

Universidad de Cienfuegos “Carlos Rafael Rodríguez”

Facultad de Ingeniería

Carrera de Ingeniería Informática

**Sistema informático para gestionar las salvas de correo electrónico de la Refinería de
Petróleo “Camilo Cienfuegos”.**

Autor:

Yuniesky Labrada Moreno

Tutor:

Ing. Jorge Luís Rivero Pérez. Universidad de Cienfuegos

Consultantes:

MSc. Alexis Gómez Domínguez PDVCupetSA “Refinería de Petróleo de Cienfuegos”.

E-mail:

inf200739@ucf.edu.cu

jlrivero@ucf.edu.cu

adominguez@refcfg.cuvenpetrol.cu

Cienfuegos, Cuba

Curso 2011-2012

Declaración de autoría

Declaro que soy el único autor de este trabajo de diploma titulado Sistema informático para gestionar las salvas de correo electrónico de la Refinería de Petróleo “Camilo Cienfuegos” y autorizo a La Refinería de Petróleo y al Departamento de Informática de la Facultad de Informática en la Universidad de Cienfuegos “Carlos Rafael Rodríguez”, para que hagan el uso que estimen pertinente con el trabajo de diploma.

Para que así conste firmamos la presente a los ____ días del mes de ____ de 2012.

Autor: Yuniesky Labrada Moreno.

Tutor: Ing. Jorge Luís Rivero Pérez.

Agradecimientos

Ante todo a mi familia porque sin su apoyo no lo habría logrado.

A mis amistades de escuela, al Harry (Yulieny), al Prieto (Lachi) y al Celso (Alain), etc.

A mi tutor Jorge Luis por ayudarme en todo lo que me hizo falta.

A mi primo Liesberkaya que siempre me apoyo en mis problemas.

A las niñas de 4to informática por su ayuda en la revisión de este documento:

Yaili, Yaneysi, Diana Rosa.

A todos aquellos profesores que de una forma u otra han tenido que ver en mi formación durante toda mi vida de estudiante.

A Dan Google por ayudarme en las respuestas de mis preguntas.

A todo aquel que siempre pensó que yo sí podía.

Dedicatoria

A mi papa que siempre me dio su mejor respuesta ante cualquier problema que se me presentara.

A mi mama que siempre me apoyo en todos los percances que me surgieron.

A mi hermano por ser la chinchia más repugnante de mi familia.

A toda mi familia.

A todas aquellas personas que quiero y que admiro.

Por ultimo y no por ser menos importante, a aquellas personas que ya no están a mi lado pero si jugaron un papel importante en mi vida.

Resumen

La gestión de las salvas de correo electrónico de La Refinería de Petróleo “Camilo Cienfuegos” es una tarea con un alto grado de complicación, unas veces por la gran cantidad de correos electrónicos entrantes y salientes y otra por el chequeo de las diferentes normas de seguridad que puedan existir en los correos electrónicos, ya que los mismos presentan en determinadas ocasiones mucha información. Dichos procesos se realizan de forma manual trayendo consigo la probabilidad de la existencia de amenazas para la red telemática de la Empresa, es por ello que la idea de contar con un sistema informático capaz de detectar anomalías en las salvas de correo electrónicos resulta necesaria para obtener una mayor seguridad en la red de la Empresa.

Índice

Capítulo I.- “Fundamentación teórica”:	6
1.1- Descripción del dominio del problema:	6
1.1.1- Servidor web. [1]:	6
1.1.2- Servidor de correo electrónico web.	6
1.1.3- Cliente de correo electrónico [4]	7
1.1.4- Correo electrónico [7]:	7
1.1.5- Ficheros o salvas de correo electrónico	8
1.2 Descripción del objeto de estudio:	8
1.2.1- Flujo actual de los procesos y análisis critico de la ejecución de estos.	8
1.2.2- Sistemas automatizados existentes al campo de acción:	9
1.2.3- SquirrelMail [9]	9
1.2.4- Grep:	9
1.2.5- Egrep	10
1.3- Tendencias, metodologías y/o tecnologías actuales	14
1.3.1- Tecnologías Web:	15
1.3.2- Fundamentación del lenguaje, gestor de bases de datos y de otros Softwares utilizados.	15
1.4 - Selección del Framework	17
1.4.1- CodeIgniter	18
1.4.2- ExtJs:	18
1.5- Fundamentación de la metodología utilizada:	19
1.5.1- Lenguaje de Modelación Unificado (UML)	19
1.5.2- Proceso Unificado de Desarrollo (RUP)	20
1.7- Conclusiones:	21
Capítulo II.- “Descripción y construcción de la solución propuesta”:	22

2.1- Definición de las entidades y los conceptos principales.	22
Reglas del negocio a considerar.....	22
2.2- Representación del modelo del dominio	23
2.3- Descripción del sistema propuesto	23
2.3.1- Concepción general del sistema.....	23
2.3.2- Requerimientos funcionales.....	24
2.3.3- Requerimientos no funcionales	25
2.4- Modelo de casos de uso del sistema.....	27
2.4.1 Actores del modelo del sistema.....	27
2.4.2- Casos de uso del sistema	28
2.4.3- Diagramas de casos de uso del sistema	29
2.4.4- Diagramas de funcionamiento del framework CodeIgniter	30
2.4.5- Descripción de los casos de uso del sistema	31
2.5- Diagrama de clases del diseño	39
2.6- Diseño de la base de datos.....	39
2.6.1- Modelo lógico de datos.....	39
2.6.2- Modelo físico de datos.....	41
2.6.3- Diagrama de implementación.....	42
2.6.4- Principios de diseño	43
2.6.5- Estándares en la interfaz de la aplicación	43
2.6.6- Tratamiento de errores	43
2.6.7- Concepción General de la ayuda.....	43
2.7- Conclusiones	44
Capítulo III.- “Análisis de prueba y estudio de factibilidad”:	45
3.1- Experimentos y Pruebas realizadas	45
3.2- Pruebas Realizadas	45

3.2.1- Buscar patrones alarmantes en los adjuntos de los correos electrónicos.	45
3.2.2- Buscar palabras claves en los correos electrónicos.	45
3.2.3- Mostrar correos electrónicos comprendidos en fechas.	46
3.2.4- Buscar alarmas en las salvas de correo electrónico.	46
3.2.5- Buscar correos electrónicos por su destino.	46
3.3- Estudio de Factibilidad	46
3.3.2- Planificación por casos de usos	47
3.3.3- Cálculo de Puntos de Casos de Uso sin Ajustar	47
3.3.4 - Cálculo de los Puntos de Casos de Uso Ajustados	48
3.3.5- Estimación del esfuerzo	50
3.3.6- Cálculo de costos	51
3.3.7- Beneficios tangibles e intangibles	51
3.3.8- Análisis de costos y beneficios	51
3.4- Conclusiones	51
Conclusiones	52
Recomendaciones	53
Referencias bibliográficas	54
Bibliografía	57
Glosario de términos	59
Anexos	62

Índice de Tablas

Tabla 1 Actores del sistema	28
Tabla 2 Descripción del caso de uso de sistema: Autenticarse.....	31
Tabla 3 Descripción del caso de uso de sistema: Gestionar url.....	32
Tabla 4 Descripción del caso de uso de sistema: Gestionar Usuario.....	33
Tabla 5 Descripción del caso de uso de sistema: Gestionar Configuración	34
Tabla 6 Descripción del caso de uso de sistema: Gestionar Norma de Seguridad	35
Tabla 7 Descripción del caso de uso de sistema: Mostrar Alarmas.....	36
Tabla 8 Descripción del caso de uso de sistema: Datos del correo electrónico.....	37
Tabla 9 Descripción del caso de uso de sistema: Buscar.....	38
Tabla 10 Diagrama de clases de diseño	39
Tabla 11 Asignación de peso a los actores.	47
Tabla 12 Factor de peso de los casos de uso sin ajustar.	48
Tabla 13 Cálculo del factor de complejidad técnica.	49
Tabla 14 Cálculo del factor ambiente.	49
Tabla 15 Criterios de distribución de esfuerzos.....	50

Índice de Figuras

Figura 1 Sistema Propuesto	12
Figura 2 Representación del modelo de dominio	23
Figura 3 Diagrama de casos de uso del sistema.....	29
Figura 4 Diagrama de funcionamiento del CodeIgniter.	30
Figura 5 Diagrama de clases persistentes para la base datos de salvas de correo.	40
Figura 6 Diagrama de clases persistentes para la base datos de usuarios	41
Figura 7 Diagrama del modelo físico de datos para la base datos de salvas de correo.	41
Figura 8 Diagrama del modelo físico de datos para la base datos de usuarios del sistema	42
Figura 9 Diagrama de implementación	42

Introducción

Con el surgimiento de internet la productividad entre las computadoras ha aumentado, trayendo consigo amenazas informáticas en las redes de las empresas. La Refinería de Petróleo es una de estas empresas que está puesta a la existencia de diferentes anomalías (amenazas informáticas en los correos electrónicos) en su servicio de correo electrónico, por esta razón la gestión de las salvas de los mismos resulta una herramienta de gran importancia a la hora de detectar alguna amenaza para la red, posibilitando un control más estricto sobre los correos electrónicos entrantes y salientes de dicha Empresa.

La gestión de las salvas de correo electrónico es una de las tareas más complicadas, consumiendo mucho tiempo y no realizándose con el tiempo requerido, de aquí que no se haga, ya que resulta ser muy tediosa. Así mismo lograr detectar de forma manual alguna anomalía en una salva de correo electrónico es muy dificultoso debido a la cantidad de información que en la misma se aloja, es por ello que la inspección de las mismas se hace necesaria.

Situación del Problema

La Refinería de Petróleo “Camilo Cienfuegos” presenta un servicio de correo electrónico el cual está configurado para que diariamente genere salvas de correo electrónico con la información tanto de los correos que entran como de los que salen de dicha Empresa, estas salvas de correo electrónico son almacenadas en diferentes carpetas de dos servidores centrales. Ante tal situación la gestión de las salvas de correo electrónico se realiza de forma manual, haciéndose una tarea complicada y que requiere de mucho tiempo, por esta razón no se realiza con la periodicidad necesaria.

Problema

Ante esta situación antes descrita resulta de gran importancia para cualquier empresa contar con un sistema informático que gestione la información contenida en las salvas de correo electrónico de las diferentes carpetas del servidor central; queda definido como problema, la ausencia de un Sistema informático para la gestión de las salvas de correo electrónico de la red telemática de La Refinería de Petróleo “Camilo Cienfuegos”.

Por estas razones nos planteamos las siguientes interrogantes:

1. ¿Qué herramientas de software utilizar para centralizar las salvas de correo electrónico?
2. ¿Cuáles son las diferentes tecnologías que pueden utilizarse para gestionar las salvas de correo electrónico generadas por el servicio de correo electrónico de La Refinería de Petróleo “Camilo Cienfuegos”?
3. ¿Qué herramienta de Software desarrollar para realizar la gestión de las salvas de correos electrónicos?

Atendiendo a tal problema se decidió desarrollar un sistema informático para la gestión de las salvas de correo electrónico, enfocado a la eliminación de las deficiencias actuales, y proyectado a elevar los niveles de seguridad y confiabilidad de la red, garantizando minimizar las incomodidades del procesamiento del presente problema.

Objeto de estudio y campo de acción

La presente investigación es desarrollada en la red telemática de La Refinería de Petróleo “Camilo Cienfuegos” y se considera como objeto de estudio de la misma la gestión de las salvas de correo electrónico de La Refinería de Petróleo “Camilo Cienfuegos”.

Idea a defender

Se planteó como ***idea a defender*** la aplicación de un sistema informático capaz de gestionar las salvas de correo electrónico para la red telemática de La Refinería de Petróleo, permitiendo facilitar las tareas de supervisión y control y garantizando la detención de cualquiera anomalía que exista en las mismas.

Justificación de la investigación

Debido a que al tener centralizada la información de las salvas de correo electrónico, la presente investigación detecta y gestiona la ocurrencia de cualquier anomalía que en estas pueda existir, aumentando los niveles de seguridad, monitoreo y búsqueda de las mismas, logrando saber en qué correo electrónico se aloja dicha deficiencia.

Objetivo general

Desarrollar un sistema informático para la gestión de las salvas de correo electrónico para la red telemática de La Refinería de Petróleo “Camilo Cienfuegos”.

Objetivos específicos

- Analizar el proceso de salvas de correo electrónico de La Refinería de Petróleo.
- Diseñar un sistema informático que gestione las salvas de correo electrónico de La Refinería de Petróleo.
- Implementar el sistema informático propuesto mediante una aplicación web.
- Validar el sistema.

Metodología de trabajo

Para cumplir con los objetivos trazados se realizaron un conjunto de *tareas*:

- Realización de entrevistas a los trabajadores que participan en el proceso.
- Realización de pruebas prácticas del sistema.
- Validación de la implementación final mediante el empleo de herramientas de Software y el análisis del comportamiento del sistema.

El presente documento va recorriendo las etapas de este proyecto, describiendo cada una de ellas. Para una mejor comprensión, ha sido estructurado en capítulos, que en su conjunto proporcionan una idea completa y acabada del proceso de desarrollo.

Capítulo I.- “Fundamentación teórica”

El capítulo aborda los puntos teóricos de la investigación realizada, juntos a los conceptos vinculados al dominio del problema. Se exponen los conceptos asociados al objeto de estudio de la investigación y las tendencias y metodologías existentes para el desarrollo de aplicaciones informáticas.

Capítulo II.- “Descripción y construcción de la solución propuesta”

En este capítulo se describen los procesos del negocio empleando los artefactos de UML, los diagramas y casos de uso del sistema, así como los requisitos funcionales o no funcionales. Se describe el diseño de la solución propuesta.

Se describe de forma general como funciona el sistema. Se especifican los requerimientos funcionales y no funcionales. Se realiza la descripción del modelo de casos de uso del sistema, incluyendo los actores y casos de uso. Se describe el diseño y la implementación del sistema, a partir del diagrama de clases del diseño, el diseño de la base de datos y el diagrama de implementación. Se analizan los principios de diseño.

Se muestra una descripción detallada de la solución propuesta del sistema, utilizando la representación gráfica de los diagramas de clases del modelo de sistema y el diagrama de implementación. Además se describió el estándar que va a tener el sistema, el tipo de letra a utilizar, se describió como serán los mensajes de error y el sistema de ayuda a utilizar.

Capítulo III.- “Análisis de prueba y estudio de factibilidad”

En este capítulo se describen los resultados obtenidos al someter a prueba al sistema, así como sus respectivas validaciones. Además se detalla el estudio de factibilidad económica del sistema mediante el análisis de puntos de casos de uso, teniendo en cuenta la estimación del esfuerzo, el cálculo de costos, los beneficios tangibles e intangibles y el análisis de costos y beneficios.

Capítulo I.- “Fundamentación teórica”:

En el presente capítulo se definen los principales conceptos asociados al dominio del problema y se describen los antecedentes relacionados con el objeto de estudio y campo de acción.

Se analizan a continuación las diferentes tendencias, metodologías, tecnologías y herramientas existentes, determinando en qué medida contribuyen a la solución del problema, permitiendo la selección de las adecuadas para el análisis, diseño e implementación del sistema.

1.1- Descripción del dominio del problema

1.1.1- Servidor web.[1]

Un servidor web o servidor HTTP es un programa informático que procesa una aplicación del lado del servidor realizando conexiones bidireccionales y/o unidireccionales y síncronas o asíncronas con el cliente generando o cediendo una respuesta en cualquier lenguaje o aplicación del lado del cliente. El código recibido por el cliente suele ser compilado y ejecutado por un navegador web. Para la transmisión de todos estos datos suele utilizarse algún protocolo, generalmente se utiliza el protocolo HTTP para estas comunicaciones.[2]

1.1.2- Servidor de correo electrónico web.

Es aquél que es accedido vía Web usando el protocolo http. En realidad no es servidor, sino un cliente de correo que corre en un servidor web. A través de dicho cliente se puede acceder al servidor de correo sin necesidad de instalar un cliente de correo en la computadora local.

En este tipo de servidor, el archivo de datos del remitente o destinatario puede ser accedido sin requerir un cliente local. En el mismo servidor se integran programas para acceder a los correos del mismo.[3]

1.1.3- Cliente de correo electrónico[4]

Es un programa de ordenador usado para leer y enviar mensajes de correo electrónico. Los mismos fueron pensados para ser programas simples permitiendo leer los mensajes del correo electrónico de usuarios localizados en un buzón local.

La información de estos correos electrónicos localizados en el buzón local presentan diferentes formatos, entre los más importantes están el mbox y Maildir, los cuales son protocolos que se utilizan para el almacenamiento local de los mensajes, posibilitando así de forma más sencilla la importación, exportación y copia de seguridad de las carpetas de correo electrónico.[5]

Mbox es un término genérico para una familia de formatos de fichero que se usan para almacenar conjuntos de correos electrónicos. Todos los mensajes en un buzón mailbox están concatenados en un único fichero. El principio de cada mensaje está marcado por una vacía que empieza por los cinco caracteres "From ", y una línea en blanco marca el final.[6]

Maildir es un formato de spool de correo electrónico que no bloquea los ficheros para mantener la integridad del mensaje, porque los mensajes se almacenan en ficheros distintos con nombres únicos. Maildir es un directorio con tres subdirectorios llamados tmp, new, y cur. Todos los subdirectorios deben residir en el mismo sistema de archivos.

1.1.4-Correo electrónico[7]

Es un servicio de red que permite a los usuarios enviar y recibir mensajes y archivos rápidamente mediante sistemas de comunicación electrónicos. Principalmente se usa este nombre para denominar al sistema que provee este servicio en Internet, mediante el protocolo SMTP, aunque por extensión también puede verse aplicado a sistemas análogos que usen otras tecnologías. Por medio de mensajes de correo electrónico se puede enviar, no solamente texto, sino todo tipo de documentos digitales.[8]

1.1.5- Ficheros o salvas de correo electrónico

Los archivos o fichero informáticos son un conjunto de bits almacenados en un dispositivo. El archivo es identificado por un nombre y la descripción de la carpeta o directorio que lo contiene. Los archivos informáticos facilitan una manera de organizar los recursos usados para almacenar permanentemente datos en un sistema informático, de esta manera se puede tener almacenada la información de diferentes correos electrónicos durante un tiempo indefinido.

1.2 Descripción del objeto de estudio

1.2.1- Flujo actual de los procesos y análisis crítico de la ejecución de estos.

La Refinería de Petróleo cuenta con dos servidores encargados del control de los correos electrónicos de entrada y salida. Cada servidor guarda la información de dichos correos electrónicos en diferentes ficheros los cuales se le suele llamar “salvas de correo electrónico”, los mismos se encuentran alojados en dos carpetas diferentes (Mx1, Mx2).

Cada fichero contiene la información de los correos electrónicos que entran y salen de La Refinería de Petróleo. Esta información se puede encontrar de varias formas (codificada y explícita), en el primer caso la información codificada es aquella que hace referencia a los adjuntos de los correos mientras que el segundo caso hace referencia a la parte del cuerpo del correo electrónico, ya sea el texto que se escribe y otros datos.

La centralización brinda beneficios desde el punto de vista de la seguridad informática y facilidades en el proceso de gestión ya que la información se concentra en un único lugar. En los momentos actuales la existencia de un sistema automatizado que realice dichas tareas se ha convertido en una gran idea, ya que la realización de los procesos sería más rápida y más segura. Actualmente esta gestión se realiza de forma manual, trayendo consigo el riesgo de no poder detectar alguna anomalía que exista en las salvas de correo electrónico.

1.2.2- Sistemas automatizados existentes al campo de acción.

Existen una gran variedad de sistemas informáticos desarrollados para realizar la captura de los diferentes correos electrónicos generados por los servidores de correo. A continuación se abordan varios de estos sistemas atendiendo a sus características.

1.2.3- SquirrelMail[9]

Es una aplicación webmail creada por Nathan y Luke Ehresman y escrita en PHP. Es un sistema libre, bajo la licencia de GNU, sigue el standard HTML 4.0, está diseñado para trabajar con plugins, lo cual hace más llevadera la tarea de agregar nuevas características entorno al núcleo de la aplicación. Squirrelmail es software libre. Viene con varios plugins por defecto, permitiendo a los administradores añadir:[10]

- Revisar ortografía.
- Filtrar emails. spam.
- Administración Web de SquirrelMail.
- Un calendario.
- Una interfaz para reportar errores semiautomáticamente.

Ha sido traducido en más de 40 lenguajes (Árabe, Chino, Francés, Alemán y Español).

Conclusión del SquirrelMail

Este software a pesar de conectarse con un servidor de correo electrónico y de leer el contenido de las diferentes carpetas que allí se alojan, no es capaz de gestionar salvas de correo electrónico en busca de algún error que pueda existir, por tanto queda descartada de ser una solución ya que no cumple con las exigencias de La Refinería de Petróleo.

1.2.4- Grep

Es una utilidad de la línea de comandos escrita originalmente para ser usada con el sistema operativo Unix. El comando grep (Global Regular Expression and Print) permite buscar las líneas que contienen una cadena de caracteres especificada mediante una expresión regular. Lee la entrada estándar o una lista de archivos y muestra en la salida sólo aquellas líneas que contienen la expresión indicada. [11]La sintaxis es

```
$ grep[opciones][expresión regular][archivo]
```

Sus opciones más interesantes son:

-c

Modifica la salida normal del programa, en lugar de imprimir por salida estándar las líneas coincidentes, imprime la cantidad de líneas que coincidieron en cada archivo.

-e PATRÓN

Usar PATRÓN como el patrón de búsqueda, muy útil para proteger aquellos patrones de búsqueda que comienzan con el signo «-».

-f ARCHIVO

Obtiene los patrones del archivo ARCHIVO.

-H

Imprimir el nombre del archivo con cada coincidencia.

-r

Buscar recursivamente dentro de todos los subdirectorios del directorio actual.

Conclusiones de Grep

Esta solución está descartada debido a que es una aplicación totalmente en consola y no cumple con las necesidades de La Refinería de Petróleo. Además la misma no presenta funciones para gestionar salvas de correo electrónico, solamente realiza búsquedas en un fichero utilizando expresiones regulares como patrón de búsqueda.

1.2.5- Egrep

Es una versión de grep que optimiza la búsqueda en casos particulares, aceptando expresiones regulares más complejas y extendidas con los parámetros + ? | (), esto hace que la búsqueda en ocasiones sea más lenta. La misma imprime todas las ocurrencias encontradas en el fichero analizado.[12]

La sintaxis de la función es:

```
egrep expresión_regular archivo
```

Escribir `grep -E` es similar a `egrep`, aunque no idéntico; `egrep` es compatible con el comando histórico `egrep`; `grep -E` acepta expresiones regulares extendidas y es la versión moderna del comando en GNU.

Conclusiones de Egrep

Esta propuesta queda descartada ya que no cumple con los requerimientos de La Refinería de Petróleo porque no tiene implementada funciones para gestionar salvas de correo electrónico, la misma es una mejora de `grep`, convirtiéndose en otra herramienta que realiza las mismas funciones pero que puede recibir otros tipos de expresiones regulares.

El sistema que se obtiene con la presente investigación brinda opciones de gestión de las salvas de correo electrónico, siendo esta la función principal a considerar, además brinda al encargado de la Seguridad Informática la opción de la detención de anomalías de forma automática posibilitando esto la seguridad y confiabilidad del sistema informático, de esta forma quedan satisfechas las necesidades de los administradores de la red de La Refinería de Petróleo.

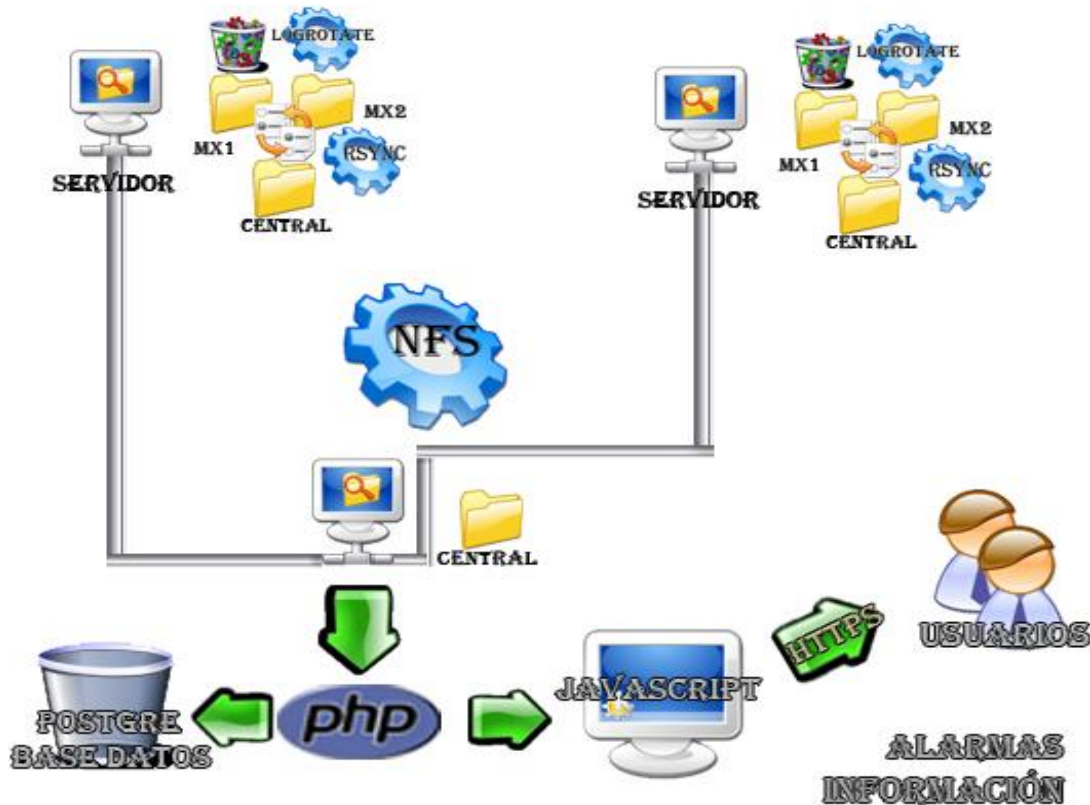


Figura 1 Sistema Propuesto

En La Refinería de Petróleo “Camilo Cienfuegos” existen dos servidores de correos electrónicos, los cuales están configurados para que diariamente realicen salvas de los correos electrónicos entrantes y salientes de La Refinería de Petróleo, esta configuración se logra utilizando una herramienta llamada LogRotate.

Este servidor de correo electrónico realiza dichas salvas en carpetas diferentes de dicho servidor, por lo que se hace necesario utilizar una herramienta que las centralice, de aquí que se tome como herramienta a RSYCN[13], la cual nos permite centralizar las salvas de correo electrónico en una carpeta del servidor.[14]

Para la comunicación del servidor central con los servidores de correo electrónico se utilizó un sistema llamado NFS la cual permite compartir carpetas en la red, posibilitando así que otras máquinas se puedan conectar a la misma.

Estas salvas de correo electrónico que están contenidas en la carpeta del servidor central se le aplican funciones de php, lográndose guardar diferentes datos obtenidos con las mismas en una base de datos, y con ayuda del sistema propuesto se logra gestionar las salvas de correo electrónico, mostrando así los datos de las mismas y posibilitando que el encargado de la seguridad informática se sienta seguro y cómodo.

LogRotate ha sido diseñado para facilitar la administración de sistemas que generan gran cantidad de archivos de registro (logs). El programa permite el cambio, compresión, eliminación y envío automático por correo de los archivos de registro. Cada archivo puede ser tratado diariamente, semanalmente, mensualmente o siempre que se haga demasiado grande. [15]

Normalmente, logrotate se ejecuta como tarea diaria de tipo cron. El programa no modificará un archivo repetidas veces a no ser que el criterio para ese archivo se base en el tamaño del mismo y logrotate se ejecute varias veces al día.

En la línea de comandos pueden indicarse un número ilimitado de archivos de configuración. Más tarde, estos archivos de configuración pueden invalidar las opciones dadas en archivos de configuración anteriores, por lo tanto, el orden en que se listan los archivos de configuración de logrotate es muy importante. Normalmente, debería usarse un sólo archivo de configuración que incluya tantos otros archivos de configuración como sean necesarios. Véase más abajo cómo usar la directiva *include* para conseguirlo. Si se proporciona un nombre de directorio en la línea de comandos, cada uno de los archivos que se encuentren en ese directorio se usará como archivo de configuración.

NFS es el sistema que utiliza Linux para compartir carpetas en una red. Mediante NFS, un servidor puede compartir sus carpetas en la red. Desde los PCs de los usuarios se puede acceder a dichas carpetas compartidas y el resultado es el mismo que si estuvieran en su propio disco duro. NFS son las siglas en inglés de Network File System que podríamos traducir como Sistema de Archivos en Red.[16]

Básicamente **NFS** permite, a PCs que utilizan Linux, compartir y conectarse a carpetas compartidas entre sí. Es el sistema nativo que utiliza Linux para compartir y acceder a carpetas compartidas en la red.

RSYNC es una aplicación libre para sistema de tipo Unix y Microsoft Windows que ofrece transmisión eficiente de datos incrementales, que opera también con datos comprimidos y cifrados. Permite sincronizar archivos y directorios entre dos máquinas de una red o entre dos ubicaciones en una misma máquina, minimizando el volumen de datos transferidos. Escucha por defecto por el puerto TCP 873, sirviendo archivos en el protocolo nativo rsync o vía terminal remoto como RSH o SSH. Se distribuye bajo la licencia GNU General Public License. [13]

Optimiza las transferencias entre dos computadoras sobre TCP/IP, incluye compresión y descompresión de los datos bloque por bloque, utilizando zlib al enviar y recibir, lo que permite la transmisión cifrada y eficientemente diferenciada de datos comprimidos usando el algoritmo rsync.[14]

SSH (Secure SHell) es un protocolo que sirve para acceder a máquinas remotas a través de una red. Permite manejar por completo la computadora mediante un intérprete de comandos, además de copiar datos de forma segura, gestionar claves RSA y pasar los datos de cualquier otra aplicación por un canal seguro tunelizado mediante SSH.

1.3- Tendencias, metodologías y/o tecnologías actuales

La calidad en el desarrollo de una herramienta de software depende de la metodología a seguir, para ello se hace necesario el estudio de las tecnologías actuales, conocidas o no, con el fin de escoger la más ajustada a nuestro problema. A continuación se hace referencia a las características fundamentales de la metodología escogida para el desarrollo del software, en el Proceso Unificado de Desarrollo.

1.3.1- Tecnologías Web

Tecnologías Cliente

Navegador Web

Mozilla Firefox.

Google Chrome.

Opera.

Tecnologías de programación

HTML.

Hojas de estilo en cascada (CSS).

JavaScript.

Tecnologías Servidor

Servidor Web

Apache.

Servidor de Base de Datos

PostgreSQL 9.0.0.1

Tecnologías de Programación

PHP.

1.3.2- Fundamentación del lenguaje, gestor de bases de datos y de otros Softwares utilizados.

HTML

El sistema mostrará diferentes interfaces web en las cuales se posibilitará las funcionalidades de gestión y control de las salvas de correo electrónico. De aquí que se tome como lenguaje de programación el HTML, siendo este un lenguaje de marcación diseñado para estructurar textos y presentarlos en forma de hipertexto, brindando herramientas para reproducir sonidos, mostrar imágenes, aglutinar textos y hacer referencias a otras páginas mediante enlaces de hipertexto. [17]

CSS

El sistema está realizado completamente con el framework ExtJs, el mismo dentro de sus bibliotecas principales posee una en la cual se aloja toda la información que utilizan los CSS, el mismo está programado para la realización de diferentes estilos a las interfaces que muestra.

Además de la biblioteca principal del framework se utiliza otro documento en el cual se le da estilos a los diferentes objetos de la pagina (botones, fotos, etc). El lenguaje analizado posibilita controlar los estilos de diferentes elementos en la web, además de darle formato a cada uno por individual utilizando un solo bloque de código situado en un fichero individual, esto posibilita la realización de cualquier cambio sin necesidad de afectar la web principal. [18]

PHP

Este lenguaje de programación por su gran característica de poder ser desplegado por la mayoría de los servidores web y en casi todos los sistemas operativos, es escogido para la implementación de las diferentes funcionalidades del sistema informático, el mismo cuenta con diferentes funcionalidades empaquetadas en bibliotecas que posibilitan la conexión de base datos y el trabajo con ellas mediante funciones adecuadas, además posee características como: velocidad, estabilidad, seguridad y simplicidad que son pruebas de sus beneficios.

Las páginas web escritas con este código pueden ser interpretadas en sistemas operativos como las diferentes versiones de Linux, Windows y Mac (multiplataforma). [19]

JavaScript

El framework que se utilizó para la realización del sistema está completamente escrito en Javascript, el mismo presenta diferentes funciones que permiten al sistema interactuar con base datos y otros frameworks, dicho lenguaje de programación es compacto y basado en objetos, diseñado para el desarrollo de aplicaciones cliente-servidor a través de Internet. [20]

PostgreSQL

Es un sistema de gestión de bases de datos, el cual presenta la capacidad de almacenar diferente información, permitiendo el guardado de los diferentes datos de los correos electrónicos, entre sus características principales se encuentran:

1. Implementación del estándar SQL92/SQL99.
2. Soporta distintos tipos de datos.
3. Incorpora una estructura de datos array.
4. Incorpora funciones de diversa índole: manejo de fechas, geométricas, orientadas a operaciones con redes, etc.
5. Permite la declaración de funciones propias, así como la definición de disparadores.
6. Soporta el uso de índices, reglas y vistas.
7. Incluye herencia entre tablas (aunque no entre objetos, ya que no existen), por lo que a este gestor de bases de datos se le incluye entre los gestores objeto-relacionales.
8. Permite la gestión de diferentes usuarios, como también los permisos asignados a cada uno de ellos.[21]

1.4 - Selección del Framework

Los frameworks han adquirido un gran valor para los desarrolladores pues simplifica el desarrollo de una aplicación mediante la automatización de algunos de los patrones utilizados para resolver las tareas comunes.

También proporcionan estructura al código fuente, esto provoca que el desarrollador del sistema deba crear código más legible y más fácil de mantener; además por su característica de encapsular operaciones complejas en instrucciones sencillas, el framework, facilita la programación de aplicaciones.

A continuación se presentan algunas de las principales características de varios frameworks con el fin de facilitar su selección.

1.4.1- CodeIgniter

CodeIgniter es un framework que contiene un grupo de herramientas que le facilitan el trabajo a las personas que gustan de construir aplicaciones web usando PHP, permitiéndole desarrollar proyectos con un costo en tiempo mucho menor que si lo escribiese desde cero.

Este framework tiene un juego de librerías para tareas comúnmente necesarias muy útil, así como una interfaz simple y estructura lógica para acceder a esas librerías; permite creativamente enfocarse en un proyecto específico minimizando la cantidad de código necesaria para una tarea dada. Es fácil de instalar y de aprender, características que provocan que sea preferido por aquellas personas que dispongan de poco tiempo para realizar un proyecto; otra de sus ventajas es que facilita la escritura de código repetitivo.[22]

- Sistema basado en Modelo Vista Controlador (MVC).
- Compatible con PHP 4.
- Extremadamente liviano.

1.4.2- ExtJs

Es una librería Javascript que permite construir aplicaciones complejas en Internet. Esta librería incluye:

- Componentes UI del alto performance y personalizables.
- Modelo de componentes extensibles.
- Un API fácil de usar.
- Licencias Open source y comerciales.

Además este framework presenta una ventana flotante que es excelente por la forma en la que funciona. Al moverla o redimensionarla solo se dibujan los bordes haciendo que el movimiento sea fluido lo cual le da una ventaja tremenda frente a otros.[23]

Usar un motor de render como ExtJS nos permite tener además:

- Existe un balance entre Cliente – Servidor. La carga de procesamiento se distribuye, permitiendo que el servidor, al tener menor carga, pueda manejar más clientes al mismo tiempo.
- Comunicación asíncrona. En este tipo de aplicación el motor de render puede comunicarse con el servidor sin necesidad de estar sujeta a un clic o una acción del usuario, dándole la libertad de cargar información sin que el cliente se de cuenta.
- Eficiencia de la red. El tráfico de red puede disminuir al permitir que la aplicación elija que información desea transmitir al servidor y viceversa, sin embargo la aplicación que haga uso de la pre-carga de datos puede que revierta este beneficio por el incremento del tráfico.

La parte negativa de este framework se describe a continuación:

- Necesita una plataforma.
- El Javascript no es tan rápido.
- Descargas lentas.
- Problemas con los motores de búsqueda.
- Accesibilidad. Estas aplicaciones tienen problemas con los programas de accesibilidad pues, al igual que los motores de búsqueda, no trabajan bien con texto cargado dinámicamente.
- No se pueden usar fuera de línea. Por su naturaleza web estas aplicaciones no pueden ser usadas en el cliente como cualquier otra aplicación.

1.5- Fundamentación de la metodología utilizada

1.5.1- Lenguaje de Modelación Unificado (UML)

“El Lenguaje de Modelado Unificado (UML - Unified Modeling Language) permite modelar, construir y documentar los elementos que forman un producto de Software que responde a un enfoque orientado a objetos. 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. Desde entonces, se ha convertido en el estándar internacional para definir, organizar y visualizar los elementos que configuran la arquitectura de una aplicación orientada a objetos.

Con este lenguaje, se pretende unificar las experiencias acumuladas sobre técnicas de modelado e incorporar las mejores prácticas actuales en un acercamiento estándar.[24]

UML no es un lenguaje de programación sino un lenguaje de propósito general para el modelado orientado a objetos y también puede considerarse como un lenguaje de modelado visual que permite una abstracción del sistema y sus componentes.” [25]

Entre sus objetivos fundamentales se encuentran:

1. Ser tan simple como sea posible, pero manteniendo la capacidad de modelar toda la gama de sistemas que se necesita construir.
2. Necesita ser lo suficientemente expresivo para manejar todos los conceptos que se originan en un sistema moderno, tales como la concurrencia y distribución, así como también los mecanismos de la ingeniería de Software, como son el encapsulamiento y el uso de componentes.
3. Debe ser un lenguaje universal, como cualquier lenguaje de propósito general.
4. Imponer un estándar mundial.

1.5.2- Proceso Unificado de Desarrollo (RUP)

El Proceso Unificado de Desarrollo (RUP, por su denominación en inglés Rational Unified Process), fue creado por el mismo grupo de expertos que crearon UML, Ivar Jacobson, Grady Booch y James Rumbaugh en el año 1998. El objetivo que se perseguía con esta metodología era producir Software de alta calidad, es decir, que cumpla con los requerimientos de los usuarios dentro de una planificación y presupuesto establecidos. Esta metodología concibió desde sus inicios el uso de UML como lenguaje de modelado. Es un proceso dirigido por casos de uso, este avanza a través de una serie de flujos de trabajo, está centrado en la arquitectura y es iterativo e incremental. Además cubre el ciclo de vida de desarrollo de un

proyecto y toma en cuenta las mejores prácticas a utilizar en el modelo de desarrollo de Software.[26]

A continuación se muestran estas prácticas.[27]

- Desarrollo de Software en forma iterativa.
- Manejo de requerimientos.
- Utiliza arquitectura basada en componentes.
- Modelación del Software visualmente
- Verifica la calidad del Software.
- Controla los cambios.

1.7- Conclusiones

En este capítulo se expone de forma detallada el riesgo que presenta La Refinería de Petróleo ante alguna amenaza en los correos electrónicos tanto entrantes como salientes, además de realizar un estudio de los frameworks a utilizar en la creación de un sistema informático capaz de gestionar y controlar las salvas de correos electrónicos que allí se realizan diariamente.

Capítulo II.- “Descripción y construcción de la solución propuesta”:

Se da a conocer la captura de los tipos de objetos más importantes que existen o los eventos que suceden en el lugar donde radicará el sistema. Además se considera que una buena solución a tal situación es aquella en donde el factor principal es la gestión y presentación de la información. El modelo de dominio se considera en RUP un subconjunto del llamado modelo de objetos del negocio.

2.1- Definición de las entidades y los conceptos principales.

En el modelo de dominio se definen las siguientes entidades:

Mbox: protocolo de encriptación que se utiliza para el guardado de datos de correo electrónico.

Rsync: Aplicación que se utiliza para la sincronización de carpetas o ficheros tanto de una máquina personal como entrequinas de una red.

Servidor de Correo: Encargado de interactuar con un cliente de correo para conjuntamente tomar los datos de los diferentes correos electrónicos que puedan llegar a una empresa.

Servidor de centralización de salvas: Servidor en el cuál se alojan todas las salvas de correos electrónicos.

Salvas de correo electrónico: Ficheros con el contenido de los correos electrónicos entrantes y salientes de La Refinería de Petróleo.

Encargado de la Seguridad Informática: Es la persona que controla que en las salvas de correo no exista ninguna negligencia, ya que esto atentaría contra la seguridad informática de la Refinería de Petróleo.

Reglas del negocio a considerar.

Los ficheros deben estar encriptados siguiendo el protocolo mbox.

Se debe utilizar la aplicación Rsync para tener todas las salvas de correo electrónicos alojadas en un mismo servidor.

Debe existir un servidor que permita la realización de salvas de correo electrónico.

Debe existir un servidor para garantizar que se pueda centralizar las salvas de correo electrónico.

Deben existir las salvas de correo electrónico para que así el sistema pueda leer su contenido. El Encargado de la Seguridad Informática es la persona que hace las diferentes gestiones en las salvas de correo para así asegurarse de que no hay ninguna negligencia en las mismas.

2.2- Representación del modelo del dominio

Representación gráfica del diagrama.

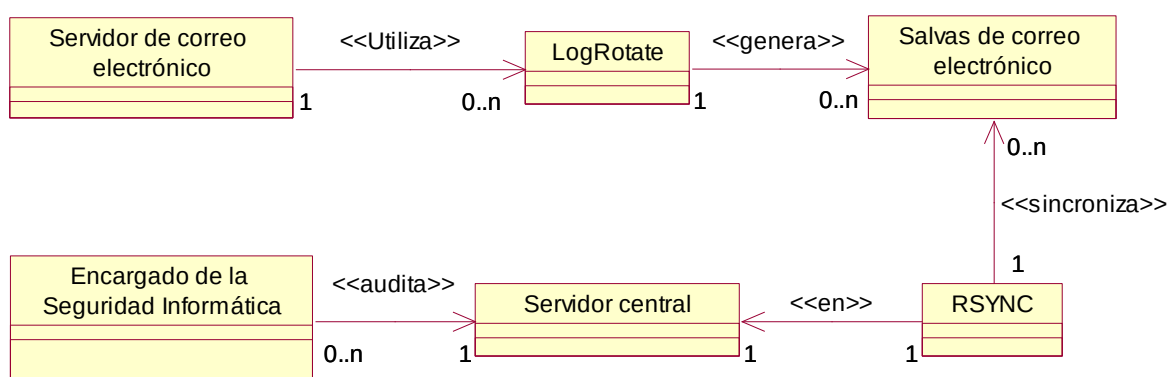


Figura 2 Representación del modelo de dominio

2.3- Descripción del sistema propuesto

El siguiente capítulo mostrara el funcionamiento detallado del sistema y a su vez una descripción de su funcionamiento.

2.3.1- Concepción general del sistema

Por la necesidad de controlar y gestionar las salvas de correo electrónico en La Refinería de Petróleo “Camilo Cienfuegos”, surge la idea de la creación de un sistema informático que permita a parte de estos procesos, llevar un control estricto sobre las mismas, ayudando al encargado de la seguridad informática tomar decisiones ante cualquier amenaza que en las mismas se encuentre.

El sistema ara revisiones diarias de las salvas de correo electrónico, controlando cualquier anomalía que en las mismas pueda existir, y enviara un mensaje de correo electrónico de advertencia a la dirección de correo electrónico la cual presenta alguna anomalía.

2.3.2- Requerimientos funcionales

El sistema debe ser capaz de darle cumplimiento a los diferentes requerimientos funcionales, es por ello que se describen utilizando el modelo de Casos de uso y especifican el comportamiento del sistema en toda su vida de desarrollo, surgiendo de la por la necesidad actual de la existencia del producto.

Para la captura de los diferentes requerimientos funcionales que aquí se manifiestan, se realizaron entrevistas al Encargado de la seguridad Informática, con el objetivo de tener mas conocimiento del tema abordado, y así poder entender de forma más detallada sus diferentes objetivos sobre la realización del sistema. Además se tomaron en cuenta diferentes sugerencias del mismo para la realización de otros casos de uso atendiendo a su punto de vista teórico.

Los requerimientos funcionales del sistema propuesto son los siguientes:

- 1- Autenticarse.
- 2- Insertar url en el sistema.
- 3- Eliminar url en el sistema.
- 4- Modificar url en el sistema.
- 5- Insertar usuario.
- 6- Cambiar contraseña.
- 7- Eliminar usuario
- 8- Insertar Configuración.
- 9- Insertar norma de seguridad.
- 10-Eliminar norma de seguridad.
- 11-Modificar norma de seguridad.
- 12-Mostrar alarmas.
- 13-Mostrar datos del correo señalado.
- 14-Buscar por adjunto.
- 15-Buscar por destinatarios.
- 16-Buscar por envíos.
- 17-Buscar por fechas.

2.3.3- Requerimientos no funcionales

Este aspecto aborda las propiedades o cualidades que el sistema informático debe tener, para que así cumpla con las exigencias de los diferentes usuarios que con el trabajen, el mismo debe ser rápido, atractivo, fácil de usar y confiable, además de tener un buen rendimiento, extensibilidad y fiabilidad.

A continuación se muestran los diferentes requerimientos no funcionales para el sistema propuesto.

- **Apariencia o interfaz externa**

Para el intercambio entre sistema-usuario, el mismo debe presentar una interfaz sencilla, simple y legible, debe ser profesional e interactivo, debe mostrar la información de forma legible, debe presentar la posibilidad de navegación, brindarle al usuario la posibilidad de saber en donde se encuentra en todo momento, para que así el mismo se sienta cómodo y confiado.

- **Usabilidad**

El encargado de la Seguridad Informática tendrá la posibilidad de gestionar y auditar las salvas de correo electrónico ya que el sistema informático está dirigido al mismo, también el sistema informático le ayudará a conocer cualquier error o anomalía que en los mismos se encuentre. Dicho sistema informático está dirigido a personas a las personas con conocimiento de la seguridad informática y el ambiente web.

- **Rendimiento**

La eficacia, eficiencia, precisión, exactitud, velocidad, son propiedades que le permitirá al sistema informático responder a las solicitud del usuario, además de aprovechar los recursos que se disponen en el modelo cliente/servidor, obteniendo así un mejor tiempo de respuesta, una mayor velocidad de procesamiento y un mayor aprovechamiento de los recursos.[28]

- **Soporte**

Con el sistema concluido se realizaran pruebas en el lugar que se va a utilizar el software para observar su comportamiento, el mismo presentara una documentación detallada para garantizar un mejor el entendimiento de la misma y se realizara mantenimiento al mismo con el fin de aumentar las funcionalidades del software.[29]

- **Portabilidad**

El producto podrá ser usado bajo los sistemas operativos *GNU/Linux* y *Windows 95* o superior.

- **Políticos-culturales**

La aplicación debe cumplir con las regulaciones políticas de la entidad que utilice el sistema informático.

- **Confiabilidad**

El sistema debe ser preciso, rápido y confiable, no debe presentar fallos, y en caso de ocurrir tratar de solucionarlos rápidamente, además las operaciones que debe realizar deben ser lo más seguras posibles.[30]

- **Interfaz Interna**

El sistema debe presentar una interfaz cómoda, fácil de usar y completamente sencilla, además de brindar una ayuda en caso de dudas del funcionamiento del sistema informático.

- **Ayuda y conexión en línea**

El sistema informático debe estar equipado con un modulo de información, para así aclarar cualquier duda que pueda existir a los usuarios, además debe brindar la opción de conexión en línea para que otros usuarios puedan compartir el sistema informático.

- **Software**

La aplicación debe poderse ejecutar en entornos *Unix* y *GNU/Linux*.

(Multiplataforma), para su ejecución del lado del servidor necesita preferentemente GNU/Linux como sistema operativo, PostgreSQL como sistema gestor de base de datos, Apache como servidor Web. Del lado del cliente cualquiera de los navegadores Web existentes en el mercado con la opción de Java Script activada.[31]

- **Hardware**

Se requiere de una computadora como servidor de base de datos con PostgreSQL como gestor de base de datos, con una capacidad de memoria RAM no menor de 384 Mbyte, microprocesador Pentium III a 700 MHz como mínimo. La terminal del cliente sólo requerirá de una computadora conectada a la red, para poder ejecutar los navegadores Web, las mismas deben cumplir los requisitos mínimos de hardware del sistema operativo que se emplee.[31]

- **Seguridad**

El contenido del sistema estará protegido contra accesos no autorizados utilizando la autenticación como mecanismo de control, además de brindar el acceso solamente a los usuarios que estén asignados para su control y mantenimiento, de manera que el encargado de la seguridad informática pueda gestionar y auditar la información de las salvas de correo electrónico.[32]

2.4- Modelo de casos de uso del sistema

2.4.1 Actores del modelo del sistema

Un actor no es más que un conjunto de roles que los usuarios desempeñan cuando interaccionan con estos Casos de Uso. Los actores representan a terceros fuera del sistema que colaboran con el mismo. Una vez identificado los actores del sistema, queda identificado el entorno externo del sistema.

Nombre del Actor	Descripción.
Encargado de la Seguridad Informática.	Es quien utiliza el sistema para conocer parámetros como estabilidad, disponibilidad y seguridad de la red. Su tarea fundamental es la de revisar que el las salvas de correo electrónico no haya ninguna anomalía, además es el encargado de configurar el sistema para la búsqueda de anomalías.

Tabla 1 Actores del sistema

2.4.2- Casos de uso del sistema

“La 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.”

- 1- Autenticarse.
- 2- Gestionar url.
- 3- Gestionar Usuario.
- 4- Gestionar Configuración.
- 5- Gestionar normas de seguridad.
- 6- Mostrar alarmas.
- 7- Datos del correo.
- 8- Buscar.

Requisitos

- 1- Cerrar sesión.
- 2- Consultar ayuda del sistema.

2.4.3- Diagramas de casos de uso del sistema

Las condiciones y posibilidades conjuntamente con los requisitos que debe cumplir el sistema permiten a los desarrolladores de software llegar a un acuerdo de cómo quedaría mejor realizado el software, de aquí que se tomen acuerdos entre los desarrolladores y se hagan diferentes análisis y pruebas en su transcurso de implementación.

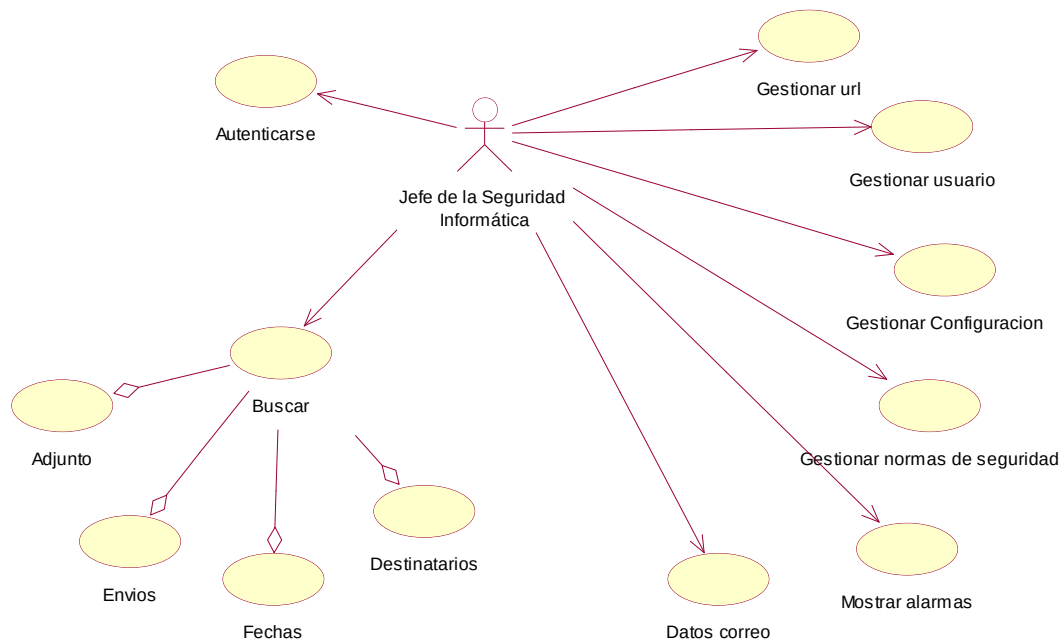


Figura 3 Diagrama de casos de uso del sistema.

2.4.4- Diagramas de funcionamiento del framework CodeIgniter

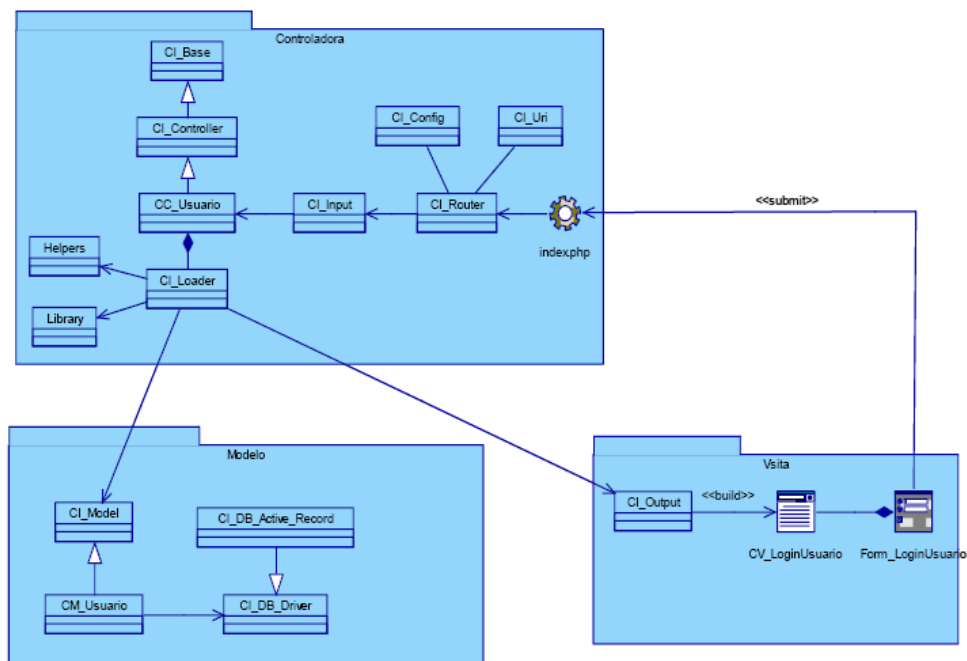


Figura 4 Diagrama de funcionamiento del CodeIgniter.

2.4.5- Descripción de los casos de uso del sistema

Nombre del caso de uso	Autenticarse.
Actores	Jefe de la Seguridad informática (inicia).
Propósito	Acceder a las funcionalidades del sistema.
Resumen	El caso de uso se inicia cuando el Jefe de la Seguridad informática desea acceder a información y/o funcionalidades brindadas por el sistema. Para esto debe introducir su cuenta y contraseña, el sistema verifica los parámetros antes mencionados. Si los mismos son correctos, es permitido el acceso al sistema, en caso contrario, es negado el acceso; culminando así el caso de uso.
Referencias	R1
Precondiciones	Debe existir un jefe de Seguridad Informática con cuenta y contraseña correcta.
Post-condiciones	El Jefe de Seguridad Informática puede acceder a la información y a las funciones brindadas por el sistema.
Requisitos especiales	-
Prototipo	Anexo 1

Tabla 2 Descripción del caso de uso de sistema: Autenticarse

Nombre del caso de uso	Gestionar url
Actores	Jefe de la Seguridad informática (inicia).
Propósito	Insertar, eliminar o cambiar url de las salvas de correo electrónico.
Resumen	Este caso de uso toma vida cuando el jefe de la Seguridad informática desea insertar, eliminar o modificar una url de las salvas de correo electrónico.
Referencias	R2, R3, R4
Precondiciones	El actor principal debe estar ingresado en el sistema.
Post-condiciones	Se cambian los datos de la Base Datos, permitiendo cambiar o eliminar estos datos.
Requisitos especiales	-
Prototipo	Anexo 2

Tabla 3 Descripción del caso de uso de sistema: Gestionar url

Nombre del caso de uso	Gestionar Usuario.
Actores	Jefe de la Seguridad informática (inicia).
Propósito	Insertar, eliminar o cambiar contraseña.
Resumen	Este caso de uso toma vida cuando el jefe de la Seguridad informática desea insertar o eliminar un usuario o cambiar contraseña de uno dentro.
Referencias	R5, R6, R7
Precondiciones	El actor principal debe estar ingresado en el sistema.
Post-condiciones	Se cambian los datos de la Base Datos, permitiendo cambiar o eliminar estos datos.
Requisitos especiales	-
Prototipo	Anexo 3

Tabla 4 Descripción del caso de uso de sistema: Gestionar Usuario

Nombre del caso de uso	Gestionar Configuración.
Actores	Jefe de la Seguridad informática (inicia).
Propósito	El Jefe de la Seguridad informática desea configurar el sistema para la realización de la búsqueda de las normas de seguridad.
Resumen	El caso de uso se inicia cuando el Jefe de la Seguridad informática desea configurar el sistema nuevamente para que el mismo realice las búsquedas establecidas en las diferentes salvas de correo electrónico, culminando el caso de uso cuando el envía esos cambios al sistema y este se encarga de hacer las búsquedas necesarias buscando los datos que se escribieron.
Referencias	R8
Precondiciones	El jefe de Seguridad Informática debe haber ingresado en el sistema.
Post-condiciones	Quedan modificados los parámetros de la configuración del sistema.
Requisitos especiales	-
Prototipo	Anexo 4

Tabla 5 Descripción del caso de uso de sistema: Gestionar Configuración

Nombre del caso de uso	Gestionar norma de seguridad.
Actores	Jefe de la Seguridad informática (inicia).
Propósito	El Jefe de la Seguridad informática desea gestionar las normas de seguridad del sistema.
Resumen	El caso de uso se inicia cuando el Jefe de la Seguridad informática desea gestionar las normas de seguridad por el que el sistema se rige para la realización de las inspecciones en las salvas de correo electrónico.
Referencias	R9, R10, R11
Precondiciones	El jefe de Seguridad Informática debe haber ingresado en el sistema.
Post-condiciones	Quedan modificadas las nuevas normas de seguridad informática.
Requisitos especiales	-
Prototipo	Anexo 5

Tabla 6 Descripción del caso de uso de sistema: Gestionar Norma de Seguridad

Nombre del caso de uso	Mostrar Alarmas
Actores	Jefe de la Seguridad informática (inicia).
Propósito	Posibilitar al Jefe de la Seguridad informática ver todos los correos electrónicos en los cuales se encontraron las normas de seguridad establecidas.
Resumen	El caso de uso se inicia cuando el Jefe de la Seguridad informática desea ver los correos electrónicos en los cuales se encontraron las normas de seguridad establecidas para la realización de la búsqueda de las mismas en el cuerpo de los correos electrónicos, finalizando con el cierre del caso de uso.
Referencias	R12
Precondiciones	El jefe de Seguridad Informática debe haber ingresado en el sistema.
Post-condiciones	Se hace una revisión de todas las salvas de correo.
Requisitos especiales	-
Prototipo	Anexo 6

Tabla 7 Descripción del caso de uso de sistema: Mostrar Alarmas

Nombre del caso de uso	Datos del correo electrónico
Actores	Jefe de la Seguridad informática (inicia).
Propósito	Posibilitar al Jefe de la Seguridad informática ver todos los datos del correo electrónico señalado.
Resumen	El caso de uso se inicia cuando el Jefe de la Seguridad informática desea ver los datos del correo electrónico señalado, posibilitando así el chequeo del mismo de forma visual, finalizando con el cierre del caso de uso.
Referencias	R13
Precondiciones	El jefe de Seguridad Informática debe haber ingresado en el sistema.
Post-condiciones	Se hace una revisión de todas las salvas de correo.
Requisitos especiales	-
Prototipo	Anexo 7

Tabla 8 Descripción del caso de uso de sistema: Datos del correo electrónico

Nombre del caso de uso	Buscar
Actores	Jefe de la Seguridad informática (inicia).
Propósito	Posibilitar al Jefe de la Seguridad informática hacer búsquedas en las diferentes salvas de correo electrónico.
Resumen	El caso de uso se inicia cuando el Jefe de la Seguridad informática desea hacer una búsqueda de algún adjunto en especial en las diferentes salvas de correo electrónico, el sistema le muestra los adjuntos de los correos que cumplan con el buscado, finalizando con el cierre del caso de uso.
Referencias	R14, R15, R16, R17
Precondiciones	El jefe de Seguridad Informática debe haber ingresado en el sistema.
Post-condiciones	Se hace una revisión de todas las salvas de correo.
Requisitos especiales	-
Prototipo	Anexo 8

Tabla 9 Descripción del caso de uso de sistema: Buscar

2.5- Diagrama de clases del diseño

En este epígrafe se describe mediante graficas las clases del software, las interfaces y sus relaciones en la aplicación. En una web, dichas clases tienen un significado importante ya que las mismas describen las relaciones entre páginas y denotando así como una clase cualquier pagina lógica.

Caso de Uso	Diagrama de Clases.
Autenticarse	Anexo 9
Gestionar url	Anexo 10
Gestionar usuario	Anexo 11
Gestionar configuración	Anexo 12
Gestionar Normas	Anexo 13
Anomalía	Anexo 14
Datos del correo	Anexo 15
Buscar	Anexo 16

Tabla 10 Diagrama de clases de diseño

2.6- Diseño de la base de datos

2.6.1- Modelo lógico de datos

En este epígrafe se muestra la vista de las entidades lógicas de la base datos y sus relaciones.

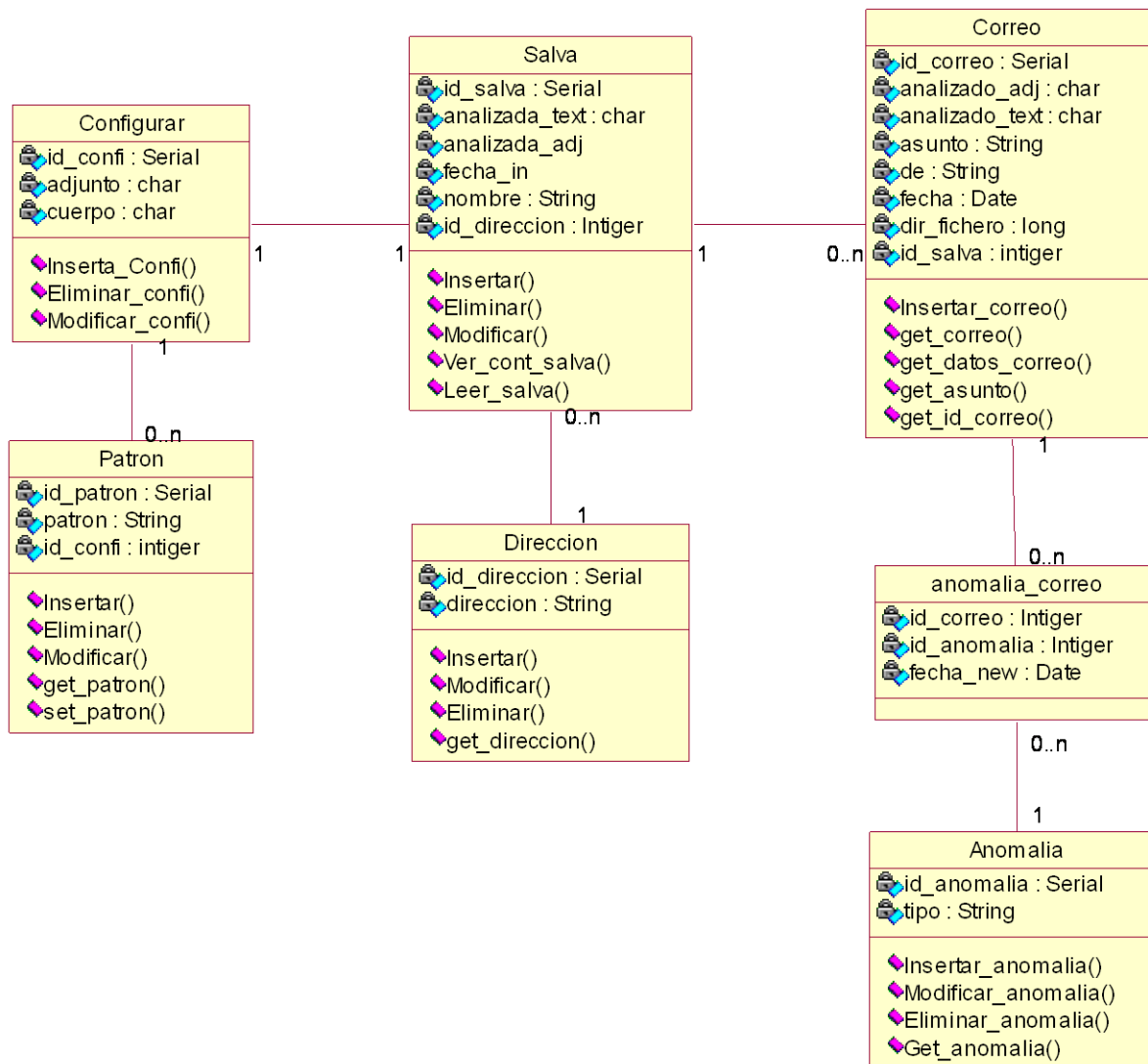


Figura 5Diagrama de clases persistentes para la base datos de salvas de correo.

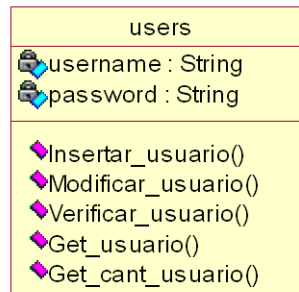


Figura 6 Diagrama de clases persistentes para la base datos de usuarios

2.6.2- Modelo físico de datos

Se detalla la representación lógica y física de los datos persistentes (las tablas que conforman la base datos). Además es el resultado de la actividad de diseño de la base datos.

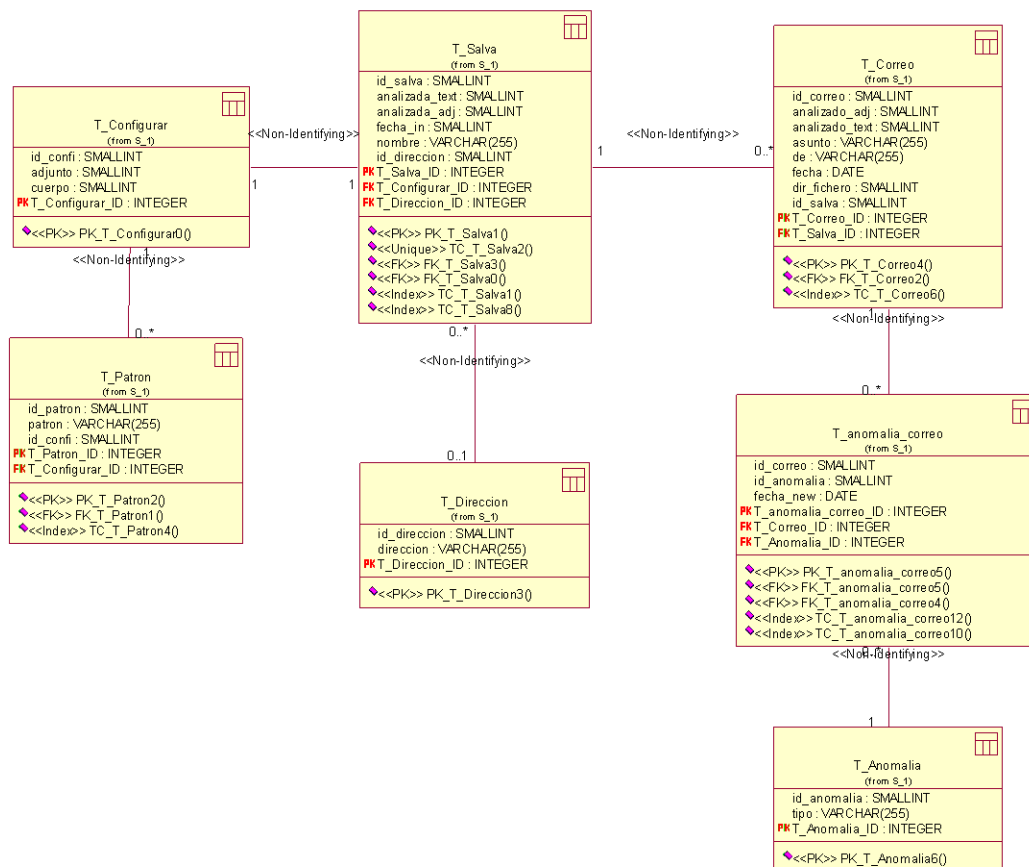


Figura 7 Diagrama del modelo físico de datos para la base datos de salvadas de correo.

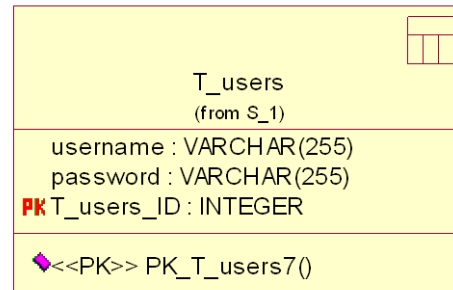


Figura 8 Diagrama del modelo físico de datos para la base datos de usuarios del sistema

2.6.3- Diagrama de implementación

Este epígrafe muestra una descripción de todos los componentes a utilizar, así como sus dependencias y organización atendiendo a los nodos que aquí se muestran, mostrando de forma más sintetizada los diagramas de Componente y de Despliegue.

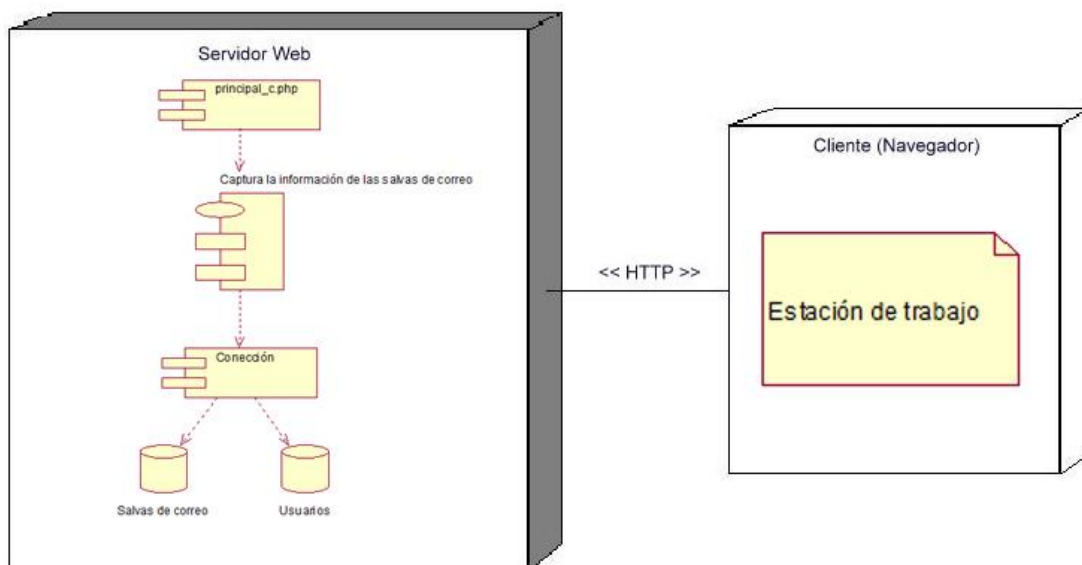


Figura 9 Diagrama de implementación

2.6.4- Principios de diseño

El desarrollo de este punto está orientado a facilitar la eficiencia y rapidez en la utilización del software, así como facilitar la gestión y el control de las salvas de correo electrónico, garantizando la comodidad a la hora de la administración de las mismas.

2.6.5- Estándares en la interfaz de la aplicación

Para la creación de la interfaz del sistema se ha escogido el ExtJs que es un framework programado con una serie de métodos y funciones las cuales son capaces de crear formularios como si fuera una aplicación desktop, el mismo ya tiene incluido el color a utilizar el cual es un azul claro, ese color esta en toda la página ya que la misma está completamente hecha con él. La letra que se utiliza es clara ya que se tomo el formato de letra arial, además se evita el empleo de muchas imágenes y animaciones ya que se desea que la aplicación sea rápida. El sistema cuenta con un diseño único, el cual lo integran un árbol de navegación en la parte izquierda del sitio con todas las salvas contenidas en una url escogida, una parte central y un banner. El mismo presenta una sola página principal ya que el framework utilizado brinda esta posibilidad.

2.6.6- Tratamiento de errores

Gracias a la utilización del framework ExtJs para la creación del sistema informático, el mismo está programado utilizando técnicas de validación de datos (expresiones regulares), posibilitando así que se minimicen los errores en el sistema ya que no hay necesidad de validar los errores en el servidor. Los mensajes que el sistema muestra se muestran en un lenguaje claro para un mejor entendimiento.

2.6.7- Concepción General de la ayuda

La ayuda de un software es una de las cosas más importantes del mismo, es por ello que el sistema propuesto cuenta con una, la cual el jefe de la seguridad informática tendrá a su disposición en todo momento, permitiéndole aclarar cualquier duda que pueda surgirle. Además el sistema informático presentara ayudas independientes atendiendo al caso de uso analizado, la misma se encontrara en la parte superior de los formularios en un botón con un signo de interrogación, y se mostrara en una ventana diferente.

2.7- Conclusiones

Con el desarrollo de este capítulo se logra un mejor entendimiento de la solución propuesta al definir las entidades que están involucradas en la misma, modelar las relaciones que se establecen entre ellas y establecer las reglas del negocio a considerar.

Se han mostrado los diferentes requerimientos funcionales y no funcionales del sistema propuesto. Además se detallo el modelo del sistema a través de sus casos de usos y se mostró los diagramas de casos de uso.

Contiene una descripción detallada de la solución propuesta del sistema, mostrando la misma utilizando la representación gráfica de los diagramas de clases del modelo de sistema y el diagrama de implementación. Además se describió el estándar que va a tener el sistema, el tipo de letra a utilizar, se describió como serán los mensajes de error y el sistema de ayuda a utilizar.

Capítulo III.- “Análisis de prueba y estudio de factibilidad”:

El siguiente capítulo muestra primeramente de forma detallada una serie de pruebas realizadas al sistema y seguido a esto un análisis de factibilidad para el sistema informático propuesto, utilizando como metodología de trabajo COCOMO.

3.1- Experimentos y Pruebas realizadas

A fin de comprobar el funcionamiento del sistema desarrollado se realizaron pruebas prácticas simulando posibles situaciones que afectarían la estabilidad, operatividad y seguridad en la red telemática.

3.2- Pruebas Realizadas

3.2.1- Buscar patrones alarmantes en los adjuntos de los correos electrónicos.

Esta prueba consistió en realizar una búsqueda de patrones alarmantes en los adjuntos de los correos electrónicos, con el encuentro de alguno se envía un correo de aviso (puede ser modificado por el administrador) a la dirección de correo electrónico examinada, y se muestra una alarma al administrador.

La prueba fue simulada haciendo uso de diferentes adjuntos de correos electrónicos, se añadió a los mismos patrones alarmantes, obteniéndose así la dirección de los correos electrónicos deseados; se envió un correo electrónico de alarma a la dirección examinada y se mostró de una alarma al administrador del sistema.

3.2.2- Buscar palabras claves en los correos electrónicos.

Esta prueba consiste en la búsqueda de una palabra clave en el cuerpo de los correos electrónicos, con su detención el sistema envía un correo de forma automática, a la dirección de correo electrónico que posea dicha alarma, con un mensaje de aviso, dicho mensaje es configurado por el administrador del sistema.

La prueba fue realizada añadiendo palabras claves en el cuerpo de diferentes correos electrónicos, en la búsqueda se obtuvieron todos los correos electrónicos con la información

buscada y se envió de forma automática por el sistema un correo de aviso (Puede ser modificado por el administrador).

3.2.3- Mostrar correos electrónicos comprendidos en fechas.

Esta prueba consiste en mostrar todos los correos electrónicos que su fecha de entrada este comprendida entre las fechas dadas, el sistema muestra los correos en una tabla, posibilita ver su contenido, enviar correos, etc.

Esta prueba fue realizada tomando como referencia dos fechas, de la misma se obtuvo todos los correos comprendidos en las mismas.

3.2.4- Buscar alarmas en las salvas de correo electrónico.

Esta prueba consiste en buscar alarmas en las salvas de correo electrónico, con su detención el sistema de forma automática muestra una alarma.

Para esta prueba el administrador del sistema configuró la alarma a buscar, luego se realizó la búsqueda en todas las salvas de correo electrónico y se mostró una alarma con las salvas de correo que poseían anomalías.

3.2.5- Buscar correos electrónicos por su destino.

Consiste en la búsqueda de los correos electrónicos que su destino sea el mismo que una dirección dada, el sistema da la posibilidad de mostrar los correos que presenten dicha dirección como destino.

Para esta prueba se tomo una dirección de correo electrónico y se realizó una búsqueda en las salvas de correo electrónico.

3.3- Estudio de Factibilidad

En siguiente capítulo se realiza un análisis de factibilidad del sistema informático propuesto, el cual consiste en determinar con cierto grado de certeza, los recursos, tiempo, costo y esfuerzo necesario para el desarrollo del mismo.

Para darle cumplimiento a tal análisis se ha aplicado el modelo de estimación llamado COCOMO (Costructive Cost Model).

3.3.2- Planificación por casos de usos

La estimación mediante el análisis de Puntos de Casos de Uso es un método propuesto originalmente por Gustav Karner, y posteriormente refinado por muchos otros autores. Se trata de un método de estimación del tiempo de desarrollo de un proyecto mediante la asignación de "pesos" a un cierto número de factores que lo afectan, para finalmente, contabilizar el tiempo total estimado para el proyecto a partir de esos factores.[33]

3.3.3- Cálculo de Puntos de Casos de Uso sin Ajustar

$$UUCP = UAW + UUCW$$

Donde:

UUCP: Puntos de Casos de Uso sin ajustar.

UAW: Factor de Peso de los Actores sin ajustar.

UUCW: Factor de Peso de los Casos de Uso sin ajustar.

Cálculo de UAW:

Actor	Factor Peso.
Jefe de la Seguridad Informática.	3

Tabla 11 Asignación de peso a los actores.

$$UAW = (\text{Cantidad de actores}) * \text{Peso}$$

$$UAW = 1 * 3$$

$$UAW = 3$$

Cálculo de UUCW:

Caso de Uso	Tipo
Autenticarse.	Simple
Generar Alarma	Medio
Buscar Adjuntos	Simple
Buscar Patrones Alarmantes	Simple

Buscar por Destinatarios	Simple
Buscar por Envíos	Simple
Buscar por Fechas	Simple
Mostrar correo	Medio

Tabla 12 Factor de peso de los casos de uso sin ajustar.

$$UUCW = 6*5 + 2 *10$$

$$UUCW = 50$$

Los puntos de casos de uso sin ajustar resultan:

$$UUCP = UAW + UUCW$$

$$UUCP = 3 + 50$$

$$UUCP = 53$$

3.3.4 - Cálculo de los Puntos de Casos de Uso Ajustados

$$UCP = UUCP \times TCF \times EF$$

Donde:

UCP: Puntos de Casos de Uso ajustados.

TCF: Factor de complejidad técnica.

EF: Factor de ambiente.

Cálculo de TCF:

Factor	Descripción	Peso	Valor Asig.	Total
T1	Sistema distribuido	2	1	2
T2	Tiempo de respuesta	1	4	4
T3	Eficiencia del usuario final	1	5	5
T4	Procesamiento interno complejo	1	4	4
T5	El código debe ser reutilizable	1	3	3
T6	Facilidad de instalación	0.5	5	2.25
T7	Facilidad de uso	0.5	5	2.25

T8	Portabilidad	2	5	10
T9	Facilidad de cambio	1	4	4
T10	Concurrencia	1	4	4
T11	Incluye objetivos especiales de seguridad	1	3	3
T12	Provee acceso directo a terceras partes	1	0	0
T13	Se requiere facilidades especiales de entrenamiento a usuarios	1	1	1

Tabla 13 Cálculo del factor de complejidad técnica.

$$TCF = 0.6 + 0.01 * S(\text{Peso } i * \text{Valor asignado } i)$$

$$TCF = 0.6 + 0.01 * (2 + 4 + 5 + 4 + 3 + 2.25 + 2.25 + 10 + 4 + 4 + 3 + 0 + 1)$$

$$TCF = 0.6 + 0.01 * 44.5$$

$$TCF = 0.6 + 0.445$$

$$TCF = 1.045$$

Cálculo de EF

Factor	Descripción	Peso	Valor Asig.	Total
E1	Familiaridad con el modelo de proyecto utilizado	1.5	4	6
E2	Experiencia con la aplicación	0.5	3	1.5
E3	Experiencia en orientación a objetos	1	2	2
E4	Capacidad del analista líder	0.5	4	2
E5	Motivación	1	5	5
E6	Estabilidad de los requerimientos	2	4	8
E7	Personal a tiempo parcial	-1	0	0
E8	Dificultad del lenguaje de programación	-1	3	-3

Tabla 14 Cálculo del factor ambiente.

$$EF = 1.4 - 0.03 * S(\text{Peso}i * \text{Valor asignado}i)$$

$$EF = 1.4 - 0.03 * (6+1.5+2+2+5+8+0-3)$$

$$EF = 1.4 - 0.03 * 21.5$$

$$EF = 1.4 - 0.645$$

$$EF = 0.755$$

Los puntos de casos de uso ajustados resultan:

$$UCP = UUCP * TCF * EF$$

$$UCP = 53 * 1.045 * 0.755$$

$$UCP = 41.815675$$

3.3.5- Estimación del esfuerzo

CF: Factor de Conversión

$$CF = 20 \text{ Horas-Hombre}$$

El esfuerzo en horas /hombre está dado por:

$$E = UCP * CF$$

$$E = 41.815675 * 20 = 836.3135 \text{ horas-hombre.}$$

Duración:

Trabajando 25 días al mes y 12 horas diarias como promedio, se tiene que:

$$\text{Duración (días)} = \text{Total de Horas /Hombre entre 12 horas al día} = 5254/12 = 437.8$$

Días.

$$\text{Duración (meses)} = \text{Total de días} / 30 \text{ días por mes} = 437.8 / 25 = 17.512 \sim 18$$

Meses.

Actividad	Porcentaje	Esfuerzo (H/H)
Análisis	10%	67.95
Diseño	20%	135.9
Programación	40%	836.3135
Prueba	15%	101.92
Sobrecarga	15%	101.92
Total de Horas	100%	1244

Tabla 15 Criterios de distribución de esfuerzos.

3.3.6- Cálculo de costos

Tomando como salario promedio mensual \$275.00

Costo = 18 meses * \$275.00 = \$4950.00

3.3.7- Beneficios tangibles e intangibles

Con el desarrollo de este sistema informático se logra obtener los datos de los correos electrónicos, ayudando de esta manera en la revisión de los mismos en busca de encontrar algún patrón o norma de seguridad establecida por el Jefe de la Seguridad Informática de La Refinería de Petróleo “Camilo Cienfuegos”. De esta manera se da cumplimiento a los objetivos trazados ya que el desarrollo del sistema informático está enfocado a los mismos.

3.3.8- Análisis de costos y beneficios

Anteriormente la tarea de revisión de los correos electrónicos encapsulados en las salvas de correo electrónico se realizaba de forma manual consumiendo un nivel de tiempo enorme.

Con la ayuda del sistema informático propuesto se logra mejorar la tarea antes descrita, con ayuda de un sistema de alarmas totalmente dinámico.

Con el análisis del costo del sistema, los beneficios que brinda el mismo y dando cumplimiento a los objetivos planteados, se concluye que la aplicación es factible.

3.4- Conclusiones

A lo largo de este capítulo se describen los resultados obtenidos al someter el sistema a pruebas, validando los mismos con herramientas a fines, obteniéndose en cada prueba los resultados deseados. Además se realizó el estudio de factibilidad económica del desarrollo del sistema.

Conclusiones

Con la realización del sistema informático propuesto, La Refinería de Petróleo se verá beneficiada, gracias a que el mismo es capaz de gestionar las salvas de correo electrónico generadas por el servicio de correo de dicha empresa, de aquí que permita facilitar las tareas de supervisión y control y garantice la detención de cualquiera anomalía que exista en las mismas.

De igual forma se puede afirmar:

- Se desarrolló una herramienta de software para el análisis de las salvas de correo electrónico de La Refinería de Petróleo.
- Se diseñó un sistema informático que gestiona las salvas de correo electrónico de La Refinería de Petróleo.
- Se implementó un sistema informático web, que brinda opciones de análisis y de alarmas al usuario en caso de encontrar alguna norma de seguridad en el cuerpo de los correos.
- Se validó el sistema final mediante el empleo de herramientas de software y el análisis del comportamiento del sistema.

Recomendaciones

- Incorporar técnicas de extracción de datos en el encuentro de las normas de seguridad establecidas en la empresa en el cuerpo de los correos, a fin de mostrarlos con un mayor rigor.

Referencias bibliográficas

- [1] «¿Qué es un servidor web (Web Servers)? - Definición de servidor web». [Online]. Available: <http://www.masadelante.com/faqs/servidor-web>. [Accessed: 01-jun-2012].
- [2] «PHP: Servidor web embebido - Manual». [Online]. Available: <http://php.net/manual/es/features.commandline.webserver.php>. [Accessed: 01-jun-2012].
- [3] «Servicio de correo electrónico (SMTP/POP3-IMAP4) — Documentación de Zentyal 2.2». [Online]. Available: <http://doc.zentyal.org/es/mail.html>. [Accessed: 02-jun-2012].
- [4] «Clientes de correo». [Online]. Available: <http://www.iata.csic.es/~fernando/doc/clientecorreo.shtml>. [Accessed: 02-jun-2012].
- [5] «El cliente de correo electrónico - Proporcionado por Suempresa.com SupportSuite». [Online]. Available: http://soporte.suempresa.com/index.php?_m=knowledgebase&_a=viewarticle&kbarticleid=74&nav=0. [Accessed: 02-jun-2012].
- [6] «Cómo utilizar el archivo MBOX en Entourage para Mac». [Online]. Available: <http://support.microsoft.com/kb/273841/es>. [Accessed: 02-jun-2012].
- [7] «Correo electrónico - EcuRed». [Online]. Available: http://www.ecured.cu/index.php/Correo_electr%C3%B3nico. [Accessed: 02-jun-2012].
- [8] «tema5b.pdf». .
- [9] «Instalación y configuración de Squirrelmail | Recursos para el CFGM Sistemas Microinformáticos y Redes.» [Online]. Available: <http://ciclo.tuinstituto.es/?q=node/199>. [Accessed: 02-jun-2012].
- [10] «Manual_Uso_correo_upel.pdf». .
- [11] «Comando grep». [Online]. Available: <http://120linux.com/comando-grep/>. [Accessed: 02-jun-2012].
- [12] «Comando Linux egrep - Ejemplo de comando Encontrar buscar Archivos, grep Palabra, Mostrar Línea». [Online]. Available: <http://es.hscripts.com/tutoriales/linux-commands/egrep.html>. [Accessed: 02-jun-2012].
- [13] «lf326, SystemAdministration: rsync: El mejor sistema de copias de seguridad existente». [Online]. Available: <http://www.linuxfocus.org/Castellano/March2004/article326.shtml>. [Accessed: 02-jun-2012].

- [14] «Rsync tutorial». [Online]. Available: <http://www.fredshack.com/docs/rsync.html>. [Accessed: 02-jun-2012].
- [15] «Manpage of LOGROTATE». [Online]. Available: <http://es.tldp.org/Paginas-manual/man-pages-es-extra-0.8a/man8/logrotate.8.html>. [Accessed: 02-jun-2012].
- [16] «Qué es NFS». [Online]. Available: http://www.ite.educacion.es/formacion/materiales/85/cd/REDES_LINUX/nfs/Que_es_nfs.html. [Accessed: 02-jun-2012].
- [17] «Manual de HTML», ene-2012. [Online]. Available: <http://wwwapp.etsit.upm.es/~alvaro/manual/manual.html>.
- [18] «Introducción a CSS», ene-2012. [Online]. Available: <http://www.maestrosdelweb.com/editorial/introcss/>.
- [19] «PHP», ene-2012. [Online]. Available: <http://www.maestrosdelweb.com/editorial/desarrollo-web/php/>.
- [20] «Introducción a JavaScript», dic-2011. [Online]. Available: <http://www.maestrosdelweb.com/editorial/%C2%BFque-es-javascript/>.
- [21] «PostgreSQL», 21-mar-2012. [Online]. Available: http://danielpecos.com/docs/mysql_postgres/x15.html. [Accessed: 21-mar-2012].
- [22] «Manual de CodeIgniter en español», 07-mar-2012. [Online]. Available: <http://techtastico.com/post/manual-codeigniter-castellano/>. [Accessed: 07-mar-2012].
- [23] «ExtJS lo bueno, lo malo y lo feo | Desarrollo en Web», 13-mar-2012. [Online]. Available: <http://blogs.antartec.com/desarrolloweb/2008/10/extjs-lo-bueno-lo-malo-y-lo-feo/>. [Accessed: 13-mar-2012].
- [24] «El Lenguaje de Modelado Unificado (UML)». [Online]. Available: <http://www.docirs.cl/uml.htm>. [Accessed: 02-jun-2012].
- [25] «ActaUML.pdf». .
- [26] «Proceso Unificado de Desarrollo - EcuRed». [Online]. Available: http://www.ecured.cu/index.php/Proceso_Unificado_de_Desarrollo. [Accessed: 02-jun-2012].
- [27] «CAP_11_2006_I_SI905_VA4_M.pdf». .

- [28] «SISTEMA INFORMÁTICO PARA LA GESTIÓN DE LA INFORMACIÓN». [Online]. Available: <http://www.eumed.net/cursecon/ecolat/cu/2012/opah.html>. [Accessed: 02-jun-2012].
- [29] «Soporte de Sistemas Informáticos». [Online]. Available: http://www.acambiode.com/empresa/soporte-de-sistemas-informaticos_280883. [Accessed: 02-jun-2012].
- [30] «Análisis de rendimiento y fiabilidad de sistemas informáticos - E-Prints Complutense». [Online]. Available: <http://eprints.ucm.es/11265/>. [Accessed: 02-jun-2012].
- [31] «Definición de software y hardware -¿Qué es software y qué es hardware?» [Online]. Available: <http://www.masadelante.com/faqs/software-hardware>. [Accessed: 02-jun-2012].
- [32] «Argentine BriefS.pdf». .
- [33] «Una técnica de estimación de proyectos basada en Casos de Uso « Iterando». [Online]. Available: <http://iterando.wordpress.com/2009/04/09/una-tecnica-de-estimacion-de-proyectos-basada-en-casos-de-uso/>. [Accessed: 03-jun-2012].

Bibliografía

- [1] «¿Qué es un servidor web (Web Servers)? - Definición de servidor web». [Online]. Available: <http://www.masadelante.com/faqs/servidor-web>. [Accessed: 01-jun-2012].
- [2] «Análisis de rendimiento y fiabilidad de sistemas informáticos - E-Prints Complutense». [Online]. Available: <http://eprints.ucm.es/11265/>. [Accessed: 02-jun-2012].
- [3] «Clientes de correo». [Online]. Available: <http://www.iata.csic.es/~fernando/doc/clientecorreo.shtml>. [Accessed: 02-jun-2012].
- [4] «Comando Linux egrep - Ejemplo de comando Encontrar buscar Archivos, grep Palabra, Mostrar Línea». [Online]. Available: <http://es.hscripts.com/tutoriales/linux-commands/egrep.html>. [Accessed: 02-jun-2012].
- [5] «El Lenguaje de Modelado Unificado (UML)». [Online]. Available: <http://www.docirs.cl/uml.htm>. [Accessed: 02-jun-2012].
- [6] «ExtJS lo bueno, lo malo y lo feo | Desarrollo en Web», 13-mar-2012. [Online]. Available: <http://blogs.antartec.com/desarrolloweb/2008/10/extjs-lo-bueno-lo-malo-y-lo-feo/>. [Accessed: 13-mar-2012].
- [7] «Introducción a CSS», ene-2012. [Online]. Available: <http://www.maestrosdelweb.com/editorial/introcss/>.
- [8] «Introducción a JavaScript», dic-2011. [Online]. Available: <http://www.maestrosdelweb.com/editorial/%C2%BFque-es-javascript/>.
- [9] «Manpage of LOGROTATE». [Online]. Available: <http://es.tldp.org/Paginas-manual/man-pages-es-extra-0.8a/man8/logrotate.8.html>. [Accessed: 02-jun-2012].
- [10] «Manual de CodeIgniter en español», 07-mar-2012. [Online]. Available: <http://techtastico.com/post/manual-codeigniter-castellano/>. [Accessed: 07-mar-2012].
- [11] «Manual de HTML», ene-2012. [Online]. Available: <http://www.wapp.etsit.upm.es/~alvaro/manual/manual.html>.
- [12] «PHP», ene-2012. [Online]. Available: <http://www.maestrosdelweb.com/editorial/desarrollo-web/php/>.
- [13] «PostgreSQL», 21-mar-2012. [Online]. Available: http://danielpecos.com/docs/mysql_postgres/x15.html. [Accessed: 21-mar-2012].
- [14] «Proceso Unificado de Desarrollo - EcuRed». [Online]. Available:

http://www.ecured.cu/index.php/Proceso_Unificado_de_Desarrollo. [Accessed: 02-jun-2012].

- [15] «Qué es NFS». [Online]. Available: http://www.ite.educacion.es/formacion/materiales/85/cd/REDES_LINUX/nfs/Que_es_nfs.html. [Accessed: 02-jun-2012].
- [16] «Rsync tutorial». [Online]. Available: <http://www.fredshack.com/docs/rsync.html>. [Accessed: 02-jun-2012].
- [17] «Servicio de correo electrónico (SMTP/POP3-IMAP4) — Documentación de Zentyal 2.2». [Online]. Available: <http://doc.zentyal.org/es/mail.html>. [Accessed: 02-jun-2012].
- [18] «Una técnica de estimación de proyectos basada en Casos de Uso « Iterando». [Online]. Available: <http://iterando.wordpress.com/2009/04/09/una-tecnica-de-estimacion-de-proyectos-basada-en-casos-de-uso/>. [Accessed: 03-jun-2012].

Glosario de términos

Cron: En el sistema operativo Unix, cron es un administrador regular de procesos en segundo plano (demonio) que ejecuta procesos o Scripts a intervalos regulares (por ejemplo, cada minuto, día, semana o mes).

CSS: Las hojas de estilo en cascada son un lenguaje formal usado para definir la presentación de un documento estructurado escrito en HTML o XML.

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.

SALVAS: Ficheros comprimidos en los que se alojan los datos de los correos entrantes y salientes de La Refinería de Petróleo.

SSH: Acrónimo en inglés de "Secure SHell" en español: intérprete de órdenes seguro. Es un protocolo que sirve para acceder a máquinas remotas a través de una red.

RSYNC: Aplicación de tipo Unix y Microsoft que sincroniza ficheros entre diferentes máquinas o entre una misma máquina.

NSF: Es un sistema que utiliza Unix para compartir carpetas en una red.

LogRotate: Sirve para administra los sistemas que generan gran cantidad de archivos de registros (Logs), tratándolos tanto diariamente, como semanalmente o mensualmente.

EGREP: Línea de comandos escrita originalmente para ser usada con el sistema operativo Unix. Permite optimizar la búsqueda en casos particulares, siendo la misma más profunda pero más lenta.

GREP: Línea de comandos escrita originalmente para ser usada con el sistema operativo Unix. Permite buscar las líneas que contienen una cadena de caracteres especificados, utilizando para esto expresiones regulares.

URL:

SQUIRRELMAIL: Cliente de correo electrónico, que permite al usuario la visualización de los correos electrónicos en un servidor dado.

FICHEROS: Documentos de datos en los cuales se guarda información de un determinado tipo.

MBOX: Formato que se utiliza en los ficheros para el resguardo de la información de los correos electrónicos.

MAILDIR: Formato de spool de correo electrónico que no bloquea los ficheros para mantener la integridad del mensaje.

PATRON: Termino o palabra que se toma para la realización de búsquedas en los correos electrónicos.

GNU/Linux: Es el término empleado para referirse al sistema operativo similar a Unix que utiliza como base las herramientas de sistema de GNU y el núcleo Linux.

LOGS: Ficheros de datos en los cuales se almacena información.

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

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.

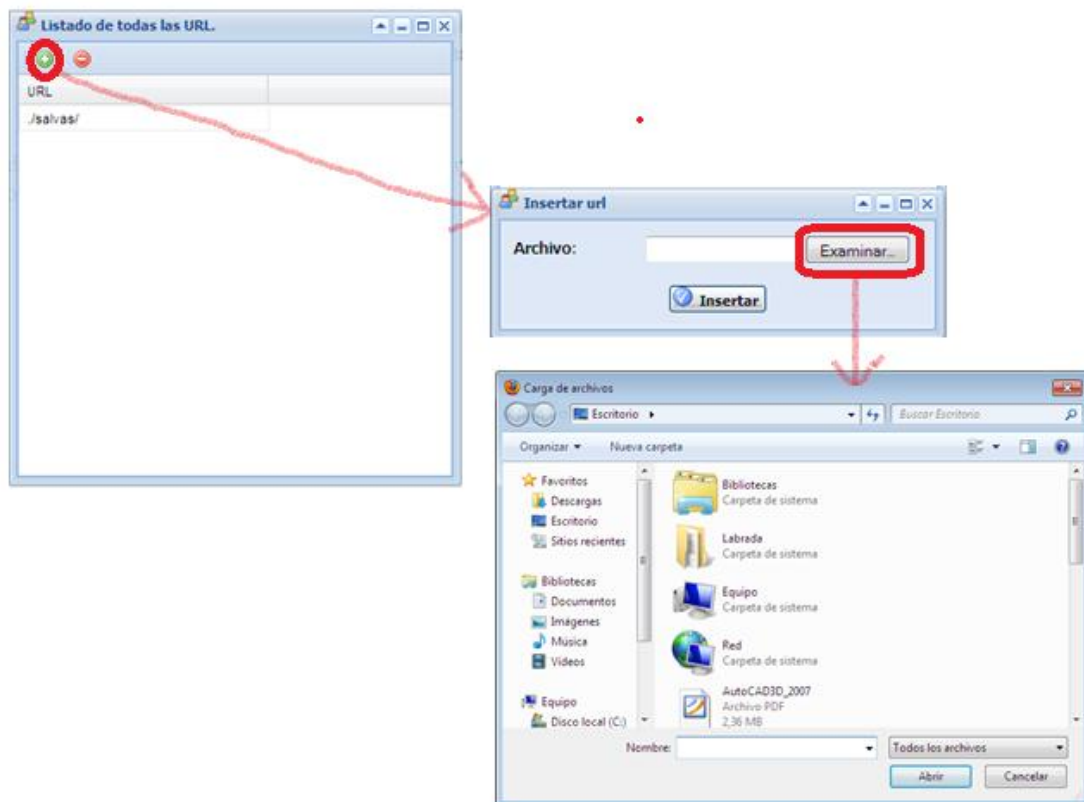
URL: Localizador uniforme de recurso. Es una secuencia de caracteres, de acuerdo a un formato estándar, que se usa para nombrar recursos, como documentos e imágenes en Internet, por su localización.

Anexos

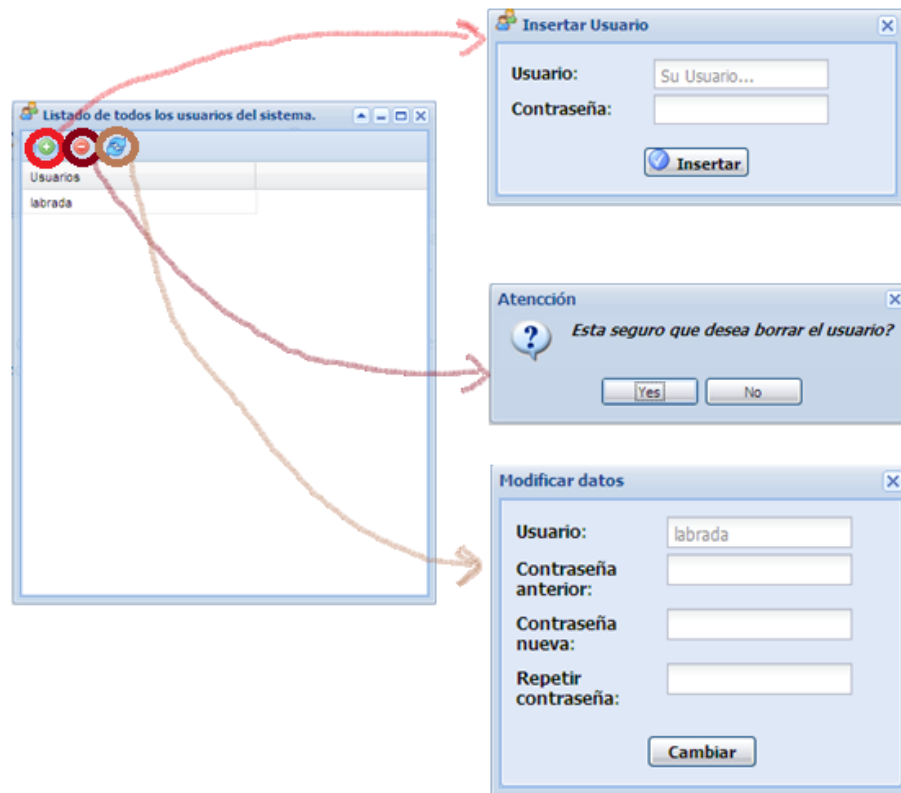


A dialog box titled "Autenticar Usuario" with a key icon. It contains two input fields: "Usuario:" with the placeholder text "Su Usuario..." and "Contraseña:" with a masked password field. At the bottom are two buttons: "Aceptar" (Accept) and "Cancelar" (Cancel).

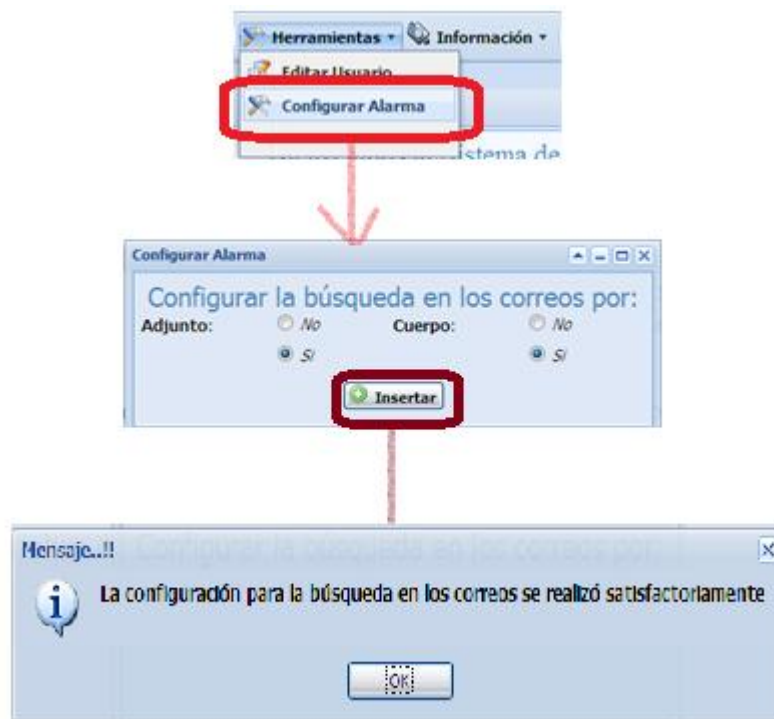
Anexo 1 Prototipo del caso de uso: Autenticarse



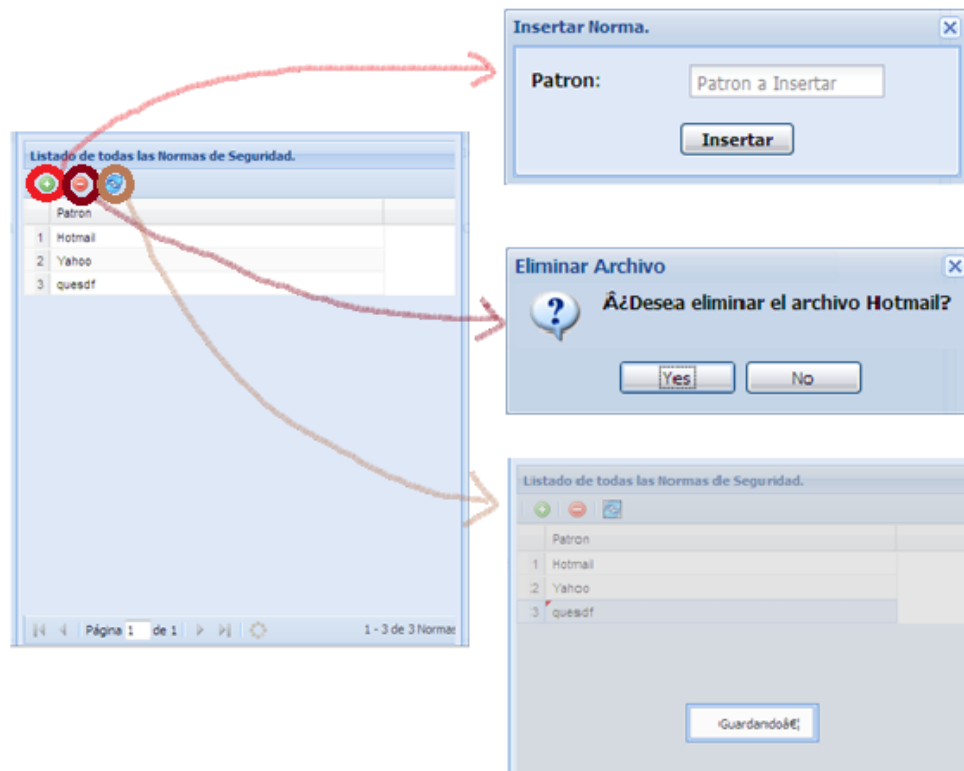
Anexo 2 Prototipo del caso de uso: Gestionar url



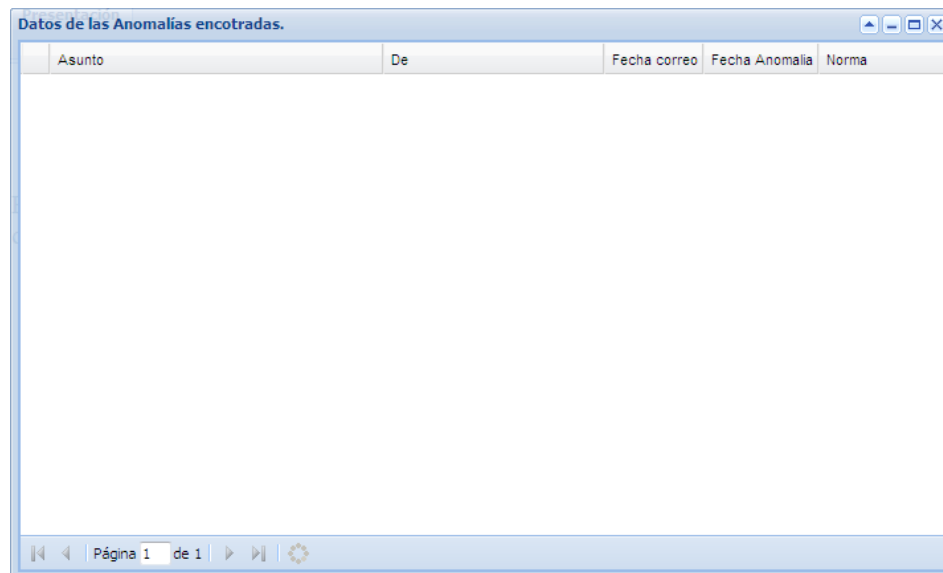
Anexo 3 Prototipo del caso de uso: Gestionar usuario



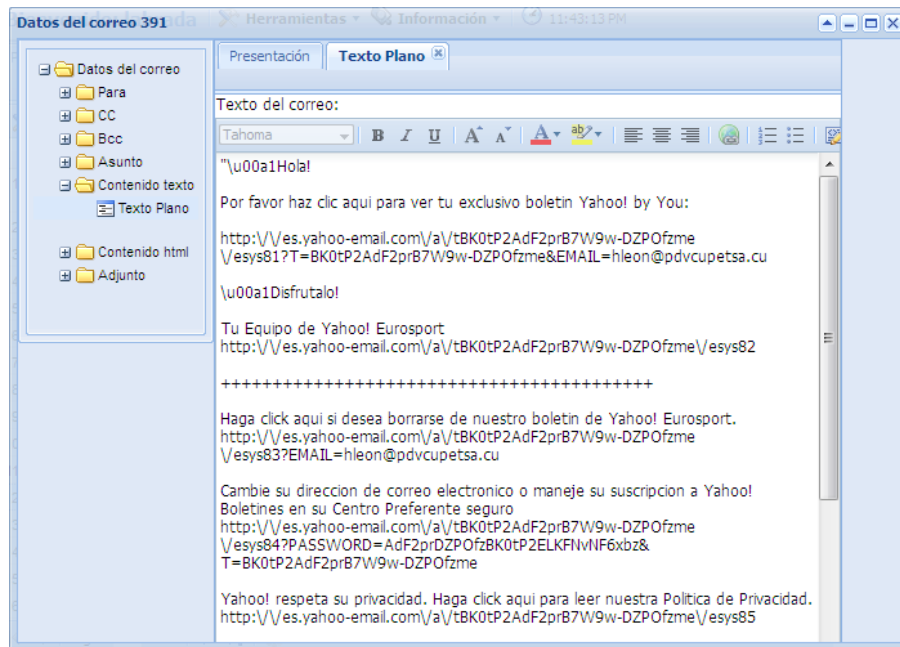
Anexo 4 Prototipo del caso de uso: Gestionar configuración



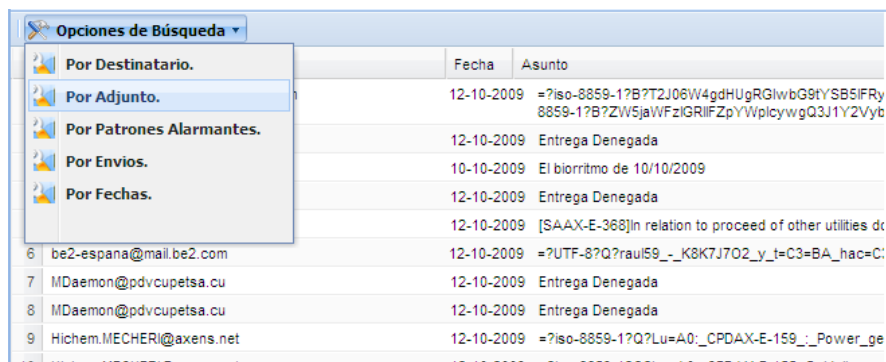
Anexo 5 Prototipo del caso de uso: Gestionar norma de seguridad



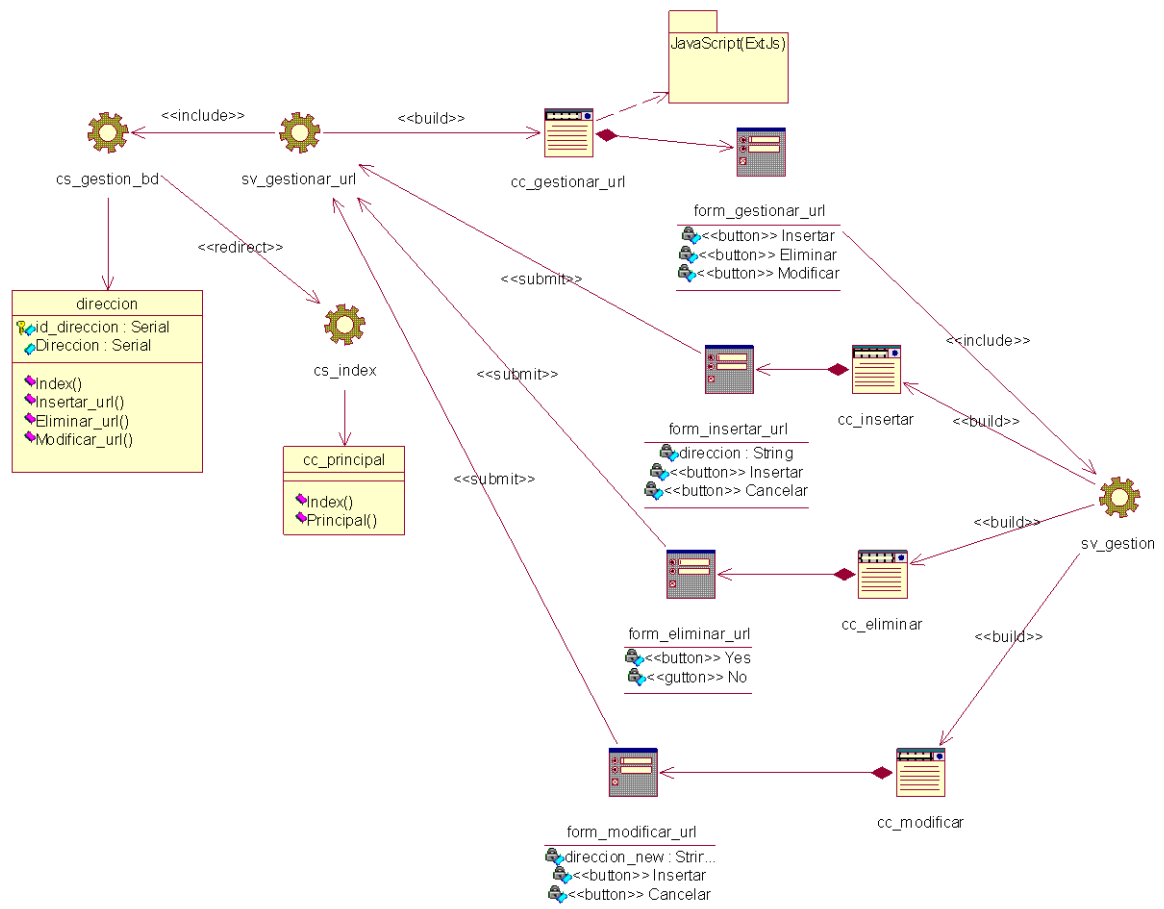
Anexo 6 Prototipo del caso de uso: Mostrar alarmas



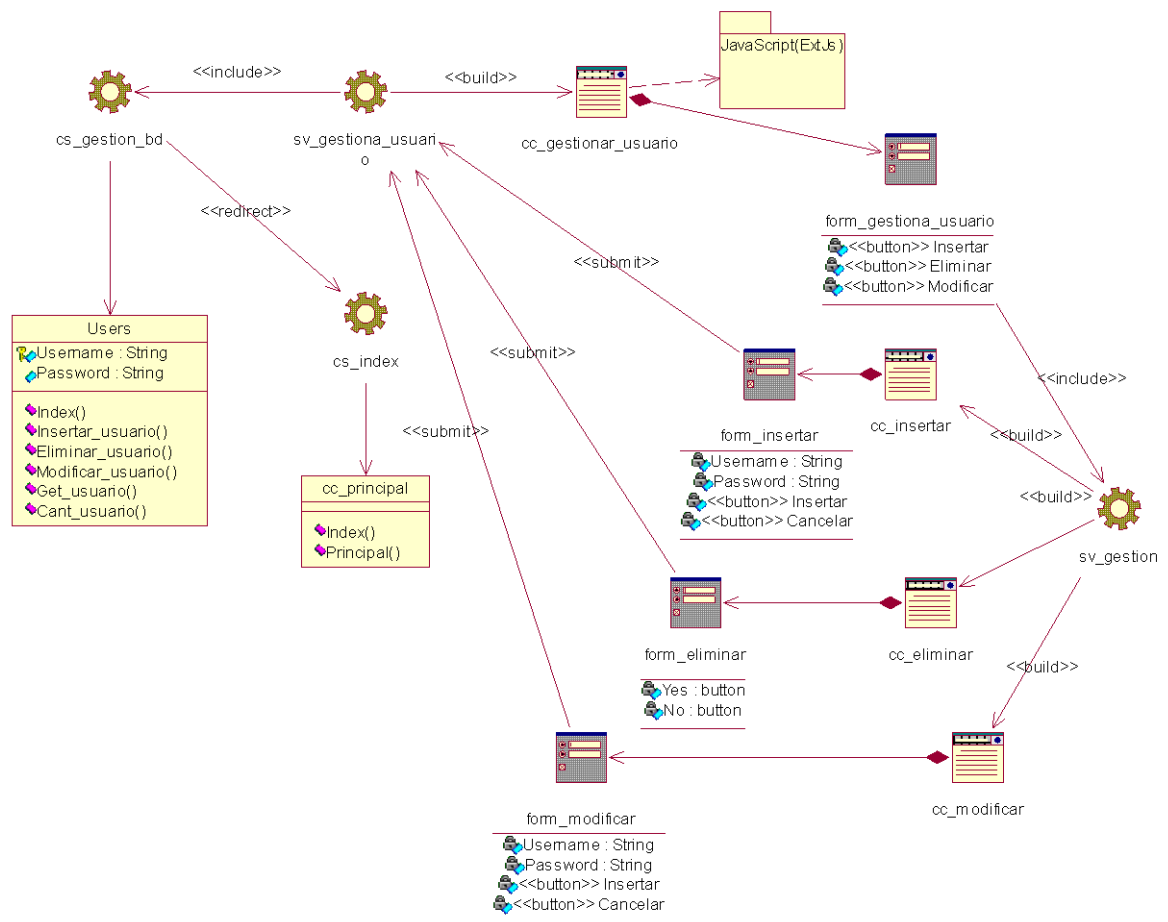
Anexo 7 Prototipo del caso de uso: Datos del correo



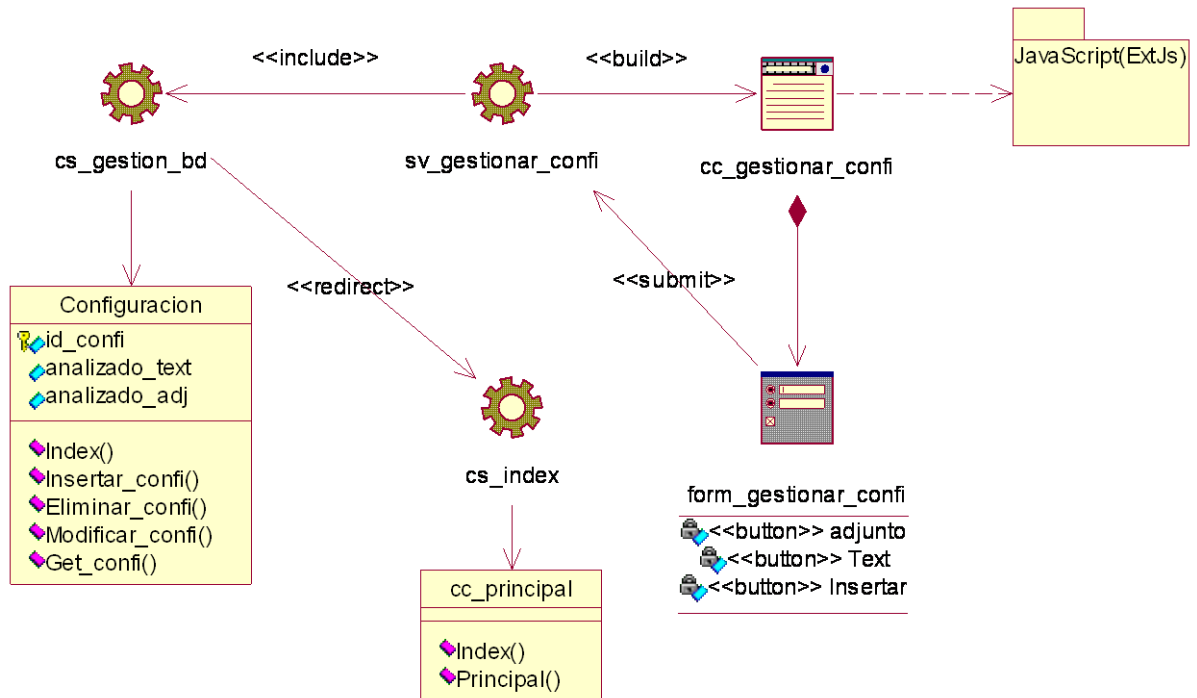
Anexo 8 Prototipo del caso de uso: Buscar



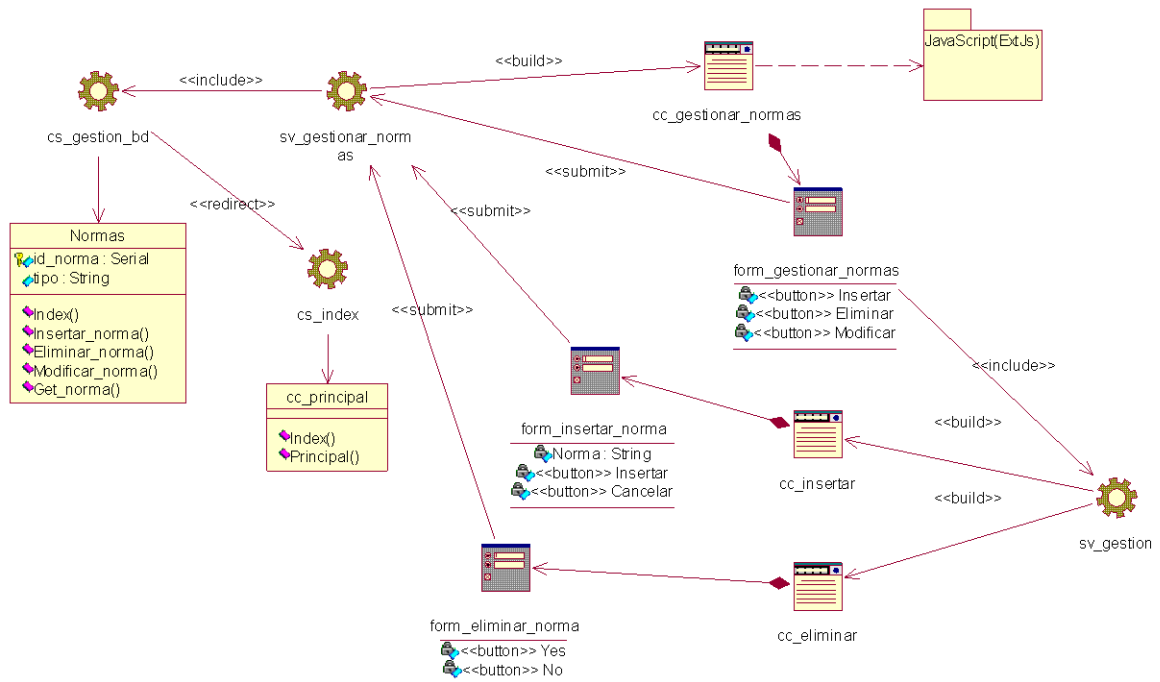
Anexo 10 Diagrama de clases: Gestionar url



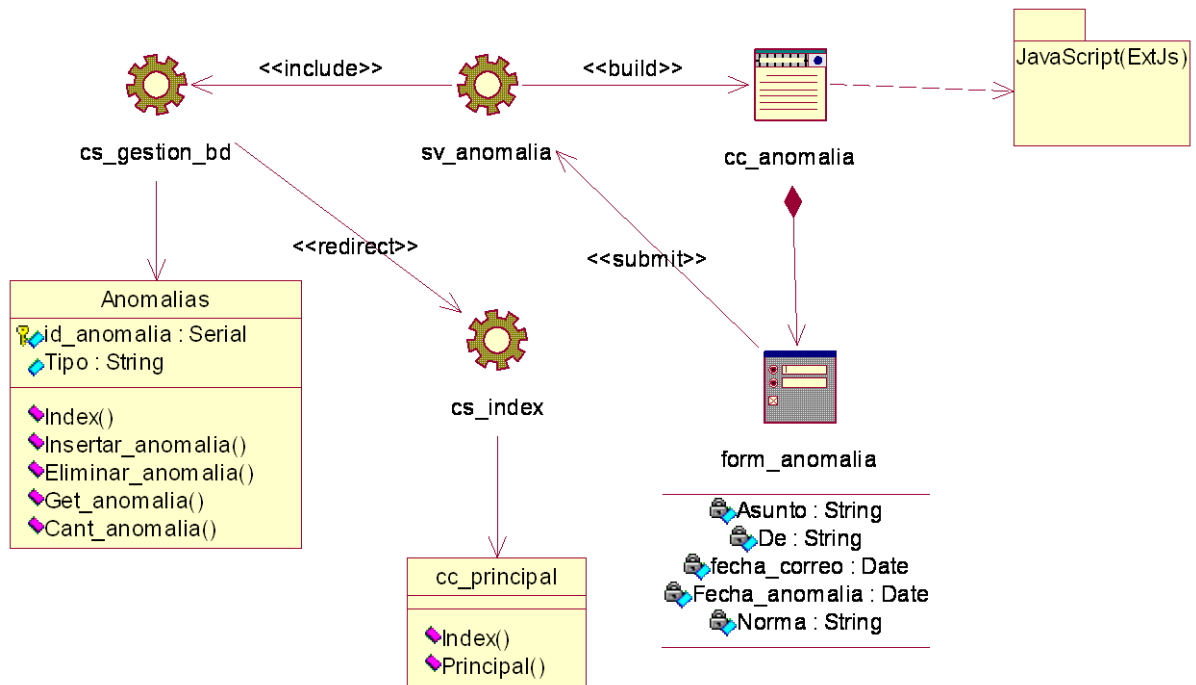
Anexo 11 Diagrama de clases: Gestionar Usuario



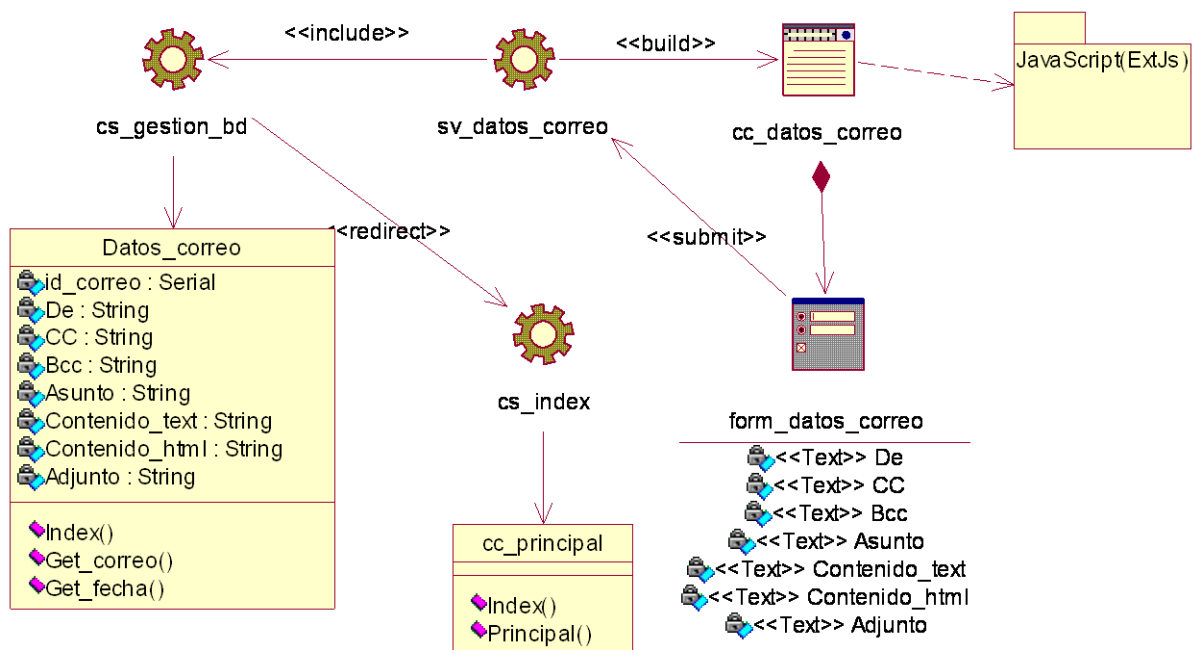
Anexo 12 Diagrama de clases: Gestionar Configuración



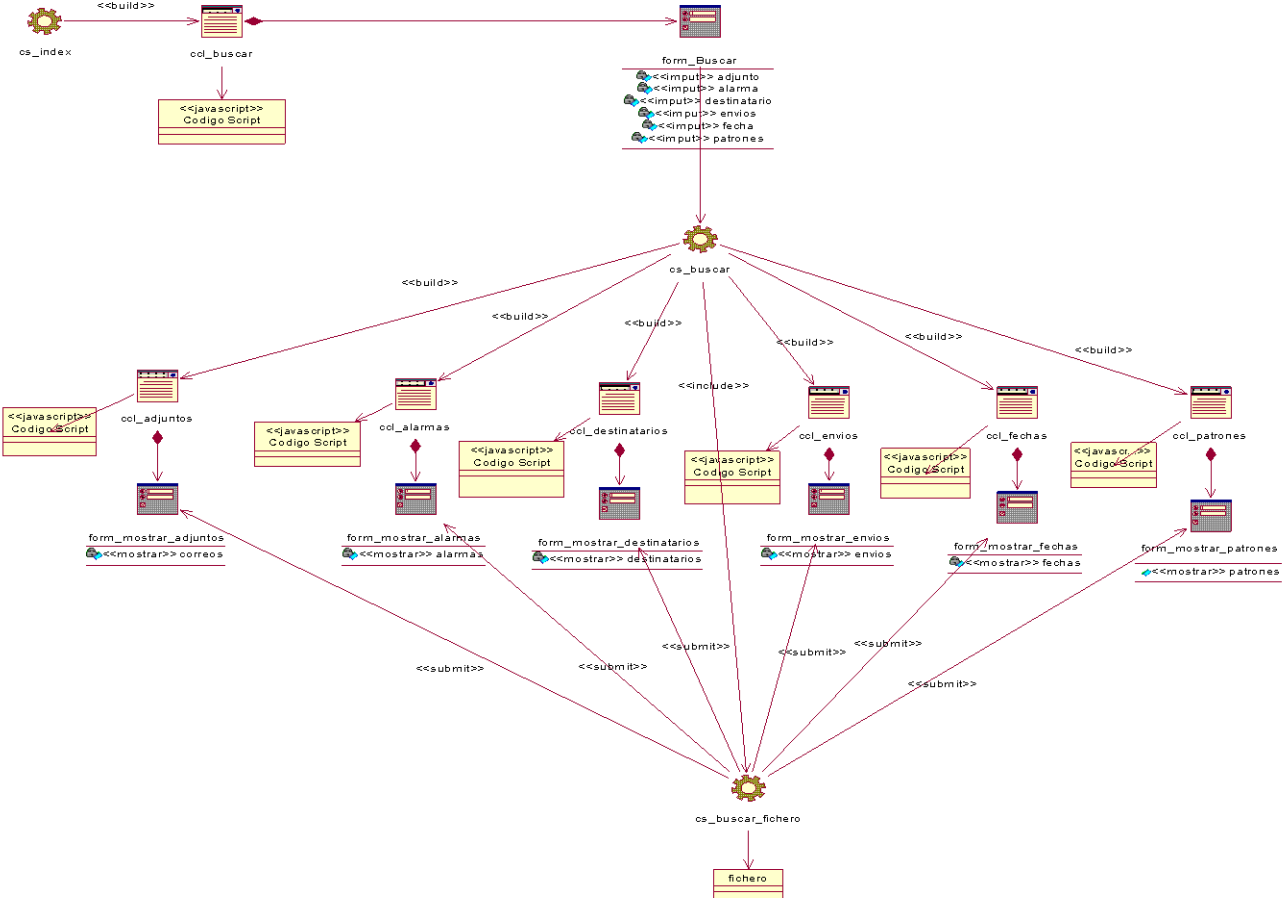
Anexo 13 Diagrama de clases: Gestionar Normas



Anexo 14 Diagrama de clases: Anomalia



Anexo 15 Diagrama de clases: Datos del correo seleccionado



Anexo 16 Diagrama de clases: Buscar