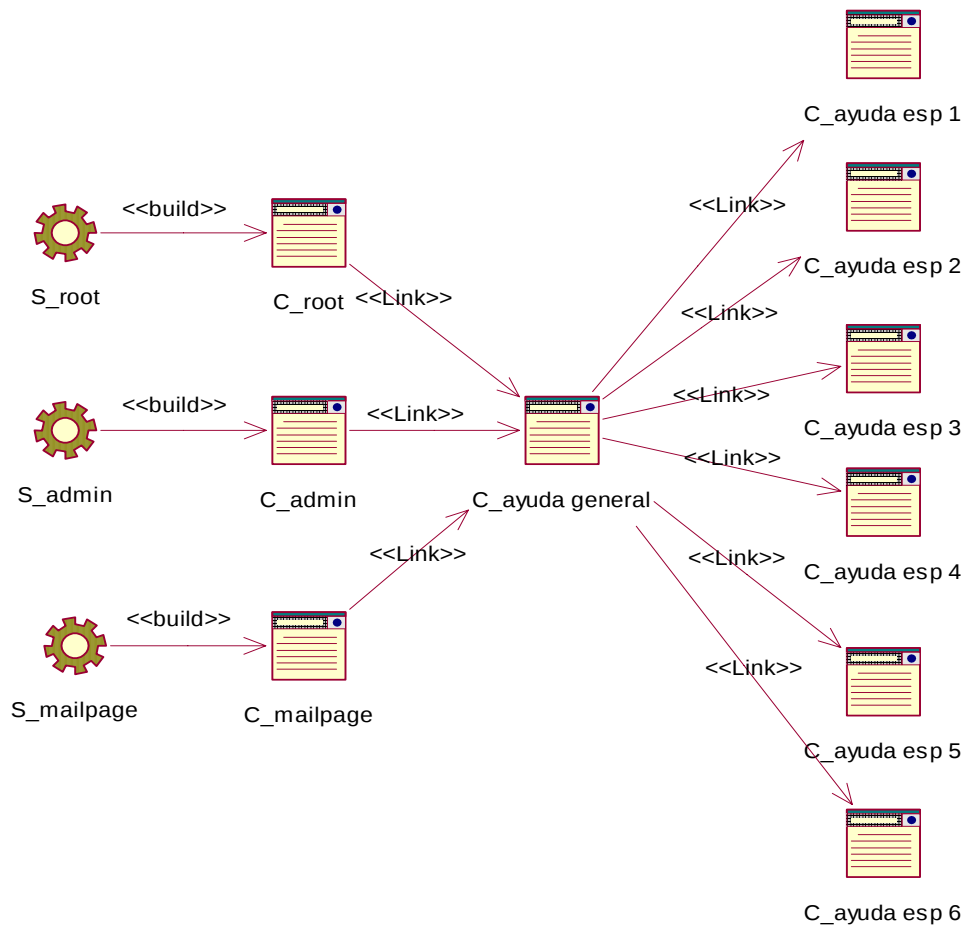


Figura 3.14 Mostrar ayuda

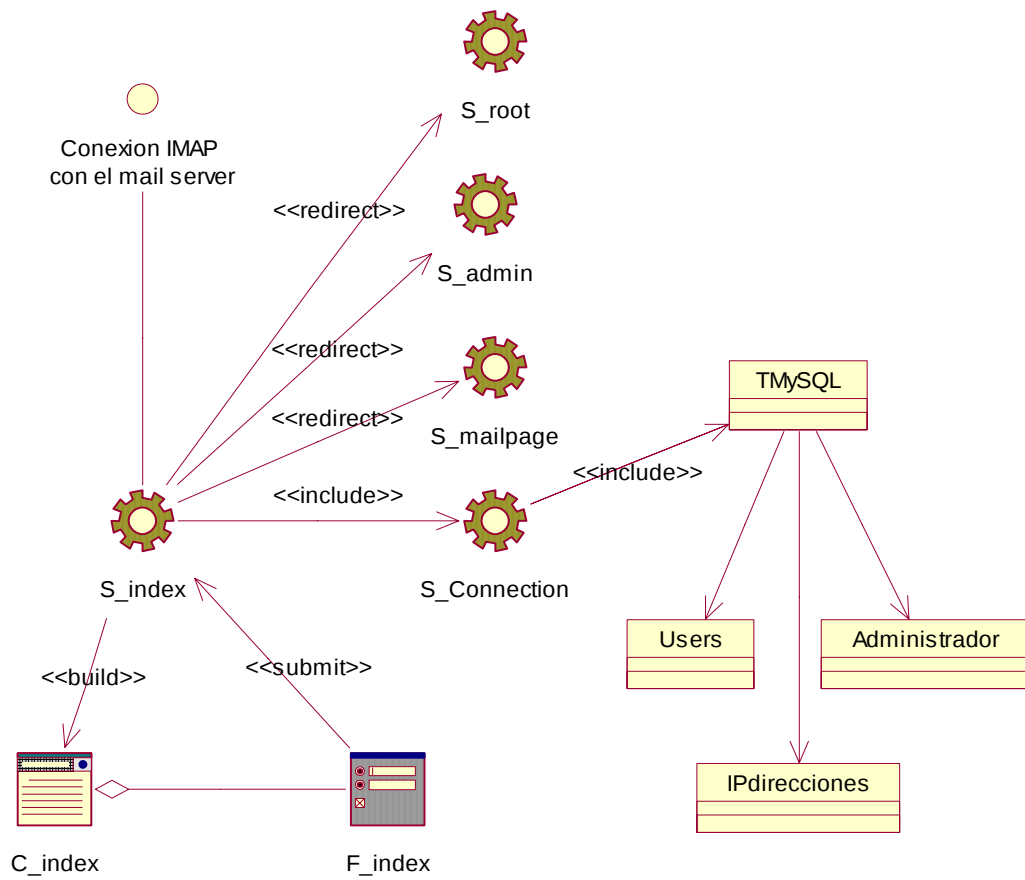


Figura 3.15 Autenticar

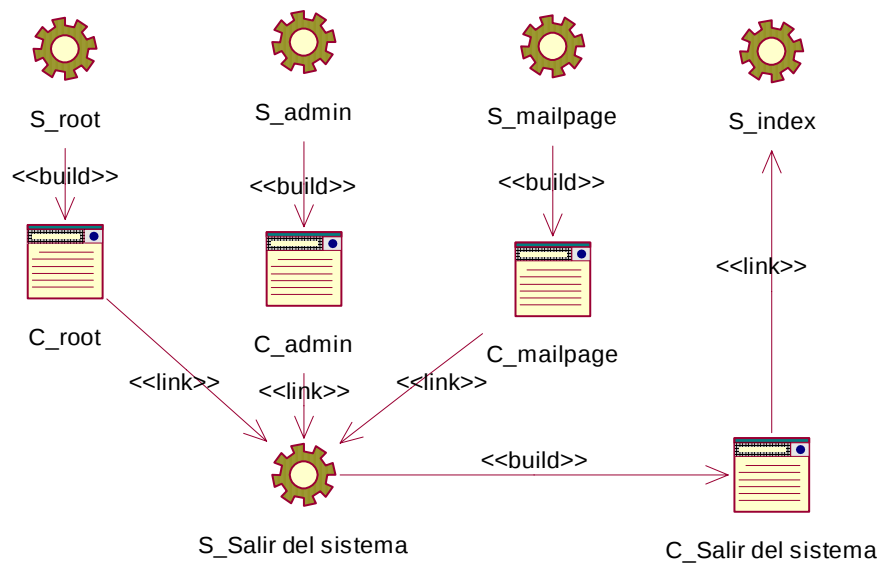


Figura 3.16 Cerrar sesión

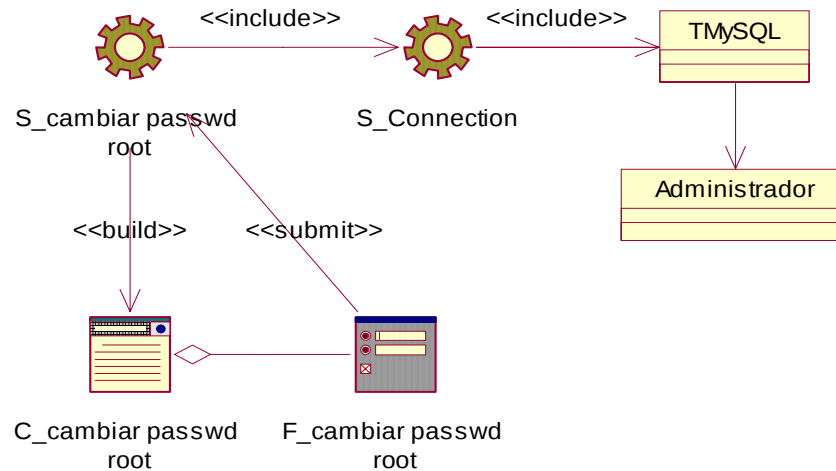


Figura 3.17 Cambiar contraseña del administrador general

3.3 Principios de diseño.

A continuación se describen los principios de diseño seguidos para el desarrollo del sistema. Es importante mencionar que el mismo está orientado, en el caso del módulo de envío de correos, a facilitar la rapidez y eficiencia en su utilización y en el caso de los módulos administrativos a garantizar su funcionalidad y comodidad en la gestión y administración descentralizada y remota.

3.3.1 Estándares en la interfaz de la aplicación.

En todo el sistema se contrastan colores con tonalidad azul oscuro, azul claro y blanco (azul oscuro para la imagen empleada como banner, las líneas y las letras utilizadas en todo el sistema, blanco y azul claro para el fondo), además la letra debe ser legible y sencilla (se utiliza el tipo de letra verdana) para que unido al color utilizado contrasten con el fondo y se vean refrescantes a la vista del usuario, se evita el uso excesivo de imágenes, animaciones y efectos visuales, en beneficio de una interfaz sencilla y rápida. Se mantuvo un diseño único en todo el sistema que proporciona un menú en la parte derecha del sitio el cual está presente en todas las páginas de administración, permitiendo que los administradores naveguen por el sistema manteniendo la posibilidad de acceso a todas las opciones en todo momento. En el caso del módulo de envío de correos se incorpora la posibilidad de

acceder directamente a las opciones de cambio de contraseña de correo y visualización de grupos asociados al usuario y las direcciones que los componen.

El sistema permite la visualización de usuarios, direcciones y grupos de direcciones con el objetivo de facilitar las tareas a realizar, para ello se mantiene el diseño antes mencionado y se adicionan las opciones necesarias en cada caso, organizándose en tablas las informaciones obtenida de la Base de Datos, las cuales tienen un encabezado referente a la información mostrada.

3.3.2 Concepción general de la ayuda.

El sistema cuenta con una ayuda general la cual se encuentra disponible al usuario en cada momento, permitiéndole navegar por sus tópicos y consultar cualquier duda sobre el sistema, lo cual permite que el usuario pueda hacer un uso eficiente de la aplicación, la ayuda mantiene los mismos principios de diseño pero se muestra en una página independiente. Además existen ayudas específicas para diferentes opciones del sistema, a la que se puede acceder desde un enlace indicado por un signo de interrogación ubicado a continuación de la opción que lo requiera y mostrándose la ayuda en una pequeña ventana independiente.

3.3.3 Tratamiento de errores.

El sistema propuesto trata de evitar en todo momento la introducción de errores para ello muestra a cada usuario la información asociada a su privilegio y aprovecha constantemente las opciones de selección de listas para minimizar la entrada de errores por teclado. Además presenta mecanismos de validación de la información, con el propósito de minimizar las posibilidades de introducir información errónea. En caso de errores se le comunica el error cometido tanto en las páginas como en cuadros de alerta. Los mensajes de error que emite el sistema se muestran en un lenguaje de fácil comprensión para los usuarios.

3.4 Diseño de la base de datos.

El diseño de la base de datos se realiza con el propósito de asegurarse que los datos persistentes son almacenados consistente y eficientemente y definir el comportamiento que debe ser implementado en la base de datos.

Para ello se desarrolla el Modelo lógico y físico de la base de datos.

3.4.1 Modelo lógico de datos.

El Modelo lógico provee una vista de las entidades lógicas de datos y sus relaciones, con independencia de la plataforma de base de datos a utilizar.

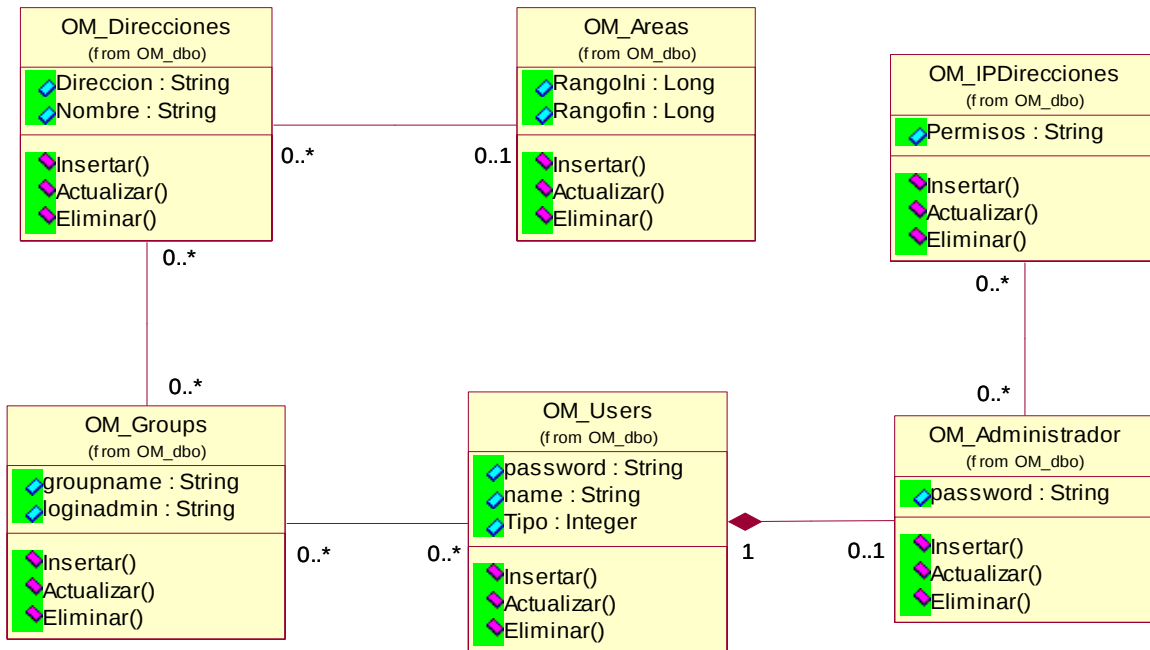


Figura 3.18 Modelo lógico de datos

3.4.2 Modelo físico de datos.

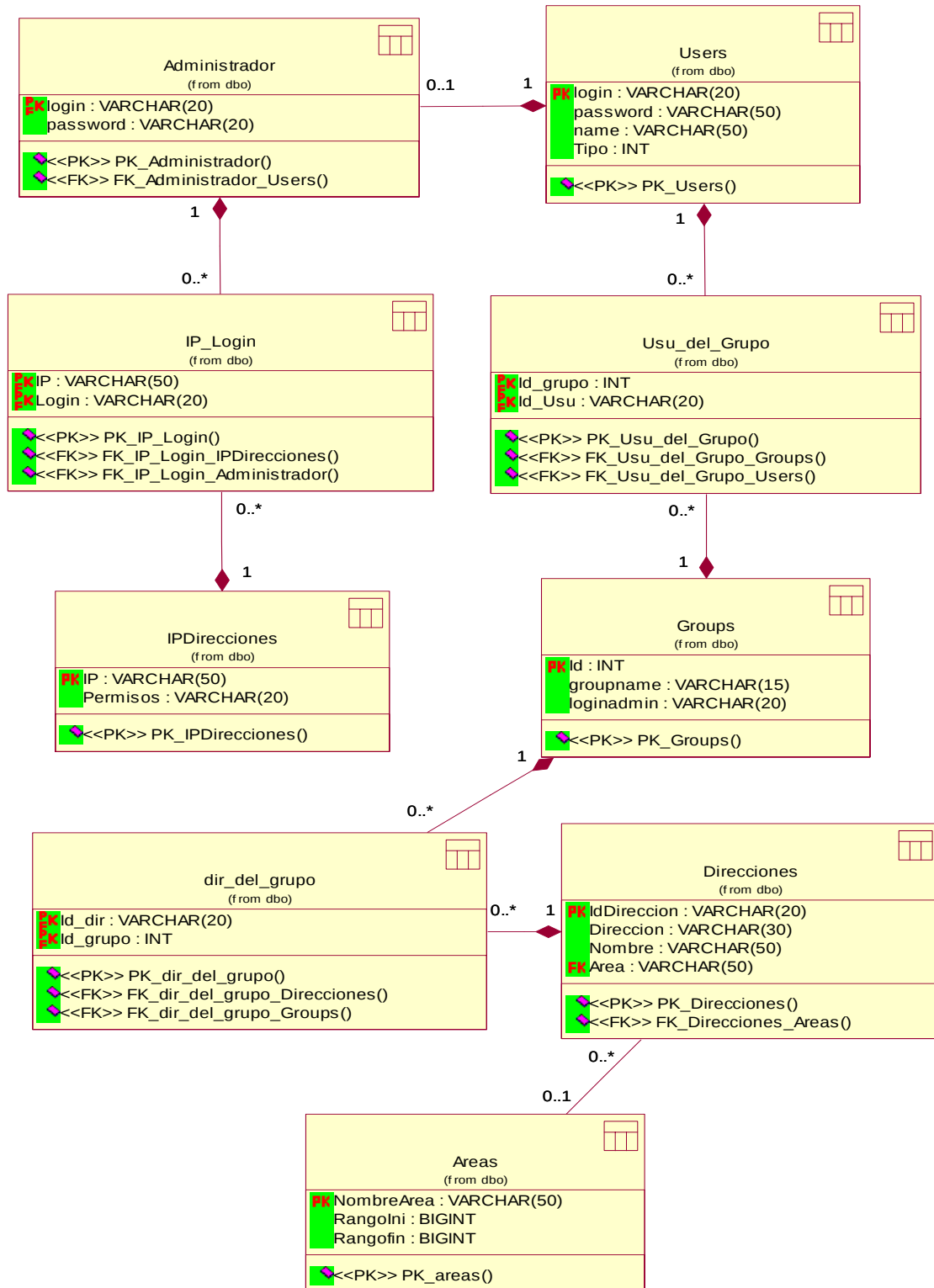


Figura 3.19 Modelo físico de datos

3.5 Diagrama de implementación.

Con el diagrama de implementación se describen los componentes y la organización de acuerdo a los nodos, así como las dependencias entre ellos, por lo que en una sola vista se pueden apreciar los Diagramas de Componente y de Despliegue. Aunque se pueden construir por separado estos diagramas, en ellos no queda clara la relación directa que puede darse entre componentes ubicados en nodos diferentes.

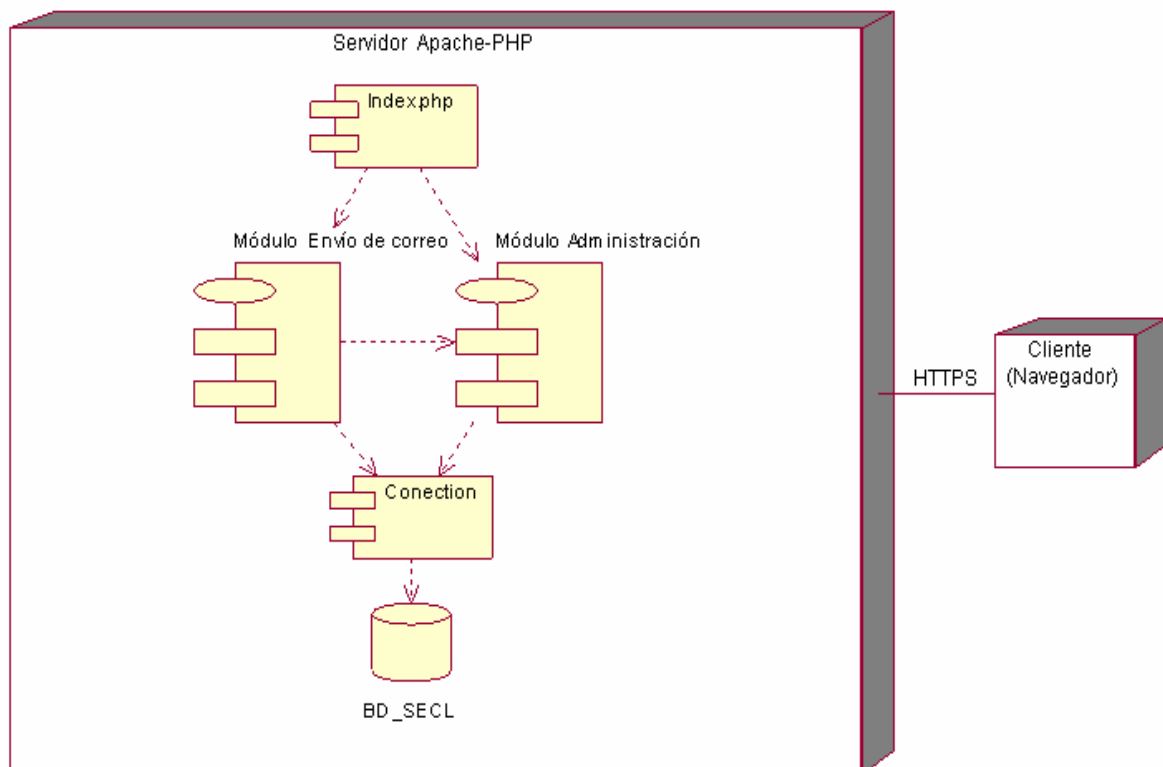


Figura 3.20 Diagrama de implementación

3.6 Conclusiones.

Como resultado en este capítulo se elaboraron los diagramas de clases web. Se describieron los principios del diseño seguidos en el sistema propuesto, profundizando específicamente en los temas de estándares de la interfaz, concepción general de la ayuda y el tratamiento de excepciones. También se realizó el diseño de la base de datos a partir del modelo lógico y físico; y se describieron los elementos fundamentales de la implementación a través del diagrama de implementación.

Conclusiones.

En el presente trabajo se ha puesto de manifiesto la importancia de contar con métodos o sistemas que permitan la difusión de información de forma simultánea, fundamentalmente el envío de correos electrónicos a grupos de usuarios, demostrándose la necesidad de contar con un sistema que favorezca la utilización de este servicio y cumpla los requisitos indispensables de fiabilidad, seguridad y eficiencia deseados.

Teniendo en cuenta los objetivos planteados, se arriba a las siguientes conclusiones:

1. Como resultado del trabajo realizado se logró el diseño de un sistema que permite el control y envío de correo electrónico a grupos de usuarios en la universidad de Cienfuegos. El mismo facilita la administración descentralizada del servicio y le incorpora mecanismos de supervisión y control encaminados a garantizar un uso adecuado del sistema.
2. Para lograr la definición del diseño propuesto se realizó un estudio de los antecedentes y estado actual de los sistemas empleados para el envío de correos por lote, determinando los principales problemas asociados al empleo de alias y se analizaron las tendencias y tecnologías actuales del campo de la Informática que posibilitaron diseñar y desarrollar el sistema propuesto.
3. Teniendo en cuenta el diseño propuesto se realizaron las adecuaciones pertinentes relacionadas con las características propias del sistema de correo empleado en la Universidad de Cienfuegos, definiéndose las herramientas y aplicaciones a emplear en la implementación final.
4. Se diseñó una base de datos para el almacenamiento de la información, siendo seleccionado como sistema gestor MySQL, garantizando con el mismo los niveles requeridos de fiabilidad, velocidad, protección y seguridad deseados en el procesamiento de la información.
5. Como parte del trabajo realizado se desarrolló e implementó un Sistema de Envío de Correos por Lote (SECL) que solucionó las deficiencias

asociadas al empleo de alias de correo, técnica utilizada anteriormente para enviar informaciones a grupos de usuarios en la red telemática de la universidad de Cienfuegos, concluyéndose todos los flujos de trabajo especificados en la metodología RUP y haciendo uso del lenguaje de modelado UML para la construcción de los diagramas que propone dicha metodología.

Recomendaciones.

A pesar del avance alcanzado en esta investigación, que concluyó con el desarrollo del Sistema de Envío de Correos por Lote (SECL) y el notable cambio en la administración y control de este servicio en la red de la universidad de Cienfuegos, existen una serie de recomendaciones en aras de mejorar los niveles de prestaciones y posibilitar la implementación del mismo en redes con características diferentes.

Como primeros pasos que den continuidad a este trabajo proponemos:

1. Desarrollar las adecuaciones pertinentes que posibiliten la obtención de forma automática de los datos necesarios a partir de sistemas operativos diferentes.
2. Coordinar los cambios necesarios para la vinculación del sistema con la base de datos del Sistema Integral de Configuración y Control (SICC) desarrollado por el departamento de telemática del ISPJAE y empleado en la red de la universidad de Cienfuegos.
3. Explotar al máximo las posibilidades que brinda el SECL para la administración y control del envío de correos por lote, permitiendo probar el sistema durante un período de tiempo significativo que garantice comprobar de forma práctica sus funcionalidades y obtener los datos necesarios para su perfeccionamiento.
4. Que con la consecución de las tres primeras acciones se pueda obtener una versión mejorada que permita su implementación en redes con características diferentes.

Referencias Bibliográficas.

- [1] Tanenbaum, Andrew S.. *Redes de Computadoras*, 3ra ed. / Andrew S. Tanenbaum. —México: Prentice- Hall Hispanoamericana, S.A., 1997. — 813p.
- [2] E. Comer, Douglas. *Redes Globales de Información con Internet y TCP/IP*, 3ra ed. / Douglas E. Comer.—México: Prentice- Hall Hispanoamericana, S.A., 1996. —621p.
- [3] García Tomas, Jesús; Alcalde Lancharro, Eduardo. *Introducción a la Teleinformática* / Jesús García Tomas; Eduardo Alcalde Lancharro.—[s.l] Interamericana: McGraw-Hill, 1995. —305p.
- [4] *Standard X.400 ISO*. Tomado de: <http://www.protocols.com/pbook/iso.htm> 23 de abril del 2006.
- [5] *Exchange Server 2003 Technical Documentation Library*. Tomado de: <http://www.microsoft.com/technet/prodtechnol/exchange/2003/library/default.mspx> 23 de abril del 2006.
- [6] *Mail Server MDaemon*. Tomado de: <http://www.imagine.com.co/conectividad/mdaemon.php> 23 de abril del 2006.
- [7] *Sendmail Consortium*. Tomado de: <http://www.sendmail.org> 23 de abril del 2006.
- [8] Dent, Kyle D. *Postfix: The Definitive Guide* / Kyle D. Dent.—[s.l] O'Reilly. December 2003. —264p.
- [9] *Microsoft Outlook Express 6.0*. Tomado de: <http://support.microsoft.com/default.asp> 23 de abril del 2006.
- [10] *Mozilla Thunderbird*. Tomado de: <http://www.mozilla.com/thunderbird/> 22 de abril del 2006.
- [11] *SquirrelMail*. Tomado de: www.squirrelmail.org 23 de abril del 2006.
- [12] *Open WebMail Project*. Tomado de: <http://openwebmail.org/> 22 de abril del 2006.
- [13] Stallings, William. *Comunicaciones y redes de computadores*, 5ª ed. / William Stallings .—Madrid: Prentice may Iberia, 1997 .—808p.

- [14] *RFC 2821 Simple Mail Transfer Protocol*. Tomado de: <ftp://ftp.rfc-editor.org/in-notes/rfc2821.txt> 24 de abril del 2006.
- [15] *RFC 2822 Internet Message Format*. Tomado de: <ftp://ftp.rfc-editor.org/in-notes/rfc2822.txt> 24 de abril del 2006.
- [16] *RFC 1521 MIME (Multipurpose Internet Mail Extensions)*. Tomado de: <http://www.ietf.org/rfc/rfc1521.txt> 24 de abril del 2006.
- [17] *RFC 1939 Post Office Protocol*. Tomado de: <http://www.ietf.org/rfc/rfc1939.txt> 24 de abril del 2006.
- [18] *RFC 2060 Internet Message Access Protocol*. Tomado de: <http://www.ietf.org/rfc/rfc2060.txt> 24 de abril del 2006.
- [19] *OpenSSL Project*. Tomado de: <http://www.openssl.org/> 24 de abril del 2006.
- [20] *Securing POP3 over SSL*. Tomado de: <http://www.linuxdocs.org/sln/pop3s/> 24 de abril del 2006.
- [21] *RFC 2595 Using TLS with IMAP, POP3 and ACAP, IMAPS*. Tomado de: <http://www.ietf.org/rfc/rfc25295.txt> 24 de abril del 2006.
- [22] *HTTPS, RFC 2818 HTTP Over TLS*. Tomado de: <http://www.ietf.org/rfc/rfc2818.txt> 24 de abril del 2006.
- [23] *LISTSERV® Email List Management Software*. Tomado de: <http://www.lsoft.com/products/listserv.asp> 24 de abril del 2006.
- [24] *Mayordomo*. Tomado de: <http://www.greatcircle.com/majordomo/> 24 de abril del 2006.
- [25] *ListProc Open Source*. Tomado de: <http://sourceforge.net/projects/listproc/> 24 de abril del 2006.
- [26] *Webmin*. Tomado de: <http://www.webmin.com> 24 de abril del 2006.
- [27] *Email Marketing Software*. Tomado de: <http://www.email-masivo.com> 24 de abril del 2006.

- [28] *Correos Masivos*. Tomado de: <http://espanol.softpicks.net/software> 24 de abril del 2006.
- [29] Larman, Craig. *UML y Patrones* / Craig Larman.-- La Habana: Editorial Félix Varela, 2004.--Tomo I y II.
- [30] Jacobson, Ivar. *El Proceso Unificado de Desarrollo de Software* / Ivar Jacobson; Grady Booch; Jame Rumbaugh.-- La Habana: Editorial Félix Varela, 2004.-- Tomo I y II.
- [31] *Rational Rose*. Tomado de: <http://www.rational.com/> 24 de abril del 2006.
- [32] December, John. *HTML & CGI Unleashed*. Professional Reference Edition / John December; Mark Ginsburg. --[s.l]. 1996.
- [33] Rodríguez, Daniel; Bravo, Joaquín. *Tutorial de HTML*. Tomado de: <http://html.programacion.net>. 15 de marzo del 2006.
- [34] Musciano, Check; Kennedy, Bill. *HTML: The Definitive Guide*. Tomado de: <http://www.oreilly.com/catalog/html3/> 20 de marzo del 2006.
- [35] *Introducción a PHP*. Tomado de: <http://www.ciberteca.net/webmaster/php>. 20 de marzo del 2006.
- [36] Munz, Stefan. *Tecnologías Web/ ASP*. Tomado de: <http://es.selfhtml.org/introduccion/tecnologias/asp.htm> 20 de marzo del 2006.
- [37] Garrett, David. *Intranets Unleashed*. Tomado de: <http://my.linkbaton.com/bibliography/> 20 de marzo del 2006.
- [38] Wille, Christoph. *Unlocking Active Server Pages*. Tomado de: <http://my.linkbaton.com/bibliography/> 20 de marzo del 2006.
- [39] Smith, David E. *Knowledge, Groupware and the Internet*. Tomado de: <http://my.linkbaton.com/bibliography/> 15 de marzo del 2006.
- [40] Flanagan, David. *Javascript: The Definitive Guide*. Fourth Edition / David Flanagan. --November 2001. 936p.

- [41] Goodman, Danny. *Programación en Javascript*. Tomado de: <http://www.iua.upf.es/~npares/docencia/mprog/> 20 de marzo del 2006.
- [42] Urman, Scout. *Oracle 8, Programación PL/SQL*. -- Editorial McGraw-Hill. 1998. 816p.
- [43] *Manual de referencia de MySQL*. Tomado de: <http://www.mysql.com>. 24 de abril del 2006.
- [44] *Tutorial de MySQL*. Tomado de: <http://www.aplushosting.com/spanish/tutorials> 24 de abril del 2006.
- [45] *SQL*. Tomado de: <http://www.arsys.es/soporte/programacion>. 24 de abril del 2006.
- [46] *Manual SQL Server*. Tomado de: <http://walter.freesevers.com/contsql.html> 24 de abril del 2006.

Bibliografía

- Arques, Pilar. *Tecnologías Web*. Tomado de:
<http://www.dccia.ua.es/dccia/inf/asignaturas/TW/teoria.htm> 15 de marzo del 2006.
- E. Comer, Douglas. *Redes Globales de Información con Internet y TCP/IP*, 3ra ed. / Douglas E. Comer.—México: Prentice- Hall Hispanoamericana, S.A., 1996. —621p.
- García Tomas, Jesús; Alcalde Lancharro, Eduardo. *Introducción a la Teleinformática* / Jesús García Tomas; Eduardo Alcalde Lancharro.—[s.l] Interamericana: McGraw-Hill, 1995. —305p.
- Halsall, Fred. *Data Communications, Computer Networks and Open Systems*, 4^a ed. / Fred Halsall .—Addison-Wesley Publishing Company Inc, 1996 .— 907p.
- Jacobson, Ivar. *El Proceso Unificado de Desarrollo de Software* / Ivar Jacobson; Grady Booch; Jame Rumbaugh.-- La Habana: Editorial Félix Varela, 2004.--Tomo I y II.
- Kruchten, P. *The Rational Unified Process: An Introduction*. Tomado de: EBSCO 25 de febrero del 2006.
- Larman, Craig. *UML y Patrones* / Craig Larman.-- La Habana: Editorial Félix Varela, 2004.--Tomo I y II.
- Letelier Torres, Patricio. *Desarrollo de Software Orientado a Objeto usando UML*. Tomado de: <http://www.creangel.com/uml/intro.php> 10 de abril del 2006.
- Marco Giménez, Luis. *Introducción a las tecnologías Web*. Tomado de: http://iier.isciii.es/cur_web/cur_web_intro/pdf/Introducci%C3%B3n_tecnolog%C3%ADasweb.pdf 15 de marzo del 2006.
- Martínez Echevarría, Álvaro. *Manual práctico de HTML*. Tomado de: <http://www.etsit.upm.es/~alvaro/manual/manual.html> 15 de marzo del 2006.

-
- Munz, Stefan. *Tecnologías Web/ ASP*. Tomado de:
<http://es.selfhtml.org/introduccion/tecnologias/asp.htm> 20 de marzo del 2006.
 - Pressman, Roger. *Software Engineering. A Practitioner's Approach* / Roger Pressman.--(E.U): McGraw – Hill, 1999.--[S.P].
 - Schmuller, J. *APRENDIENDO UML EN 24 HORAS*. UML / J. Schmuller.— México:[S:E], 2000.—[S.P].
 - Rodríguez, Daniel; Bravo, Joaquín. *Tutorial de HTML*. Tomado de:
<http://html.programacion.net>. 15 de marzo del 2006.
 - Ross, Keith W. *Computer networking: a top-down approach featuring the internet* / Keith W. Ross, James F. Kurose .-- Addison-Wesley Longman Inc, 2001 .—712p.
 - Rumbaugh, J.; Blaha, M.; Premerlani, W.; Eddy, F.; Lorensen, W. *Object-oriented modeling and design*. Tomado de: EBSCO 25 de febrero del 2006.
 - Stallings, William. *Comunicaciones y redes de computadores*, 5ª ed. / William Stallings .—Madrid: Prentice may Iberia, 1997 .—808p.
 - Tanenbaum, Andrew S.. *Redes de Computadoras*, 3ra ed. / Andrew S. Tanenbaum. —México: Prentice- Hall Hispanoamericana, S.A., 1997. — 813p.
 - Vidgen, Richard. *Requirements analysis and uml: use cases and class diagrams*. Tomado de: EBSCO 25 de febrero del 2006.

Glosario de Términos.

A

ARTEFACTO: es un término general para cualquier tipo de información creada, producida, cambiada o utilizada por los trabajadores en el desarrollo del sistema.

ASCII: American Standard Code for Information Interchange (Código Estadounidense Estándar para el Intercambio de Información) es un código de caracteres basado en el alfabeto latino tal como se usa en inglés moderno y otras lenguas occidentales.

ASP: (Active Server Pages) Páginas de Servidor Activas.- Tecnología creada por Microsoft, con el fin de que el usuario de Internet pueda recibir páginas generadas dinámicamente en el "Servidor".

C

CCITT: International Telegraph and Telephone Consultative Committee. Comité de Consultas de Telégrafo y Teléfono Internacional. Organización responsable de hacer recomendaciones técnicas sobre sistemas de comunicaciones. Es parte de ITU (International Telecommunications Union).

D

Dirección IP: Las direcciones IP son el método mediante el cual se identifican los ordenadores individuales (o, en una interpretación más estricta, las interfaces de red de dichos ordenadores) dentro de un red TCP/IP. Todas las direcciones IP consisten en cuatro números separados por puntos, donde cada número está entre 0 y 255.

E

Ethernet: Estándar para las LAN. Los ordenadores se conectan a un único canal, y acceden a el mediante el procedimiento CSMA/CD (Acceso al Medio por detección de portadora con detección de colisiones).

F

Frame Relay: Retransmisión de tramas, norma de red LAN/WAN definida por el ANSI y CCITT para tramas de conmutación en un modo de paquete similar a X.25, pero a velocidades mayores y con menos procesamiento de nodos.

H

HTTP: HyperText Transfer Protocol (Protocolo de Transferencia de Hipertexto) Protocolo usado para la transferencia de documentos WWW.

I

IBM: Acrónimo de International Business Machine Se trata de la empresa norteamericana de hardware más importante a escala mundial; creadora del 'PC' (Personal Computer),

IMAP: Interactive Mail Access Protocol: Protocolo Interactivo para el Acceso de Correos. Supone una mejora de POP3. La principal ventaja es que IMAP permite, no sólo la manipulación de los mensajes en nuestro buzón, sino también la gestión de un conjunto de buzones, distribuidas como carpetas de información.

IMAPS: Se refiere al protocolo IMAP cuando se utiliza en combinación con SSL.

IP: Internet Protocol (Protocolo de Internet). Uno de los protocolos más representativos del estándar TCP/IP, es el responsable del esquema de direccionamiento utilizado en Internet y define la estructura y el intercambio de los datagramas IP entre redes distantes, definido en el RFC 791.

ISO: International Organization for Standardization (Organización Internacional para la Estandarización). Agrupación que crea muchos de los valores estándar internacionales. Incluye la ANSI de los EEUU.

ISP: Internet Service Provider (Proveedor de Servicios Internet) Organización, que además de dar acceso a Internet a personas físicas y/o jurídicas, les ofrece una serie de servicios (por ejemplo, hospedaje de páginas web, consultoría de diseño e implantación de webs e intranets).

L

LAN: Local Area Network (Red de área local). Red de datos para dar servicio a un área geográfica máxima de unos pocos kilómetros cuadrados, por lo cual pueden optimizarse los protocolos de señal de la red para llegar a velocidades de transmisión de Gbps (gigabits por segundo).

M

MD5: En criptografía, MD5 (acrónimo de Message-Digest Algorithm 5, Algoritmo de Resumen del Mensaje 5) es un algoritmo de reducción criptográfico de 128 bits ampliamente usado. El código MD5 fue diseñado por Ronald Rivest en 1991.

MySQL: My SQL es una de las bases de datos más populares desarrolladas bajo la filosofía de código abierto. Servidor de Bases de datos existente en plataforma Linux, recomendable para desarrollos que necesiten manejar numerosos registros y sesiones simultáneas.

MX: Mail Exchange (Intercambiador de Correo).

O

Open Source: En la actualidad Open Source es utilizado para definir un movimiento nuevo de software, diferente al movimiento del Software Libre, aunque no completamente incompatible con este, de modo que es posible (como de hecho ocurre) que ambos movimientos trabajen juntos en el desarrollo práctico de proyectos.

OSI: Open Systems Interconnection (Interconexión de Sistemas Abiertos). Convenio de protocolos, diseñados por los comités ISO, para ser el estándar internacional de la arquitectura de redes. Modelo de Referencia OSI. Describe las arquitecturas de redes y la forma en que los datos pasan a través de ellas.

P

PHP: PHP (acrónimo recursivo de "PHP: Hypertext Preprocessor", originado inicialmente del nombre PHP Tools, o Personal Home Page Tools) es un lenguaje de programación interpretado. Se utiliza entre otras cosas para la programación de páginas Web activas, y se destaca por su capacidad de mezclarse con el código HTML.

POP3: POP3 (Post Office Protocol 3). Tercera versión del protocolo diseñado para la gestión, el acceso y la transferencia de mensajes de correo electrónico entre dos máquinas, un servidor y una máquina de usuario.

POP3S: Se refiere al protocolo POP3 cuando se utiliza en combinación con SSL.

R

RFC: Acrónimo inglés de Request For Comments "Petición de comentarios". Conjunto de archivos de carácter técnico donde se describen los estándares o recomendaciones. Resultado y proceso de creación de un estándar en Internet. Los nuevos estándares se proponen y publican en Internet como RFC. El grupo de trabajo de ingeniería de Internet (IETF) es un cuerpo de opinión que admite discusión a través de comentarios, en los que se establece un nuevo estándar.

RSYNC: Es una poderosa implementación de un pequeño y eficiente algoritmo. Su principal poder es la habilidad de replicar eficientemente un sistema de archivos. Usando rsync, es fácil configurar un sistema que mantendrá una copia actualizada de un sistema de archivos.

S

Sniffers: Programa que sorbe datos de la red. Todo lo que le pasa por delante lo absorbe y lo almacena para su análisis posterior. De esta forma, sin poseer acceso a ningún sistema de la red, se puede obtener información, claves de acceso o incluso mensajes de correo electrónico en el que se envían estas claves.

SNMP: Simple Network Management Protocol (Protocolo Simple de Gestión de Red) Protocolo estándar de Internet definido en STD15, RFC 1157, desarrollado para manejar nodos sobre una red IP.

SMTP: Simple Mail Transfer Protocol (SMTP), o protocolo simple de transferencia de correo electrónico. Protocolo de red basado en texto utilizado para el intercambio de mensajes de correo electrónico entre computadoras y/o distintos dispositivos (PDA's, Celulares, etc).

SQL: Structured Query Language (Lenguaje estructurado de consulta). Lenguaje de programación que se utiliza para recuperar y actualizar la información contenida en una base de datos. Fue desarrollado en los años 70 por IBM. Se ha convertido en un estándar ISO y ANSI.

SUM: Sedes Universitarias Municipales.

T

TCP: Transmission Control Protocol (Protocolo de Control de Transmisión). Protocolo de nivel de transporte estándar de la internet definido en STD 7, RFC 793.

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

U

UDP: User Datagram Protocol. Protocolo de nivel de transporte estándar de la Internet definido en STD 6, RFC 768. Añade flexibilidad al IP al no estar orientado a la conexión, se manda el paquete y no se espera confirmación.

UTP: Par Trenzado No apantallado.(LAN).

V

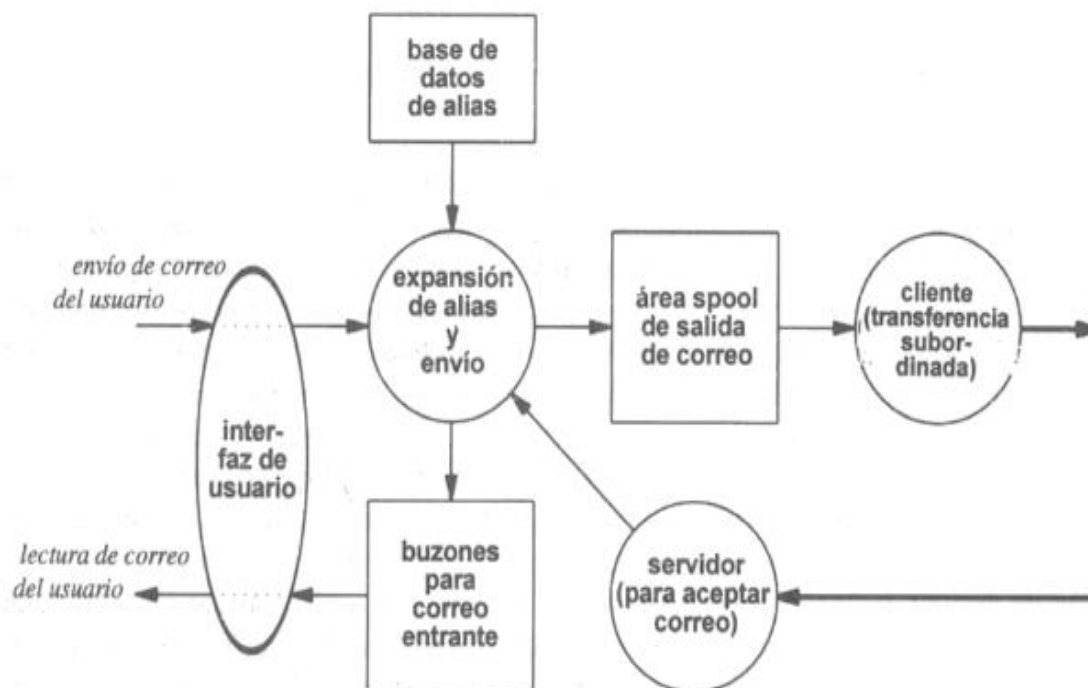
VLAN: Grupo de dispositivos de una LAN que están configurados (usando el software de administración) de tal modo que se pueden comunicar como si estuvieran conectados al mismo cable, cuando, en realidad, están ubicados en una serie de segmentos de LAN distintos. Debido a que las LAN virtuales están basadas en conexiones lógicas en lugar de físicas, son extremadamente flexibles.

W

WAN: Wide Area Network (Red de área amplia). Red, normalmente constituida por líneas serie, que cubre una gran área geográfica.

Anexos

Anexo 1. Sistema de correo electrónico que soporta alias de correo y listas de expansión.



Anexo 2. Prototipos.

Prototipo 2.1 Gestionar usuarios.

SECL _____
sistema de envío de correo por lote

administrador de área (admin)	
>>gestión de usuarios >>gestión de grupos >>gestión de direcciones >>cambiar contraseña de admin >>cambiar contraseña de correo >>buscar informaciones >>enviar correos por lotes >>ayuda del sistema >>cerrar sesión	gestión de usuarios visualizar usuarios crear usuarios eliminar usuarios

SECL _____
sistema de envío de correo por lote

administrador de área (admin)	
>>gestión de usuarios >>gestión de grupos >>gestión de direcciones >>cambiar contraseña de admin >>cambiar contraseña de correo >>buscar informaciones >>enviar correos por lotes >>ayuda del sistema >>cerrar sesión	crear usuario Elija el Usuario Elige usuario ▾ Selecciona nombre Contraseña resetear crear necesita llena todos los campos!

SECL _____
sistema de envío de correo por lote

administrador de área (admin)										
>>gestión de usuarios >>gestión de grupos >>gestión de direcciones >>cambiar contraseña de admin >>cambiar contraseña de correo >>buscar informaciones >>enviar correos por lotes >>ayuda del sistema >>cerrar sesión	eliminar usuario <table border="1"> <thead> <tr> <th>login</th> <th>nombre</th> <th>tipo</th> </tr> </thead> <tbody> <tr> <td>☐ vidal</td> <td>vidal edwards</td> <td>1</td> </tr> <tr> <td>☐ aantelo</td> <td>Aurelio Antelo Collado</td> <td>2</td> </tr> </tbody> </table> seleccionar todos deshacer selección eliminar	login	nombre	tipo	☐ vidal	vidal edwards	1	☐ aantelo	Aurelio Antelo Collado	2
login	nombre	tipo								
☐ vidal	vidal edwards	1								
☐ aantelo	Aurelio Antelo Collado	2								

Prototipo 2.2 Actualizar privilegios.

SECL _____
sistema de envío de correo por lote

administrador general (root:root)
cambiar privilegios

>> gestión de áreas

>> gestión de usuarios

>> gestión de grupos

>> gestión de direcciones

>> configurar acceso desde PC's

>> cambiar contraseña de root

>> cambiar contraseña decorreo

>> buscar informaciones

>> enviar correos por lotes

ayuda del sistema

cerrar sesión

Id del administrador	—Selecciona usuario—
Tipo del administrador	Selecciona tipo (0:root 1:admon)
resetear insertar	

Prototipo 2.3 Configurar acceso desde IP.

SECL _____
sistema de envío de correo por lote

administrador general (root:root)

>> gestión de áreas

>> gestión de usuarios

>> gestión de grupos

>> gestión de direcciones

>> configurar acceso desde PC's

>> cambiar contraseña de root

>> cambiar contraseña decorreo

>> buscar informaciones

>> enviar correos por lotes

ayuda del sistema

cerrar sesión

Id del administrador	—Selecciona administrador—	
Ip de la maquina		
resetear insertar		

Permiso	Nombre del admon	Ip del maquina
☐ todos	Alexis Gómez	132.90.78.09
☐ todos	Alexis Gómez	234.90.45.06
☐ todos	root	12.13.15.17
☐ todos	root	10.14.13.5
☐ todos	root	100.111.50.67

opción de los ip existentes

☒ permitir todos
☐ aceptar
☐ denegar

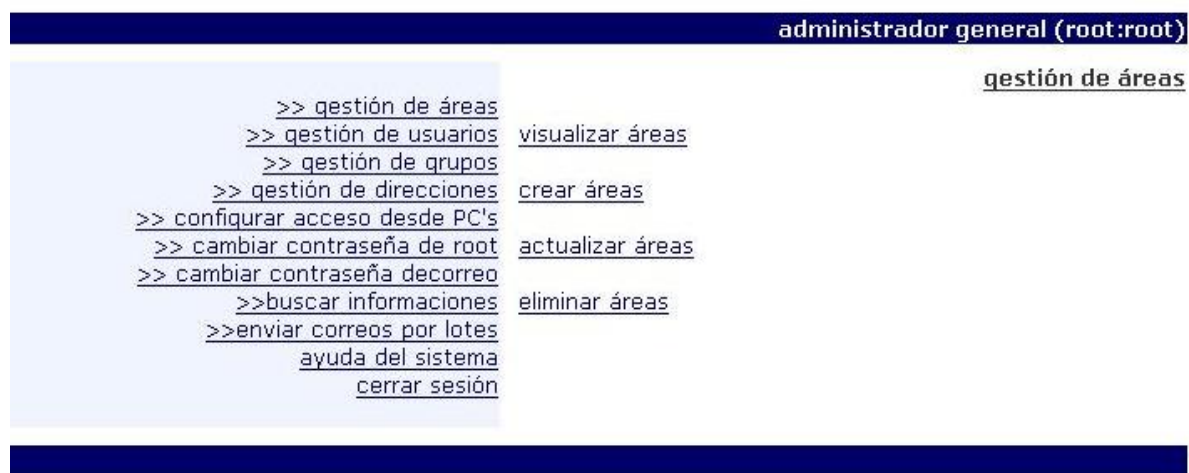
Prototipo 2.4 Importar direcciones de correo.

SECL _____
sistema de envío de correo por lote



Prototipo 2.5 Gestionar áreas.

SECL _____
sistema de envío de correo por lote



SECL _____

sistema de envío de correo por lote

administrador general (root:root)

[crear área](#)

[>> gestión de áreas](#)

[>> gestión de usuarios](#)

[>> gestión de grupos](#)

[>> gestión de direcciones](#)

[>> configurar acceso desde PC's](#)

[>> cambiar contraseña de root](#)

[>> cambiar contraseña decorreo](#)

[>> buscar informaciones](#)

[>> enviar correos por lotes](#)

[ayuda del sistema](#)

[cerrar sesión](#)

Insertar Nombre del área	
Insertar Valor inicial del área	
Insertar Valor final del área	
resetear insertar	

SECL _____

sistema de envío de correo por lote

administrador general (root:root)

[eliminar área](#)

[>> gestión de áreas](#)

[>> gestión de usuarios](#)

[>> gestión de grupos](#)

[>> gestión de direcciones](#)

[>> configurar acceso desde PC's](#)

[>> cambiar contraseña de root](#)

[>> cambiar contraseña decorreo](#)

[>> buscar informaciones](#)

[>> enviar correos por lotes](#)

[ayuda del sistema](#)

[cerrar sesión](#)

Nombre del area	Valor inicial	Valor final
☐ FINF	2000	5999
☐ FMEC	6000	7999
☐ FCEE	8000	9999
☐ FSCH	10000	11999
☐ Rectorado	12000	12999

SECL _____

sistema de envío de correo por lote

administrador general (root:root)

[actualizar área](#)

[>> gestión de áreas](#)

[>> gestión de usuarios](#)

[>> gestión de grupos](#)

[>> gestión de direcciones](#)

[>> configurar acceso desde PC's](#)

[>> cambiar contraseña de root](#)

[>> cambiar contraseña decorreo](#)

[>> buscar informaciones](#)

[>> enviar correos por lotes](#)

[ayuda del sistema](#)

[cerrar sesión](#)

Escoger nombre del area	-Selecciona Area-
nuevo valor inicial del área	
nuevo valor final del área	
resetear actualiza	

Prototipo 2.6 Gestionar grupos de direcciones.

SECL _____
sistema de envío de correo por lote

administrador de área (admin)	
>>gestión de usuarios >>gestión de grupos >>gestión de direcciones >>cambiar contraseña de admin >>cambiar contraseña de correo >>buscar informaciones >>enviar correos por lotes >>ayuda del sistema >>cerrar sesión	gestión de grupos visualizar grupos crear grupos eliminar grupos asociar grupos a usuarios desvincular grupos de los usuarios

SECL _____
sistema de envío de correo por lote

administrador de área (admin)											
>>gestión de usuarios >>gestión de grupos >>gestión de direcciones >>cambiar contraseña de admin >>cambiar contraseña de correo >>buscar informaciones >>enviar correos por lotes >>ayuda del sistema >>cerrar sesión	crear o eliminar grupos Nombre del grupo Resetear Insertar <table border="1"> <thead> <tr> <th>id del grupo</th> <th>nombre del grupo</th> </tr> </thead> <tbody> <tr> <td>☐ 75</td> <td>Facultad INF</td> </tr> <tr> <td>☐ 76</td> <td>Facultad MEC</td> </tr> <tr> <td>☐ 77</td> <td>Profesores FINF</td> </tr> <tr> <td>☐ 78</td> <td>Estudiantes FIN</td> </tr> </tbody> </table> seleccionar todos deshacer selección eliminar	id del grupo	nombre del grupo	☐ 75	Facultad INF	☐ 76	Facultad MEC	☐ 77	Profesores FINF	☐ 78	Estudiantes FIN
id del grupo	nombre del grupo										
☐ 75	Facultad INF										
☐ 76	Facultad MEC										
☐ 77	Profesores FINF										
☐ 78	Estudiantes FIN										

SECL _____

sistema de envío de correo por lote

[>>gestión de usuarios](#)

[>>gestión de grupos](#)

[>>gestión de direcciones](#)

[>>cambiar contraseña de admin](#)

[>>cambiar contraseña de correo](#)

[>>buscar informaciones](#)

[>>enviar correos por lotes](#)

[>>ayuda del sistema](#)

[>>cerrar sesión](#)

administrador de área (admin)

desvincular grupos de usuarios

-Selecciona Usuario-

▼

Id grupo	Nombre del grupo	Id Usuario
☐ 76	Facultad MEC	vidal
☐ 77	Profesores FINF	vidal
☐ 78	Estudiantes FIN	vidal

Prototipo 2.8 Gestionar direcciones de un grupo.

SECL _____

sistema de envío de correo por lote

[>>gestión de usuarios](#)

[>>gestión de grupos](#)

[>>gestión de direcciones](#)

[>>cambiar contraseña de admin](#)

[>>cambiar contraseña de correo](#)

[>>buscar informaciones](#)

[>>enviar correos por lotes](#)

[>>ayuda del sistema](#)

[>>cerrar sesión](#)

administrador de área (admin)

gestión de direcciones de correo

[visualizar direcciones](#)

[asociar direcciones a los grupos](#)

[desvincular direcciones de los grupos](#)

[desvincular grupos de las direcciones](#)

Prototipo 2.9 Buscar Información.

SECL _____
sistema de envío de correo por lote

administrador general (root:root)														
<u>buscar</u>														
>> gestión de áreas >> gestión de usuarios >> gestión de grupos >> gestión de direcciones >> configurar acceso desde PC's >> cambiar contraseña de root >> cambiar contraseña de correo >> buscar informaciones >> enviar correos por lotes ayuda del sistema cerrar sesión	<table border="1"> <thead> <tr> <th>qué buscas</th> <th>criterio 1</th> <th>criterio 2</th> </tr> </thead> <tbody> <tr> <td>-Escoger Una-</td> <td>-Escoger Una-</td> <td></td> </tr> <tr> <td></td> <td></td> <td></td> </tr> <tr> <td colspan="3" style="text-align: center;"> Buscar </td> </tr> </tbody> </table>	qué buscas	criterio 1	criterio 2	-Escoger Una-	-Escoger Una-					Buscar			
qué buscas	criterio 1	criterio 2												
-Escoger Una-	-Escoger Una-													
Buscar														

Prototipo 2.10 Cambiar contraseña de administradores de área.

SECL _____
sistema de envío de correo por lote

administrador de área (admin)											
<u>cambiar contraseña</u>											
>> gestión de usuarios >> gestión de grupos >> gestión de direcciones >> cambiar contraseña de admin >> cambiar contraseña de correo >> buscar informaciones >> enviar correos por lotes >> ayuda del sistema >> cerrar sesión	<table border="1"> <tbody> <tr> <td>Escoger el id del usuario</td> <td> Selecciona usuario </td> </tr> <tr> <td>Contraseña vieja</td> <td> </td> </tr> <tr> <td>Contraseña nueva</td> <td> </td> </tr> <tr> <td>Confirmar Contraseña</td> <td> </td> </tr> <tr> <td colspan="2" style="text-align: center;"> resetear actualizar </td> </tr> </tbody> </table>	Escoger el id del usuario	Selecciona usuario	Contraseña vieja		Contraseña nueva		Confirmar Contraseña		resetear actualizar	
Escoger el id del usuario	Selecciona usuario										
Contraseña vieja											
Contraseña nueva											
Confirmar Contraseña											
resetear actualizar											

Prototipo 2.11 Cambiar contraseña de usuarios.

Similar al prototipo del caso de uso Cambiar contraseña de administradores de área.

Prototipo 2.12 Visualizar direcciones de un grupo.

SECL _____
sistema de envío de correo por lote

[visualizar direcciones](#)

dirección	grupo	nombre del grupo
☐ adrian@ucf.edu.cu	75	Facultad INF
☐ alex@ucf.edu.cu	75	Facultad INF
☐ alexey@ucf.edu.cu	75	Facultad INF
☐ alexismacia@ucf.edu.cu	75	Facultad INF
☐ alina@ucf.edu.cu	75	Facultad INF

Prototipo 2.13 Enviar correo a grupos de direcciones.

SECL _____
sistema de envío de correos por lote

[cerrar sesión](#)
[cambiar contraseña](#)
[ver direcciones de grupos](#)

para:

asunto:

Prototipo 2.14 Mostrar ayuda.

SECL _____
sistema de envío de correo por lote

Ayuda del sistema	
>> ayuda para la gestión de áreas >> ayuda para la gestión de usuarios >> ayuda para la gestión de grupos >> ayuda para la gestión de direcciones >> ayuda para la configurar acceso desde PC's >>ayuda para enviar correos	Página de ayuda del SECL, seleccione la categoría que usted desee para ver la información relacionada.

Prototipo 2.15 Autenticar.

SECL _____
sistema de envío de correos por lote

login:	
contraseña:	
	limpiar aceptar

Prototipo 2.16 Cerrar sesión.

desconectarse
se ha desconectado correctamente
pulse aquí para volver a ingresar

Prototipo 2.17 Cambiar contraseña del administrador general.

SECL _____
sistema de envío de correo por lote

administrador general (root:root)	
<u>cambiar contraseña de root</u>	
>> gestión de áreas	usuario root
>> gestión de usuarios	Contraseña nueva
>> gestión de grupos	Confirmar Contraseña
>> gestión de direcciones	resetear actualizar
>> configurar acceso desde PC's	
>> cambiar contraseña de root	
>> cambiar contraseña decorreo	
>> buscar informaciones	
>> enviar correos por lotes	
ayuda del sistema	
cerrar sesión	