

**Universidad de las Ciencias Informáticas**

**Facultad 4**



**Título: Sistema de Reportes de la Navegación por  
Internet**

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

**Autores:** Luis Orlando Martín Alvarez  
Yassier García Martínez

**Tutores:** Ing. Raydel Montesino Perurena  
Ing. Pablo Yunier Medina Martínez

Junio de 2007

## **DECLARACIÓN DE AUTORÍA**

---

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo la presente a los 25 días del mes de Junio del año 2007.

Luis Orlando Martín Álvarez

Ing. Raydel Montesino Perurena

---

Firma del Autor

---

Firma del Tutor

Yassier García Martínez

Ing. Pablo Yunier Medina Martínez

---

Firma del Autor

---

Firma del Tutor

## **DATOS DE CONTACTO**

---

Ing. Raydel Montesino Perurena graduado de Ingeniero en Telecomunicaciones en el curso 2002-03.  
Director de Redes y Seguridad Informática de la UCI desde el 2005.

Ing. Pablo Yunier Medina Martínez graduado de Ingeniero en Telecomunicaciones en el curso 2002-03.  
Especialista General de Seguridad Informática desde el año 2005.

## **AGRADECIMIENTOS**

---

*El resultado obtenido en un trabajo de diploma es el fruto de muchas horas de investigación, sacrificio y dedicación; pero a la vez es un logro que no se puede llevar a solas sobre la espalda, sino que detrás de cada mensaje, cada idea que se refleja están los conocimientos, experiencias y consejos que aportaron otras personas. Es en esta pequeña sección que quiero agradecerles a todas ellas.*

*Primeramente quiero agradecerle a mi compañero de tesis, por su esfuerzo y dedicación, y reconocer que sin su ayuda esta investigación no hubiera terminado en tiempo. El tesoro más preciado que se puede tener son los amigos, a ellos de corazón les agradezco por el apoyo transmitido, por esas palmadas en la espalda que te llenan de espíritu. Gracias Adolfo por ser un ejemplo entre nosotros, a Adalberto y Hermes por sus granitos en este trabajo, a Martori por sus aportes en este sistema, negro sinceramente este sistema no funcionaría sin tu ayuda. A Greisy por los momentos de reflexión y por su amistad. Muchas gracias también a los especialistas del Dpto. de Seguridad Informática en especial a Mario por su ayuda imperecedera. Agradecido a mis tutores por su confianza y sus conocimientos, en especial a Raydel, por ser más que tutor mi amigo de siempre.*

*Eternamente agradecido a mi familia, en especial mi hermana querida, mi cuñado del alma y mí primo Michel por el apoyo cuando no estuvieron mis padres y a ellos gracias por guiarme por el camino correcto.*

**Luis Orlando.**

*La ardua escalada hasta la cima para ver realizados mis sueños a pesar de mi sacrificio no hubiese sido posible sin la ayuda de muchas personas y considero este momento preciso para agradecer a todos ellos.*

*Quiero agradecer a los especialistas del Departamento de Redes y Seguridad Informática de la Universidad en especial a mis tutores Pablo y Raydel por el apoyo brindado en todo momento, a mi compañero de tesis y amigo Luis por haberme dado la oportunidad de trabajar a su lado y haber podido compartir juntos esta experiencia inolvidable.*

*Agradecer a todos mis amigos y a mi familia, cada palabra de preocupación, la mano en el hombro, las fiestas (ron, dominó) en las que siempre se tocaba el tema de la escuela, fueron un compromiso para esforzarme cada día más. Un agradecimiento especial a mis padres por la educación que me han dado y por siempre guiarme por el camino correcto. A mi novia Meybis que es una persona muy especial en mi vida (TQLM) que me ha ayudado mucho no sólo en lo profesional sino en lo personal. A mis hermanos Yandy y Yosnet por haberme dado ánimo en los momentos difíciles.*

*Las gracias también a nuestro Comandante en Jefe por concebir esta Universidad y a la Universidad en si por haber marcado importantes momentos en mi vida.*

*Gracias...*

**Yassier.**

## **DEDICATORIA**

---

*A mis padres que lo han dado todo por el  
éxito de mi carrera profesional y a mi abuelo  
Fidencio que Dios lo tenga en la Gloria...*

**Luis Orlando Martín Alvarez.**

*... A mis padres y hermano...*

**Yassier García Martínez.**

## **RESUMEN**

---

En la Universidad de las Ciencias Informáticas ocurre con frecuencia que algunos usuarios se sienten afectados por un uso descontrolado de su cuenta de Internet, y se ven en la necesidad de reportarlo al Departamento de Seguridad Informática. En ocasiones los resultados del análisis de ese uso inadecuado de la navegación indican el acceso a sitios Web no permitidos por la Universidad. Se hace necesaria entonces la creación de un sistema al que los usuarios puedan acceder para obtener estadísticas de su navegación. Básicamente el sistema debería analizar los ficheros logs del servidor Proxy Squid de la Universidad para la generación de los reportes.

En el presente trabajo se incluye un estudio de los software analizadores de logs más utilizados en la actualidad. Además se describe la solución del sistema siguiendo los flujos de trabajo del Proceso Unificado de Rational. El sistema consiste en una aplicación Web en la que los usuarios pueden consultar las trazas de la navegación por Internet de forma eficiente y sencilla. A través de reportes estadísticos como el consumo por URLs, por días, por horas y por direcciones IP, el usuario podrá estar al tanto de lo que sucede realmente con su cuenta. Además algunas personas a través del sistema podrán supervisar la navegación de los usuarios que les fueron asignados. Las evidencias del acceso a los sitios Web en Internet pueden contribuir a disminuir el acceso a sitios restringidos y la efectividad del robo de contraseñas, que es uno de los incidentes de seguridad informática más problemáticos en la UCI.

El software desarrollado es multiplataforma y aunque ha sido realizado para su aplicación en la Universidad de las Ciencias Informáticas, es completamente adaptable a otras instituciones para el control del acceso a Internet.

## **PALABRAS CLAVE**

---

Analizador

Fichero Log

Framework

Internet

Reportes

Servidor Proxy

## **TABLA DE CONTENIDOS**

---

INTRODUCCIÓN .....	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA .....	3
1.1. Introducción .....	3
1.2. Software internacionales utilizados para el análisis de log de Servidores Proxy .....	4
1.2.1. Webalizer .....	4
1.2.2. Calamaris .....	6
1.2.3. Sawmil.....	7
1.2.4. WebTrends.....	8
1.3. Software analizadores de log desarrollados en Cuba .....	10
1.3.1. Isa Web .....	10
1.3.2. AAInternet .....	12
1.3.3. SICC .....	14
1.4. Software desarrollados en la Universidad .....	16
1.4.1. Sistema Adaptativo de Filtrado de Contenidos .....	16
1.5. Herramientas propuestas para el desarrollo del sistema .....	17
1.5.1. La tecnología Java .....	17
1.5.2. Frameworks utilizados en la Arquitectura .....	18
1.5.3. Sistemas Gestores de Base de Datos (SGDB).....	21
1.5.4. Lenguaje de modelación. UML (Unified Modeling Language) .....	22
1.6. Conclusiones.....	23
CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA .....	24
2.1. Introducción .....	24
2.2. Procesos de negocio en el Departamento de Seguridad Informática .....	24
2.2.1. Representación de los actores y trabajadores del negocio .....	26
2.2.2. Diagrama de casos de uso del negocio .....	28
2.2.3. Diagramas de actividades del negocio .....	28
2.2.4. Diagrama de clases del Modelo de Objetos .....	30
2.3. Propuesta del Sistema de Reportes de la Navegación por Internet.....	31

2.4. Especificación de los requisitos de software .....	32
2.4.1. Requerimientos no Funcionales .....	32
2.4.2. Requerimientos Funcionales .....	35
2.5. Definición de los actores del Sistema .....	35
2.6. Listado de los casos de uso del Sistema .....	36
2.7. Diagrama de casos de uso del Sistema .....	38
2.8. Casos de uso expandidos .....	39
2.8.1. Caso de uso CU_Autenticar .....	39
2.8.2. Caso de uso CU_Gestionar Reporte .....	40
2.8.3. Caso de uso CU_Gestionar Supervisión .....	43
2.8.4. Caso de uso CU_Supervisar Navegación.....	47
2.8.5. Caso de uso CU_Actualizar Información .....	48
2.9. Conclusiones.....	49
CAPÍTULO 3: ANÁLISIS Y DISEÑO .....	50
3.1. Introducción .....	50
3.2. Análisis.....	50
3.2.1. Modelo de clases de análisis.....	50
3.3. Diseño.....	52
3.3.1. Diagramas de secuencia .....	53
3.3.2. Diagramas de clases Web.....	57
3.3.3. Descripción de las clases .....	63
3.3.4. Diseño de la BD .....	82
3.4. Conclusiones.....	85
CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBA .....	86
4.1. Introducción .....	86
4.2. Implementación.....	86
4.2.1. Diagrama de Despliegue .....	86
4.2.2. Diagrama de Componentes.....	87
4.3. Modelo de Prueba.....	91
4.3.1. Descripción de los casos de prueba .....	91

4.4. Conclusiones.....	100
CONCLUSIONES .....	101
RECOMENDACIONES .....	102
REFERENCIAS BIBLIOGRÁFICAS.....	103
BIBLIOGRAFÍA.....	104
ANEXOS.....	105
Anexo 1: Formato de los ficheros logs del servidor Proxy Squid. ....	105
Anexo 2: Interfaz de autenticación del sistema. ....	105
Anexo 3: Interfaz para seleccionar rango de días de los reportes. ....	106
Anexo 4: Interfaz que muestra reportes de consumo por IP y consumo por día.....	106
Anexo 5: Interfaz que muestra reportes de consumo por Urls. ....	107
GLOSARIO .....	108

## **INTRODUCCIÓN**

---

En la Universidad de las Ciencias Informáticas se brinda poca información a los usuarios acerca de su navegación por Internet, básicamente solo la relacionada con el estado de las cuotas asignadas. Trabajadores y estudiantes no pueden informarse sobre cómo ha sido su navegación por Internet de forma detallada, en el momento que lo desean. En varias ocasiones se ha dado el caso de usuarios sancionados por navegar en sitios indebidos, que alegan no ser ellos los que cometieron la infracción y solicitan un esclarecimiento del problema (día en que ocurrió el incidente, desde que computadora, que sitios se visitaron). En otras palabras, los usuarios no pueden o no tienen argumentos instantáneos que les permitan determinar si otra persona está usando sus credenciales para la navegación por Internet. La vía para detectar este tipo de situaciones es solicitar un reporte a los responsables de seguridad informática, si existiera alguna sospecha. Además puede existir la creencia de que las trazas de la navegación no se registran, lo cual pudiera contribuir a violaciones del código de ética pensando que existe cierto descontrol.

Existe entonces la siguiente problemática: ¿Cómo lograr reducir el acceso a Sitios Web restringidos y hacer menos efectivo el robo de contraseñas en la Universidad de las Ciencias Informáticas? Esta interrogante lleva a formular la siguiente hipótesis para este trabajo:

- Si se desarrolla un sistema informático de reportes de la navegación por Internet para los usuarios de las tecnologías en la Universidad de las Ciencias Informáticas entonces se logrará reducir el acceso a sitios Web restringidos y se hará menos efectivo el robo de contraseñas.

Por tanto el objetivo general del trabajo es el siguiente:

- Desarrollar un sistema informático de reportes de la navegación por Internet para que contribuya a reducir el acceso a Sitios Web restringidos y hacer menos efectivo el robo de contraseñas en la Universidad de las Ciencias Informáticas.

Para llevar a cabo este objetivo es necesario realizar las siguientes tareas:

- Analizar el diseño de propuestas de otros desarrolladores para la resolución de este problema en otras instituciones.
- Definir las tecnologías informáticas a utilizar para desarrollar el sistema.

- Estudiar el formato de los ficheros log del servidor Proxy Squid para esclarecer la forma en que se almacena la información de la navegación por Internet y poder extraer la misma hacia una base de datos.
- Estudiar frameworks candidatos a utilizar en la arquitectura para el sistema de reportes que se desea implantar.
- Diseñar un modelo de datos relacional eficiente para almacenar el gran volumen de información contenida en los ficheros log.
- Definir la arquitectura del sistema a desarrollar y los patrones de diseño a seguir.
- Implementar una solución informática para mostrar a los usuarios reportes estadísticos de la navegación por Internet, que puedan ser filtrados dinámicamente.

Las tareas de investigación determinan la estructura de este trabajo el cual se divide en cuatro capítulos. En el capítulo uno se hace un análisis profundo de otras soluciones informáticas relacionadas con la revisión de trazas de la navegación por Internet y además se definen las tecnologías informáticas a utilizar. En el capítulo dos se hace un análisis de los procesos de negocios del Departamento de Seguridad Informática y se realiza una propuesta del sistema para la automatización de algunos de estos procesos. En el capítulo tres se realiza el análisis y diseño detallado de este sistema y finalmente en el capítulo cuatro se lleva a cabo la implementación de la solución propuesta.

## **CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA**

---

### **1.1. Introducción**

En los últimos años Internet ha tenido un profundo impacto en el mundo. Gracias a la red de redes millones de personas tienen acceso a una cantidad extensa y diversa de información en línea, el libre acceso a esta información puede ocasionar que los internautas se vean afectados por la calidad de la misma, además de que si no desarrollan habilidades para buscar dentro de la variedad de contenidos que se encuentran en la Web se puede perder mucho tiempo intentando encontrar lo que realmente interese. Internet ha dado un fuerte cambio en la vida de las personas, una de sus principales desventajas es que se está dejando de lado el contacto directo con la gente por estar más tiempo en el computador, ya que este sirve para todo, estudiar, realizar compras, conocer gente, escuchar música etc. Como Internet es libre las personas pueden navegar en sitios dedicados al juego, la pornografía o cualquier otro sitio de riesgo.

En el caso de las empresas se crea una problemática para los ejecutivos, por un lado las compañías tienen mucho que ganar con Internet como para eliminarlo del lugar de trabajo, pero a su vez hay mucho que perder al otorgar el acceso ilimitado a los empleados. El libre acceso a la red deja al empleado de una empresa la posibilidad de ver su correo personal, preparar sus vacaciones, leer el periódico o hacer compras online en horas de trabajo. Todo el tiempo que se utiliza en hacer estas “tareas” es tiempo que el empleado deja de trabajar, esto representa una pérdida de productividad. Para contrarrestar los problemas que suceden con el uso inadecuado de la Web es clave establecer algún criterio para el uso de la misma. No se trata de restringir ciento por ciento el acceso, sino tener políticas de uso.

Para obtener informes sobre el tráfico Web, se utilizan archivos log. El archivo log de un equipo captura un historial de las actividades que se desempeñan en el servidor. Graba qué documentos fueron servidos, y cuándo, así como también un sumario de información sobre los pedidos de los archivos. Esto último, incluye parámetros del usuario, tales como el referente (dónde estaba el visitante antes de llegar a cada página), la dirección IP, información de cookies que es enviada por medio del navegador, el estado de códigos, y el ancho de banda transferido.(YANOVER)

Estos ficheros log son muy ricos en información pero el problema está dado porque los datos se suelen almacenar en forma de un fichero de texto delimitado por comas, con abreviaturas y bastante críptico, lo

que dificultaría el entendimiento por parte de los usuarios que quieran observar las estadísticas de su navegación. Surgen entonces los software analizadores de log producto de la necesidad de desarrollar informes a fondo en poco tiempo y mostrarlos a los usuarios de una forma más detallada que les proporcione un mejor entendimiento.

En este capítulo se brinda información detallada de los software analizadores de log de Servidores Proxy que en la actualidad son más utilizados. Primeramente se ofrece una panorámica de los software más utilizados a nivel Internacional, haciendo énfasis en las características y procesos que resultan de interés para el desarrollo del trabajo. Posteriormente se presentan las propuestas que a nivel nacional han dado algunos desarrolladores para lograr un uso racional y controlado de Internet en el ámbito empresarial y se ofrece una explicación de cómo corroboran con el desarrollo del trabajo; y finalmente se presenta un Sistema de Filtrado de Contenidos desarrollado en la Universidad con el propósito de regular los contenidos inadecuados a los que son expuestos los usuarios en Internet. Además se hará un breve estudio acerca de la metodología, Sistema Gestor de base de Datos así como el lenguaje de programación propuesto para el desarrollo de este sistema.

### **1.2. Software internacionales utilizados para el análisis de log de Servidores Proxy**

Existen muchos software analizadores de log de Servidores Proxy en la actualidad. Producto de la gran diversidad que existe algunos brindan soluciones similares, mientras que otros difieren mucho en cuanto a sus funcionalidades, configuración, estándares de log soportados, reportes, etc. En las siguientes secciones se presentarán algunos software de carácter internacional. Dentro de las aplicaciones analizadoras de log más usadas por administradores de red podemos mencionar Webalizer, Calamaris, Sawmil, WebTrends, AWStats, Advanced Log Analyzer, Absolute Log Analyzer, Deep Log Analyzer, Log Analyze, Squid Report Generator, entre otros.

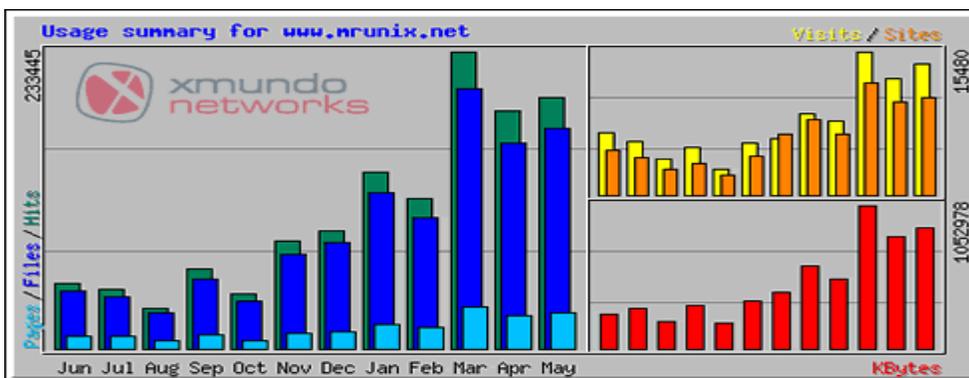
#### **1.2.1. Webalizer**

Webalizer es un pequeño programa que permite realizar análisis estadísticos del uso de Internet basándose en los log de servidores como Squid o Apache. Webalizer está desarrollado en C y es extremadamente rápido. Puede procesar un archivo con unas 150.000 entradas (40 MB) en aproximadamente quince segundos. Genera una gran diversidad de reportes analizando el formato de los

ficheros log (access.log) y seguidamente genera un archivo html (el cual es interpretado por el navegador) que es accedido desde la Web.

Gracias a esos reportes, puede observarse el número de personas que han entrado en la Web donde se vaya a instalar(o ejecutar) el Webalizer. Este programa no sólo nos da los reportes cuantitativos, si no que también nos da reportes gráficos, lo que hace más elegante y sencillo de observar las estadísticas de nuestra página Web. Entre otras cosas Webalizer es sumamente útil para saber que archivos son los que poseen más número de descargas en nuestra página Web.(MOTA 2006)

Permite obtener reportes mensuales únicos y comparativos, semanales, diarios, y conocer los horarios de mayor tráfico en su sitio Web, orígenes de procedencia de los usuarios, páginas de entrada y de salida, las palabras que los usuarios utilizaron para dar con su sitio Web en la red, páginas más visitadas y mucho más.(NETWORKS 2007)



**Figura 1.1 Resumen de consumo**

Este software a pesar de ser fácilmente configurable, con una gran diversidad de reportes no es completamente adaptable a las necesidades existentes de los usuarios de la universidad a pesar de ser usado en la misma por los especialistas de redes. Sus reportes pueden ser muy valiosos pero a un nivel de organización o entidad, o sea está orientado hacia empresas o especialistas y no hacia usuarios no especializados. En este caso esos reportes no son del todo eficientes para una persona, que le pudiera interesar más los sitios específicos que ha visitado y no estadísticas generales del comportamiento de la navegación en la Web; tales como los sitios mas visitados en el mes. A pesar de los indecoros mencionados anteriormente sus reportes con gráficos amigables para mostrar estadísticas de

comportamiento de la navegación, son una experiencia provechosa para el desarrollo de otros analizadores.

### **1.2.2. Calamaris**

Calamaris es muy adaptable a una extendida variedad de servidores Web y Proxy. Sus análisis exhiben informes picos de uso, métodos de peticiones, informes de estado de peticiones internas y externas, destinos de segundo y alto nivel, tipos de contenidos, y rendimiento.

Su integración con uno de los logs más usados y tediosos, los generados por Squid, lo pone a la vanguardia en la lista de los más usados conjuntamente con Webalizer. Es muy acogido por los administradores, pues sus reportes son muy valiosos para mejorar el servicio y detectar irregularidades.

La actual versión estable de Calamaris es capaz de generar tanto informes en texto plano (ideal para mandar por e-mail) como en HTML, con las estadísticas que nos interesen ordenadas por tablas. La nueva versión en desarrollo también genera gráficos de estas estadísticas. Aunque podemos usar Webalizer para este propósito, los informes de Calamaris son más sencillos de configurar y más detallados.(SUÁREZ).

La simple serie de tablas que Calamaris genera explican ciertos aspectos de consumo y rendimiento de la caché. Las secciones generadas son:

- Resumen
- Peticiones entrantes por métodos
- Peticiones-UDP entrante por estado
- Peticiones-TCP entrante por estado
- Peticiones salientes por estados
- Peticiones salientes por destino
- Destino de peticiones por dominio de segundo nivel
- Destino de peticiones por dominio de nivel superior
- Protocolo de petición TCP
- Tipos de contenido de las peticiones
- Peticiones UDP recibidas por computadora
- Peticiones TCP recibidas por computadora

- Rendimiento en intervalos de 60 minutos

Calamaris al igual que Webalizer puede ser muy útil para especialistas de la red, que constantemente combaten las vulnerabilidades de la misma. Sus reportes son muy generales, enfocados a la administración de la navegación; pero no están dirigidos a los usuarios no especializados, los cuales están más interesados por detalles específicos de su navegación.

### **1.2.3. Sawmill**

Sawmill es una potente herramienta de análisis de logs (registros de sucesos/eventos). Está especialmente diseñado para analizar logs de acceso a servidores Web, pero puede procesar casi cualquier log. Se ejecuta como un programa CGI en el servidor Web, y publica una intuitiva interfaz gráfica de usuario, que puede ser usada desde cualquier navegador, para configurar y ejecutar Sawmill, o para ver estadísticas de páginas. Las estadísticas son jerárquicas, atractivas, y llenas de enlaces que facilitan la navegación. El programa incluye una completa documentación. Sawmill ofrece una gran cantidad de opciones, incluyendo una base de datos persistente, control sobre la apariencia de las páginas de estadísticas, y opciones de filtrado sobre el log. (SOFTONIC 2007)

Permite la creación de diferentes perfiles en los que se especifica que información se presentará en el reporte. Es muy dinámico a la hora de filtrar la información procesada de los logs, lo cual permite extraer estadísticas que no pueden obtenerse con otros analizadores de logs.

Puede usarse en servidores de Windows y Linux, aunque sólo en estos últimos es posible integrarlos con el gestor de base de datos MySQL lo cual aumenta considerablemente el rendimiento para ficheros logs de gran tamaño y en la presentación de la información. En la versión de Windows solo puede usarse una base de datos propietaria del Sawmill que puede ser ineficiente para grandes volúmenes de logs.

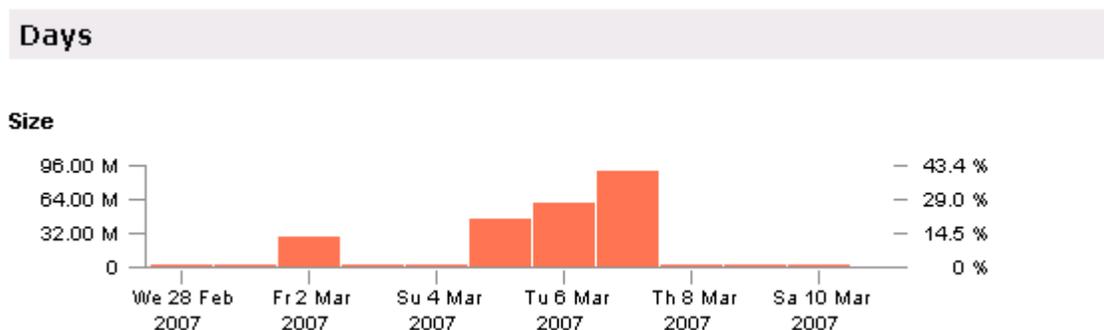


Figura 1.2 Resumen de consumo diario

Esta aplicación no es un buen partido para cualquier tipo de usuarios; solamente se puede considerar muy práctico para aquellos administradores que tienen acceso a los ficheros logs que contienen la información de la navegación por Internet. Actualmente este software es utilizado en el Departamento de Seguridad Informática de la Universidad. Para generar los reportes, Sawmil necesita analizar los logs; por esta razón no puede ser publicado en la Web para brindar un servicio de información a usuarios. Su integración con sistemas de base de datos se considera una buena elección para generar reportes eficientes en tiempo de respuesta. Mostrar reportes a través de la Web es una excelente solución a implementar en un analizador para mostrar las trazas de la navegación de los usuarios.

#### 1.2.4. WebTrends

El Analizador de Log de WebTrends, corresponde a una poderosa herramienta de medición y análisis de la actividad del Sitio Web en tiempo real, el cual permite obtener completos Reportes de Tráfico y Actividad del Sitio. WebTrends es una herramienta esencial para Administradores de Servidores Internet / Intranet, Webmasters, y Profesionales del Área de Marketing y Ventas de la compañía.(CYBERCENTER) Este software analiza los log de servidores Web y proporciona información valiosa referente al sitio Web y los usuarios que lo acceden. Su compatibilidad con los servidores se extiende por Microsoft IIS, Apache, CERN, NCSA, O'reilly, Lotus Domino, Oracle, Open Market, NetWare, y además reconoce más de 30 formatos de log.

Contiene entre sus funcionalidades el análisis de la segmentación, informes comparativos, guión personalizable, búsqueda de reportes y análisis de URL.

WebTrends ofrece un completo detalle gráfico del perfil del usuario que visita su Web, proporcionando información referente de las áreas de su Sitio Web más visitadas. Genera reportes HTML, que pueden ser visualizados por cualquier Navegador del sistema local o también desde Internet si el administrador lo considera. También exporta reportes en formato Microsoft Excel, PDF y texto. Sus reportes pueden ser configurados por tiempo e idioma.

WebTrends provee más de 70 tipos de tablas y gráficos diferentes:

- Estadísticas Generales
  - Número total de sesiones de usuario
  - Promedio diario de sesiones de usuario
  - Duración media de una sesión de usuario
- Trayectoria más frecuente a través del Sitio
- Archivos más transferidos o "bajados"
- Usuarios más autenticados países más activos
- Resumen de actividad diaria
  - Cantidad de usuarios en los días hábiles
  - Cantidad de usuarios en los fines de semana
  - El día más activo y el menos activo de la semana
  - El día más activo y el menos activo en la historia del sitio
- Organizaciones más activas
- Actividad por día de la semana
- Resumen de actividad por hora del día
- Análisis y estadísticas técnicas
- Actividad por tipo de organización

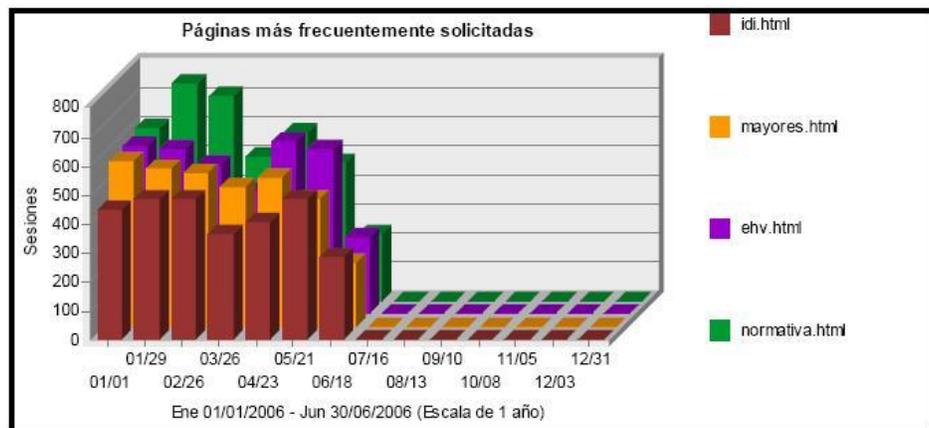


Figura 1.3 Páginas más visitadas en el Sitio Web

Este software está completamente especializado en todo tipo de reportes referentes a la navegación de un Sitio Web, a diferencia de los software mencionados anteriormente que controlan el tráfico referente a la navegación por Internet; es por esto que no puede emplearse en el análisis de los logs del servidor Squid de la Universidad. Sin embargo el formato de sus reportes y la manera en que los publica al servicio de los administradores puede ser tomado como base para el desarrollo de otros sistemas.

### 1.3. Software analizadores de log desarrollados en Cuba

En la sección anterior se hizo alusión a algunos de los software analizadores de log desarrollados internacionalmente. Algunas empresas cubanas han visto en ellos una alternativa para controlar el tráfico de la navegación por Internet. Sin embargo estos software no siempre se adaptan a las necesidades específicas de algunas instituciones y estas han tenido que desarrollar sus propios analizadores de log. En esta sección se exponen algunos de estos software desarrollados a nivel nacional.

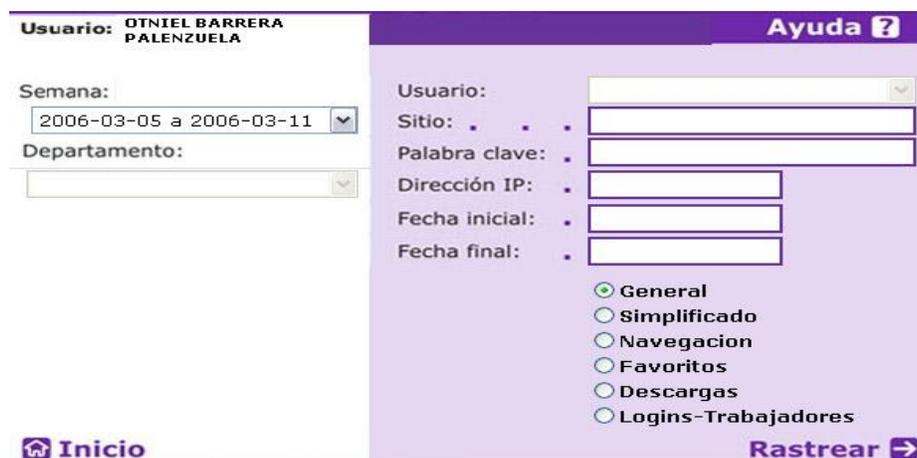
#### 1.3.1. Isa Web

Es una aplicación Web desarrollada en ASP.NET en el Instituto Central de Investigación Digital (ICID), permite que un usuario autenticado pueda visualizar información de su navegación en Internet. La aplicación verifica las credenciales del usuario a través de un servidor de "Directorio Activo" del dominio, usando el protocolo LDAP. Una vez que un usuario está autenticado puede filtrar información generada por su propio tráfico y el de sus subordinados.

Este software analiza los ficheros logs generados por servidores Proxy Isa Server. Solo la información relevante para los usuarios es extraída de estos ficheros y transformada en una base de datos desarrollada en SQL Server. En ISAWEB toda la información que se necesita es generada por otras aplicaciones en disímiles formatos. Para transformar toda esa información a bases de datos relacionales en SQL Server, se crearon aplicaciones de consola que se ejecutan como tareas de Windows en los servidores de Internet y de datos. (PALENZUELA and PÉREZ 2006)

La información es almacenada en trece tablas relacionadas con un trimestre, donde cada tabla contiene información de una semana a partir del momento en que se pone a funcionar el sistema.

La información que se desea ver puede ser filtrada por varios criterios: departamentos, usuarios, palabra clave, fecha, dirección IP, etc. El rango de días en que se quiere realizar la consulta es de carácter obligatorio, debido al gran volumen de información que puede tomar la base de datos.



The screenshot shows the ISAWEB application interface. At the top left, the user is identified as 'OTNIEL BARRERA PALENZUELA'. Below this, there are several filter options: 'Semana:' with a date range of '2006-03-05 a 2006-03-11', and 'Departamento:' with a dropdown menu. On the right side, there is a search form with fields for 'Usuario:', 'Sitio:', 'Palabra clave:', 'Dirección IP:', 'Fecha inicial:', and 'Fecha final:'. Below these fields are radio buttons for selecting the type of search: 'General' (selected), 'Simplificado', 'Navegacion', 'Favoritos', 'Descargas', and 'Logins-Trabajadores'. At the bottom left, there is a 'Inicio' button with a house icon, and at the bottom right, there is a 'Rastrear' button with a magnifying glass icon. A purple header bar at the top right contains the text 'Ayuda' and a question mark icon.

**Figura 1.4 Filtrado de información**

Esta aplicación parece brindar una solución completamente adaptable a las necesidades de los usuarios de la Universidad. La propuesta del sistema es eficiente, con autenticación de usuarios utilizando servicios LDAP, integración con base de datos que mejora la eficiencia en el procesamiento de grandes volúmenes de información, especificación de filtros para generar consultas; pero como se señaló anteriormente esta diseñado para analizar ficheros logs de un servidor Isa Server y no está preparado para analizar ficheros logs de un servidor Proxy Squid, que es el que actualmente se usa en la

Universidad. Por esta razón no podemos adoptar esta solución aunque sus funcionalidades pueden ser tomadas como referencia para el diseño de un nuevo sistema.

### **1.3.2. AAIInternet**

AAIInternet es un software analizador de log desarrollado por la empresa de Segurmática capaz de extraer adecuadamente la información de los ficheros logs de los servidores Proxy generando un reporte personalizado sobre la actividad del uso de Internet, donde puede observarse la fecha y hora de visita al sitio, la transferencia por usuarios y los dominios o sitios Web visitados, entre otros datos.

Sus versiones están desarrolladas para funcionar sobre Windows pero difieren en dependencia de los logs del servidor Proxy que está destinado a analizar. Tiene tres versiones, una para analizar la información de acceso generada por el servidor Proxy Squid, otra para IIS Server y la otra para Winroute.

AAIInternet utiliza una base de datos la cual es creada a partir del procesamiento y análisis de archivos (log files) generados por el servidor Proxy. De ahí que se hace necesario copiar manualmente hacia la estación donde está instalada esta herramienta los archivos logs que se deseen procesar o a una estación dentro de la red de su empresa donde sean accesibles estos archivos desde la estación donde se encuentra instalado AAIInternet. En el caso de servidores Proxy que funcionen sobre plataformas NO Windows podrán establecerse procedimientos FTP para ejecutar esta tarea. (SEGURMATICA 2007)

Su configuración permite refinar la información a importar a la base de datos, especificando los accesos a la Web que se desean a través de filtros.

AAIInternet maneja 2 tipos de Filtros:

- Tipos de Páginas Accedidas.
- Respuestas del servidor al acceder la página.

Ejemplo:

Si seleccionamos los filtros Música y Successful sólo importaremos los accesos a páginas asociadas con música (archivos MP3 por ejemplo) o audio, que hayan sido accedidas satisfactoriamente. (SEGURMATICA 2007)

Cuando se analizan los log la base de datos se encuentra lista para ofrecer información, AAIInternet admite realizar consultas para analizar el comportamiento de los usuarios en la Web. Algunas de estas consultas:

- Sitios Visitados por Usuario: muestra los sitios en la WEB visitados por cada usuario autorizado dentro de su empresa.
- Sitios Visitados por Estación (Dirección IP): permite conocer los sitios visitados por cada estación de trabajo, apoyándose en la dirección IP.
- Páginas Visitadas por Usuario: informa las páginas visitadas en la WEB por cada usuario autorizado dentro de su empresa.
- Direcciones IP NO Autenticadas: descubre las estaciones de trabajo dentro de su empresa que están haciendo uso del WEB sin control de autenticación.

Por otra parte, AAInternet se basa en la información contenida en la base de datos para construir reportes gráficos de análisis de la navegación de los usuarios en Internet. Ejemplo de ellos:

- Transferencia por Usuario: reporta aquellos usuarios que más suben y descargan archivos desde la Web.
- Transferencia por Dirección IP: reporta los ordenadores que más información mueven desde y hacia la Web.
- Sitios más Visitados: como su nombre lo indica lista aquellos sitios más visitados de la empresa.

### TRANSFERENCIAS POR DIRECCION IP (PRIMEROS CONSUMIDORES)

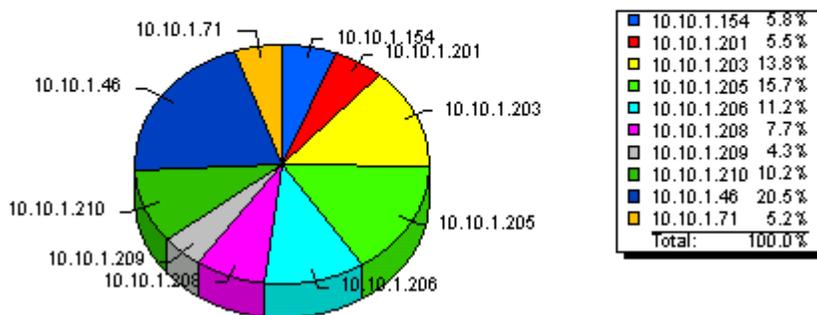


Figura 1.5 Transferencia por dirección IP

Como se ha descrito anteriormente AAInternet es una aplicación con flexibilidad en el filtrado de información a analizar contenida en los ficheros logs, que permite hacer consultas refinadas así como

reportes gráficos amigables para los usuarios. Al igual que algunos de los analizadores anteriores este crea una base de datos donde almacena la información extraída de los ficheros que será objeto de análisis. Aunque existe una versión para el servidor Proxy Squid (servidor configurado en la Universidad), sus reportes no pueden ser exportados a través de la Web; o sea que cada usuario tendría que instalarlo en su ordenador y luego contar con los log generados por el Proxy de navegación, información que esta muy protegida; por tanto no se considera una alternativa para cobijar un servicio de información de la navegación a los usuarios.

### **1.3.3. SICC**

La RED-CUJAE como muchas otras redes brinda un grupo de servicios. El Sistema Integrado de Contabilidad y Configuración (SICC) es un sistema modular y abierto, lo que significa que permite la incorporación de nuevas aplicaciones, a medida que se oferten nuevos servicios, sin necesidad de cambios estructurales. Esta flexibilidad se debe a la arquitectura establecida. El sistema cuenta con dos servidores Squid que controlan el acceso a la red nacional e internacional. El consumo de la navegación por Internet extraído de los logs generados por los servidores es traducido a unidades monetarias; que representan las unidades de la cuota controlada en el sistema de contabilidad. La conversión es realizada en dependencia de la clasificación del sitio al que se accede y el horario de acceso al mismo. Los resúmenes de la información contable generada luego de haberse filtrado y contabilizado son enviados al servidor de contabilidad, el cual contiene una base de datos centralizada en PostgreSQL. Por otro lado los detalles de la navegación son conservados en ficheros logs del SICC. Esta información es construida por el Retbill, un software encargado de generar información contable desde los logs del Squid. El consumo de la navegación por Internet obtenida de los servidores Proxy no se desecha, sino que esta también es almacenada en una estructura de ficheros serializados en Python; los cuales serán explotados por otros sistemas. Básicamente este es el funcionamiento del SICC.

Este sistema es aplicado en diferentes entornos, donde la autenticación puede realizarse de diversas formas, por ejemplo sobre la base de datos del SICC o sobre un directorio LDAP. Una vez autenticados los usuarios pueden obtener un resumen del consumo en unidades monetarias de su navegación. Esta información es extraída del servidor de contabilidad y es visualizada en forma de barras. En caso de que los usuarios soliciten información detallada, esta se obtendrá de los ficheros logs del SICC, que contienen

los detalles. El sistema muestra reportes estadísticos de consumo en unidades monetarias y el equivalente en bytes por los sitios accedidos y por días de navegación.

Debido a la arquitectura que ya se explicó existe un módulo de otro sistema que permite supervisar la navegación de los usuarios. El sistema en cuestión es el Topint, el mismo permite obtener los hits de navegación por usuario, este sistema se apoya en los ficheros serializados en Python para extraer la información necesaria. El reporte que se desea obtener se filtra inicialmente por áreas de la Universidad; como por ejemplo las facultades y seguidamente se selecciona el mes del año que se desea supervisar. Las estadísticas obtenidas son la cantidad de hits y el consumo general en bytes por cada usuario. Además se pueden visualizar hits de cada usuario por dominios visitados y por IP.

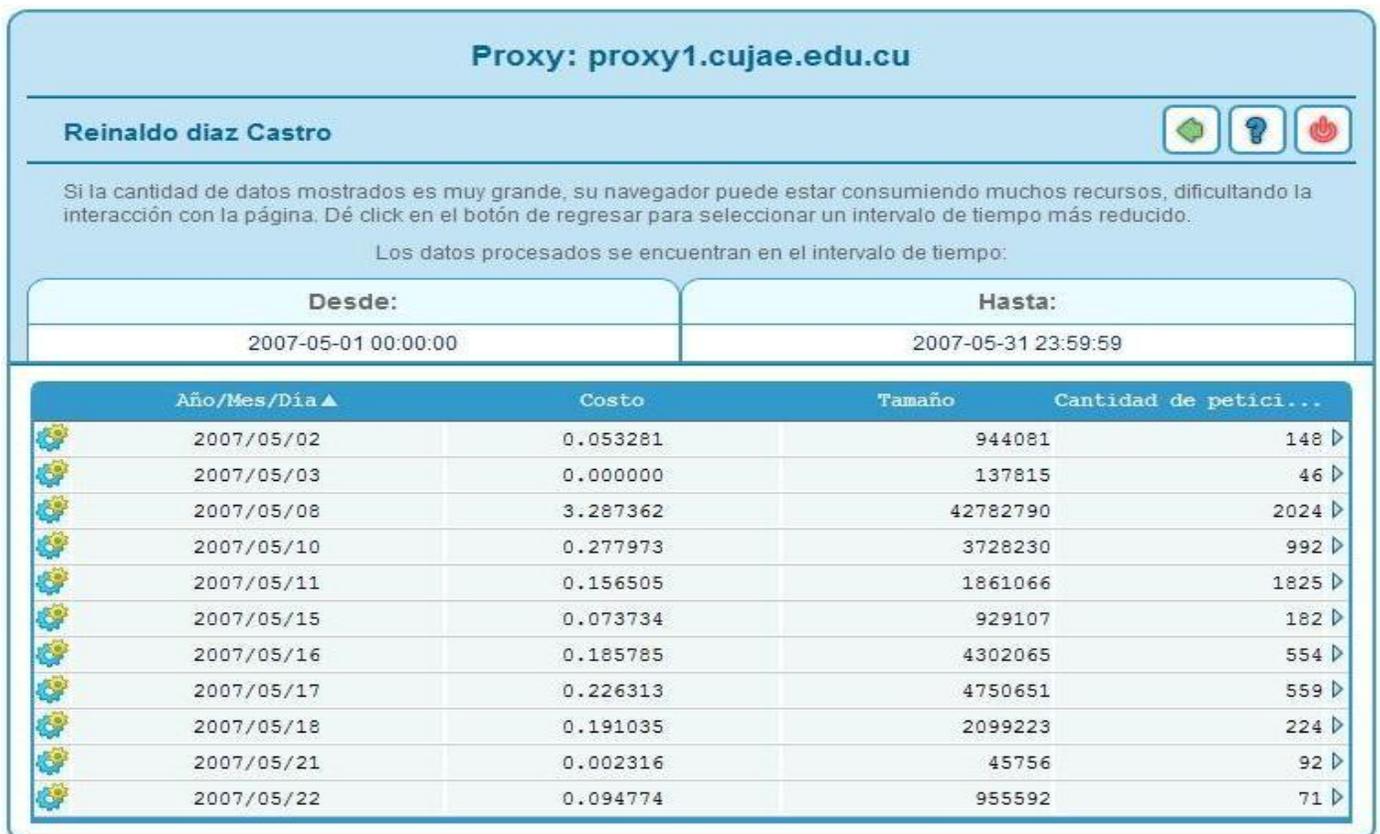


Figura 1.6 Tabla de consumo en unidades monetarias por días

Como se puede observar el SICC tiene una arquitectura estable y flexible a cambios que puede ser aprovechada para futuros sistemas. Sus reportes pueden ser objetos de estudio para el desarrollo de

nuevos sistemas en otras instituciones; pero son completamente dependientes de su arquitectura, debido a la estrategia de conversión del consumo a unidades monetarias establecida en el mismo.

### **1.4. Software desarrollados en la Universidad**

Internet es una rica fuente de conocimiento pero a su vez expone un serio problema por la presencia de contenidos nocivos e ilícitos; como consecuencia en la Universidad de las Ciencias Informáticas se han estado buscando soluciones para regular un tanto que contenidos son accesibles por los usuarios. En esta sección se hace referencia a un sistema que está en proceso de desarrollo en la Universidad, el mismo posee funcionalidades básicas que hacen más seguro el acceso a Internet.

#### **1.4.1. Sistema Adaptativo de Filtrado de Contenidos**

El principio de funcionamiento de este Sistema de Filtrado Adaptativo es basado en el uso de listas blancas y negras, con las listas blancas se persigue el objetivo de permitir el acceso a los recursos que estén contenidos en ellas y con las listas negras se persigue el objetivo de negar el acceso a los recursos que estén contenidos en ellas. El sistema dispone de una Base de Datos en la cual se encuentran infinidad de URLs que están preclasificadas en diferentes categorías atendiendo al contenido que albergan. Las categorías también se encuentran clasificadas por el administrador del sistema según el tipo de contenidos al que se estima pertenecen y las características propias del lugar. Prevee además en un futuro la inclusión de un subsistema de Recuperación y Clasificación Automática de Información. El servidor Proxy Squid es parte inseparable del sistema desde el momento en punto en que un cliente solicita un recurso de Internet al servidor, en ese momento el servidor le entrega al Subsistema de Filtrado los parámetros de la solicitud: identificador del usuario, dirección IP de origen y URL, comenzando así un proceso de interacción con la Base de Datos el cuál va a ser determinante en la decisión que se tome sobre si al usuario se le entrega el recurso solicitado o se le muestra una página en la cual se le expliquen los motivos por los cuales se le negó el acceso al recurso.

Aunque este sistema como se explica en su funcionamiento no analiza los ficheros log del Squid sino que más bien bloquea el acceso a algunas direcciones de sitios clasificados por los administradores, es el software desarrollado en la Universidad que hasta el momento, nos toca de cerca en cuanto al control de la navegación. Queda claro que su funcionamiento no brinda ninguna estadística del comportamiento de la

navegación de los usuarios; esta es la razón por la que el problema que afecta a los usuarios en cuanto a la desinformación de su propia navegación sigue estando vigente.

### **1.5. Herramientas propuestas para el desarrollo del sistema**

Para el desarrollo del sistema se realizó un estudio de las tendencias y tecnologías actuales para buscar las posibles herramientas a utilizar en su construcción. Teniendo en cuenta el estudio realizado se hace la siguiente propuesta.

#### **1.5.1. La tecnología Java**

Para el desarrollo del trabajo se escogió la tecnología Java porque es una tecnología madura, muy eficaz y sorprendentemente versátil, se ha convertido en un recurso inestimable ya que permite a los desarrolladores crear software en una plataforma y ejecutarlo en prácticamente cualquier otra plataforma, posibilita además crear programas para que funcionen en un navegador Web y en servicios Web. La tecnología Java se ha convertido en la tecnología ideal para aplicación a redes por su versatilidad y eficiencia, la portabilidad de su plataforma y la seguridad que aporta.

La tecnología Java está compuesta por dos partes: El lenguaje de programación, y la plataforma.

##### **1.5.1.1. La plataforma**

Las plataformas son descritas como la combinación del hardware y el software que hacen funcionar a un ordenador. La diferencia de la plataforma Java está dada porque es una plataforma sólo de software y se ejecuta sobre las otras plataformas de hardware. La plataforma Java tiene dos componentes: La máquina virtual de Java (JVM) y el API Java (Application Programming Interface). La máquina virtual de Java es la base de la plataforma Java, es llevada a diferentes plataformas de hardware. Esta es una idea revolucionaria en el mundo de la programación. Básicamente un programa desarrollado en java cuando es compilado es convertido a un código neutro, el cual es interpretado por esta excelente máquina convirtiéndolo a código particular de la CPU utilizada. Esto justifica lo que se ha convertido en el principal lema del lenguaje: "Escríbelo una vez, corre donde quiera". Para los programadores el API de Java le proporciona muchas utilidades por la gran colección de componentes de software que le brinda.

Java incorpora muchos aspectos que en cualquier otro lenguaje son extensiones propiedad de empresas de software o fabricantes de ordenadores (threads, ejecución remota, componentes, seguridad, acceso a bases de datos, etc.).(JALÓN 1999)

### **1.5.1.2. Lenguaje de programación**

La compañía Sun describe el lenguaje Java como “simple, orientado a objetos, distribuido, interpretado, robusto, seguro, de arquitectura neutra, portable, de altas prestaciones, multitarea y dinámico”.(JALÓN 1999)

La definición anterior manifiesta un lenguaje de programación de alto nivel diseñado para que un programa escrito en él, sea ejecutado independientemente de la plataforma (hardware, software y sistema operativo) en la que se esté actuando. Dentro de sus características más importantes podemos mencionar además de que es independiente de la plataforma, que es un lenguaje ideal para Internet por lo integrado que tiene el protocolo TCP/IP. Java surgió entonces como el lenguaje de programación para entorno de red por excelencia. Esto explica porque java fue pensado para ser seguro desde sus principios. Los programas escritos en Java no pueden ser atacados por virus, pues para que estos tengan efecto deben utilizar rutinas de acceso directo a memoria, que Java no tiene.(BECERRIL 1998)

Otra de las ventajas que encontramos en el lenguaje es la capacidad de realizar varias tareas a la vez; esto significa que un programa puede dividirse en varios fragmentos para ser más eficiente en realizar su tarea.

### **1.5.2. Frameworks utilizados en la Arquitectura**

Un Framework es una estructura de soporte definida en la cual otro proyecto de software puede ser organizado y desarrollado, incluye soporte de programas y librerías que ayudan a desarrollar e integrar los diferentes componentes de una aplicación.

Un Framework representa una Arquitectura de Software que modela las relaciones generales de las entidades del dominio. Provee una infraestructura y "manera de trabajo" la cual extiende y/o utilizan las aplicaciones del dominio.(INFORMÁTICA 2006)

### **1.5.2.1. Spring**

Spring es un framework de código abierto dividido en siete módulos bien definidos, donde cada cual se identifica por sus propias ventajas. No necesariamente hay que estar ligado a cada uno de ellos, los desarrolladores son libres de usar aquellos módulos que le sean necesarios para el desarrollo de una aplicación. Este es uno de los proyectos más sorprendentes en el panorama actual de Java por la medida en que ayuda a que los diferentes componentes que forman una aplicación trabajen entre sí, este framework de Java tiene un conjunto de librerías de las cuales se pueden escoger aquellas que facilitan la creación de aplicaciones. Precisamente este fue el objetivo específico desde su creación: hacer el desarrollo de aplicaciones J2EE más fácil. Dentro de sus potencialidades está la Inversión de Control (Inversión de Control, también llamado Inyección de Dependencias, que es una técnica alternativa a las clásicas búsquedas de recursos vía Interfaz de Nombrado y Directorio Java (JNDI)) perteneciente al modulo central de Spring. Cuando la Inversión de Control es aplicada los objetos son asignados pasivamente a sus dependencias en lugar de ser creados o buscar por sus dependencias. Spring, como contenedor al fin, permite administrar el ciclo de vida de los objetos de su sistema; cómo son creados, si se crea una instancia nueva por cada uno en el momento que son necesitados o se comparte siempre la misma instancia. Otra de sus ventajas es el soporte integrado para la programación orientada a aspectos contenido en el modulo de AOP, la cual puede ser muy útil en la administración de transacciones. Este framework hace posible crear y configurar aplicaciones complejas a través de componentes sencillos. En Spring todas estas ventajas pueden ser explotadas debido a que los objetos de la aplicación son creados de forma declarativa, típicamente en un fichero XML, permitiendo definir en él las dependencias entre los mismos. Uno de sus módulos es quien lo convierte realmente en framework, el módulo del Contexto de la Aplicación. El mismo extiende las funcionalidades del modulo principal, agregando soporte para internacionalización, correo, integración con EJB y tareas programadas. También el modulo MVC Framework integra una completa implementación del patrón MVC para construir aplicaciones Web. Además puede ser integrado con otros MVC frameworks como Struts. No por ser mencionado de último deja de ser significativa la capacidad que muestra el modulo ORM de Spring, permitiendo la integración con otros frameworks de persistencia; por ejemplo uno de los más usados es Hibernate, muy útil en el acceso a datos de una aplicación. De este último se hablará con más detalles en una de las siguientes secciones.

### **1.5.2.2. Acegi**

Acegi es uno de los mejores frameworks de seguridad existentes, potente, versátil, flexible, y a la vez sencillo de configurar y está diseñado fundamentalmente para ser usado con Spring. Este framework ofrece una configuración de la seguridad para aplicaciones de forma declarativa. Además su integración con el framework de Spring le permite ofrecer una colección de clases que pueden ser configuradas dentro del propio contexto de la aplicación tomando las ventajas que ofrecen algunos módulos de Spring; como son la Inversión de Control y la programación orientada a aspectos. Para que una aplicación pueda considerarse asegurada se tienen que garantizar la autenticación de los usuarios y la autorización de los accesos a determinados recursos del sistema, aspectos que pueden ser tratados por Acegi. Cuando las aplicaciones Web son aseguradas, Acegi usa filtros que interceptan las peticiones para llevar a cabo la autenticación y asegurar el acceso a los recursos del sistema. Este framework proporciona una librería de etiquetas muy fácil de usar, para proteger las partes de una página Java Server Pages (JSP) que no se desea que puedan ser visualizadas por determinados usuarios. Acegi proporciona una capa que envuelve diversos estándares de seguridad presentes en Java y ofrece una forma unificada de configuración a través de un descriptor en XML. Este framework ha sido elegido como opción frente a los sistemas propietarios de los diferentes vendedores, por su universalidad de uso no es necesario cambiar nada si se cambia de proveedor en los servidores, así como por su potencia, que engloba las API's de seguridad de Java.

### **1.5.2.3. Hibernate**

Hibernate es un framework de persistencia código abierto de alto rendimiento que esta ganando una popularidad significativa en estos tiempos. Es una herramienta ORM (Mapeo objeto-relacional) completa, es un framework de persistencia que ofrece una alternativa a la persistencia gestionada por el contenedor (Container Managed Persistence o CMP) de EJB. Hibernate configura como mapear objetos de una aplicación con una base de datos relacional a través de ficheros XML de configuración. Además del Mapeo objeto-relacional, Hibernate suministra otras características sofisticadas que se podrían esperar de una herramienta ORM completa, tales como cache, cache distribuida y carga perezosa.

La arquitectura de caché de Hibernate es muy potente. Hibernate, ofrece dos niveles de caché, sólo el primero de ellos es obligatorio. Por si fuese poco, Hibernate incluye una caché de consultas que ofrece la posibilidad de obtener rápidamente resultados que ya habían sido consultados previamente.

Hibernate ofrece un lenguaje de consultas que agrupa un potente y flexible mecanismo de consulta, almacenamiento, actualización y recuperación de objetos desde una base de datos. Este lenguaje, el Hibernate Query Language (HQL), es una extensión orientada a objetos de SQL. HQL permite acceder a los datos de varias formas, incluyendo consultas orientadas a objetos. (PALOS 2007)

Hibernate se integra dentro de los servicios de una plataforma J2EE siendo capaz de obtener conexiones a través de objetos DataSource vía JNDI (Java Naming and Directory Interface), ejecutar sus transacciones dentro de un entorno JTA (Java Transaction API), etc. Aparte de esto ofrece también vía de integración con otras tecnologías, por ejemplo JMX (Java Management Extensions) y JCA (Java Cryptography Architecture). (INFORMÁTICA 2006)

### 1.5.3. Sistemas Gestores de Base de Datos (SGDB)

Numerosos sistemas gestores de bases de datos existen en la actualidad, entre ellos el Microsoft SQL, Oracle, MySQL, Visual Fox Pro, etc., sin embargo se seleccionó el PostgreSQL por ser un gestor de base de datos libre que ofrece considerables ventajas.

#### 1.5.3.1. PostgreSQL

PostgreSQL es un servidor de base de datos altamente sofisticado, posee un alto rendimiento, estable y capacitado para lidiar con grandes volúmenes de datos. Cada vez más las empresas eligen este sistema gestor de base de datos, debido al hecho de que, para que pagar por una base de datos propietaria si existe una que es altamente eficiente y que no requiere licencia de uso.

Las funcionalidades implementadas por PostgreSQL impresionan tanto por su amplitud como por su complejidad. Como las funcionalidades técnicas ofrecidas por el PostgreSQL son bastantes extensas, citamos aquí solamente las principales:

- Base de datos objeto-relacional (ORDBMS)
- Soporte a transacciones
- Bloqueo a nivel de registro
- Integridad referencial
- Número ilimitado de registros e índices en tablas
- Soporte para GIS (base de datos geo-referenciales)
- Acceso vía drivers ODBC y JDBC

- Interfase de administración gráfica
- Uso optimizado de recursos del sistema operativo
- Soporte a estándares ANSI SQL 92 y 99
- Joins: Implementa todos los tipos de join definidos por el estándar SQL99: inner join, left, right, full outer join, natural join
- Disparadores (triggers), Vistas (views) y procedimientos almacenados
- Soporte para almacenamiento de BLOBs (binary large objects)
- Consultas y sub-consultas definidas en cláusula FROM
- Backup online
- Sofisticado mecanismo de tuning
- Soporte a conexiones de base de datos seguras (criptografía)
- Modelo de seguridad para acceso a objetos de base de datos por usuarios y grupos de usuarios
- Recuperación Point-In-Time que en forma automática y continua genera backups del servidor (versión 8)
- Tablespace (versión 8) (HTTP-PERU 2006)

### 1.5.4. Lenguaje de modelación. UML (Unified Modeling Language)

El Lenguaje de Modelación Unificado (UML - Unified Modeling Language) es un lenguaje usado para especificar, visualizar y documentar los componentes de un sistema en desarrollo orientado a objetos. UML es una notación y no un proceso/método, es decir, es una herramienta útil para representar los modelos del sistema en desarrollo, mas no ofrece ningún tipo de guía o criterios acerca de cómo obtener esos modelos.(VERA 2006)

UML capta la información sobre la estructura estática y el comportamiento dinámico de un sistema, es importante aclarar que UML no es un lenguaje de programación sino que las herramientas pueden ofrecer generadores de código de UML para una gran variedad de lenguajes de programación, así como construir modelos por ingeniería inversa a partir de programas existentes.

Como prácticamente todas las herramientas CASE y de desarrollo lo han adaptado como lenguaje de modelado, cuando se desea adquirir una herramienta pueden evaluarse varias alternativas como por ejemplo Rational Rose, Enterprise Architect, Poseidón y Visual Paradigm for UML, el trabajo se desarrolla con Visual Paradigm for UML porque es una herramienta poderosa que se integra con Eclipse.

El Proceso Unificado de Rational (RUP Rational Unified Process) y UML están muy relacionados, mientras RUP establece las actividades y los criterios para conducir un sistema desde su máximo nivel de abstracción hasta su nivel más concreto, UML ofrece la notación gráfica necesaria para representar los sucesivos modelos que se obtienen en el proceso de refinamiento, juntos constituyen la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos.

### **1.6. Conclusiones**

Como se ha estado precisando en las secciones anteriores existe una gran variedad de software desarrollados para el análisis de logs de servidores Proxy. Como resultado del estudio de su funcionamiento muchos han mostrado soluciones geniales para especialistas de seguridad que controlan el tráfico de la navegación de una institución; pero no se adaptan a las necesidades específicas de los usuarios o algunos poseen requerimientos imposibles de cumplir según las políticas de seguridad de la información implementadas en la Universidad; por ejemplo prohibir el libre acceso a los log generados por el servidor Proxy Squid. Dada estas condiciones, se está desarrollando un software tomando como base las ventajas de estos analizadores; pero que a la vez brinde las flexibilidades que muchos de estos no han sido capaces de congregar. El sistema permitirá el acceso por cada usuario a los reportes específicos del comportamiento de su navegación por Internet a través de la Web. Sin embargo los usuarios no tendrán que facilitarles los log generados por el servidor Proxy Squid para que sean analizados, sino que esa tarea se iniciará de forma automática y se almacenará en una base de datos para que sea explotada en la generación de reportes por esta aplicación. Apoyándose en el software IsaWeb algunos usuarios contarán con privilegios para poder acceder a la navegación de otros. La información que se procesará será solo la necesaria para satisfacer las necesidades de los usuarios, así el volumen de información podría disminuir y los tiempos de respuestas a las peticiones de los usuarios pueden ser más eficientes.

## **CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA**

---

### **2.1. Introducción**

Para dar comienzo con el proceso de desarrollo de software lo primero que hay que hacer es alcanzar cierto nivel de conocimientos sobre el problema en cuestión para determinar si es factible seguir adelante con el proyecto. En este capítulo se realiza la propuesta del sistema explicando previamente los procesos del negocio, se definen los casos de uso así como los actores que interactúan con los mismos, se analizan además los requerimientos funcionales y no funcionales describiendo de esta forma las funcionalidades que debe tener el sistema así como las propiedades o cualidades que debe tener el producto.

### **2.2. Procesos de negocio en el Departamento de Seguridad Informática**

En la Universidad de las Ciencias Informáticas (UCI) se brindan servicios de acceso a Internet, los cuales son explotados por los estudiantes y trabajadores de la misma considerando la valiosa fuente de información que esta representa. Para garantizar el control del tráfico de la navegación por este medio se utiliza un servidor Proxy Squid a través del cuál realmente es que se consigue el acceso a la red de redes mencionada.

El Departamento de Seguridad Informática de la Universidad tiene como objetivo mantener el buen uso de las tecnologías de la información y las comunicaciones (TIC). En este departamento se hacen análisis de los logs de la navegación generados por el servidor Proxy. Los logs contienen registros de información valiosa para el análisis del comportamiento de la navegación por Internet: como son la hora en que se realizó la conexión, el IP del ordenador que realizó la conexión, el tamaño de la descarga realizada, la dirección de la URL visitada, el tipo de información a la que se accedió, etc. [Ver Anexo 1.](#)

Este proceso se considera ya una ventaja en áreas de controlar aquellos usuarios que rompen con el código de ética de la Universidad. Las violaciones que frecuentemente aparecen son el robo de contraseñas y el acceso a sitios restringidos (pornográficos, contrarrevolucionarios, etc.) por la Universidad.

En ocasiones los usuarios se encuentran afectados por un descontrolado consumo de su cuota de Internet, que pudiera sospecharse un robo de su contraseña, ante esta situación pueden hacer un reporte

del incidente ocurrido. Estos reportes pueden conducirse por varios caminos que se explicarán en detalles. Las personas que se ven afectadas por un robo de su contraseña podrían registrar el incidente en el sitio de seguridad informática de la Universidad, enviarlo por correo electrónico a la dirección [seguridad.informatica@uci.cu](mailto:seguridad.informatica@uci.cu) o pueden presentarse en el departamento de seguridad informática de la Universidad donde son atendidos por los especialistas en la materia. Una vez que son reportados los detalles y conocido el usuario afectado los especialistas proceden a investigar el incidente. Para la generación de los reportes de la navegación por Internet del usuario los especialistas utilizan un software al cual se le envía el usuario que solicitó el análisis y un rango de días (generalmente un mes) y se obtiene la información con la que el Sawmill generará los reportes, un analizador que basándose en la información contenida en los logs es capaz de generar tablas y gráficos amigables con detalles que resumen todo el flujo de la navegación. Finalmente los reportes obtenidos son enviados a la persona afectada vía correo electrónico. Todo este proceso se puede hacer crítico en dependencia del volumen de información a procesar para generar los reportes analizados por el personal del departamento. Cuando el rango de días en que ocurrió el incidente es excesivo el proceso se ralentiza, debido a la cantidad de información reunida para procesar.

Estos escenarios de robo de contraseña son efectivos debido a que el servicio de información que se brinda a los usuarios de la Universidad para que tengan detalles de su navegación por Internet como pudieran ser los sitios visitados para que los pueda comparar y verificar si en realidad él accedió a ellos, no está disponible para los usuarios, es por esta razón que esta verificación solo puede realizarse de la manera antes mencionada. La ausencia de información experimentada por los usuarios diariamente crea una falsa expectativa de descontrol del tráfico de la navegación por Internet, causando que los navegantes de Internet incurran en violaciones del código de ética de la Universidad, accediendo a lugares en la red que están restringidos por la misma.

Para detectar el acceso a sitios prohibidos en la Universidad los especialistas hacen revisiones mensuales del tráfico en Internet. Para su ayuda ya existe un log que solo contiene los posibles accesos a estos sitios. El análisis de esta posible información sospechosa también se realiza utilizando Sawmill; pero esta vez la información que contiene el log está más filtrada. Basándose en las estadísticas generadas por el software analizador los especialistas construyen el reporte que contiene a los usuarios que incurrieron en la violación. La medida tomada en el departamento de Seguridad Informática es limitar el acceso a Internet de estos usuarios por un período de dos meses. En el caso de los estudiantes y los profesores el

reporte es enviado al decano de la facultad correspondiente y de ser un trabajador entonces el reporte es enviado al jefe administrativo del área. En ambos casos son tomadas las medidas administrativas pertinentes.

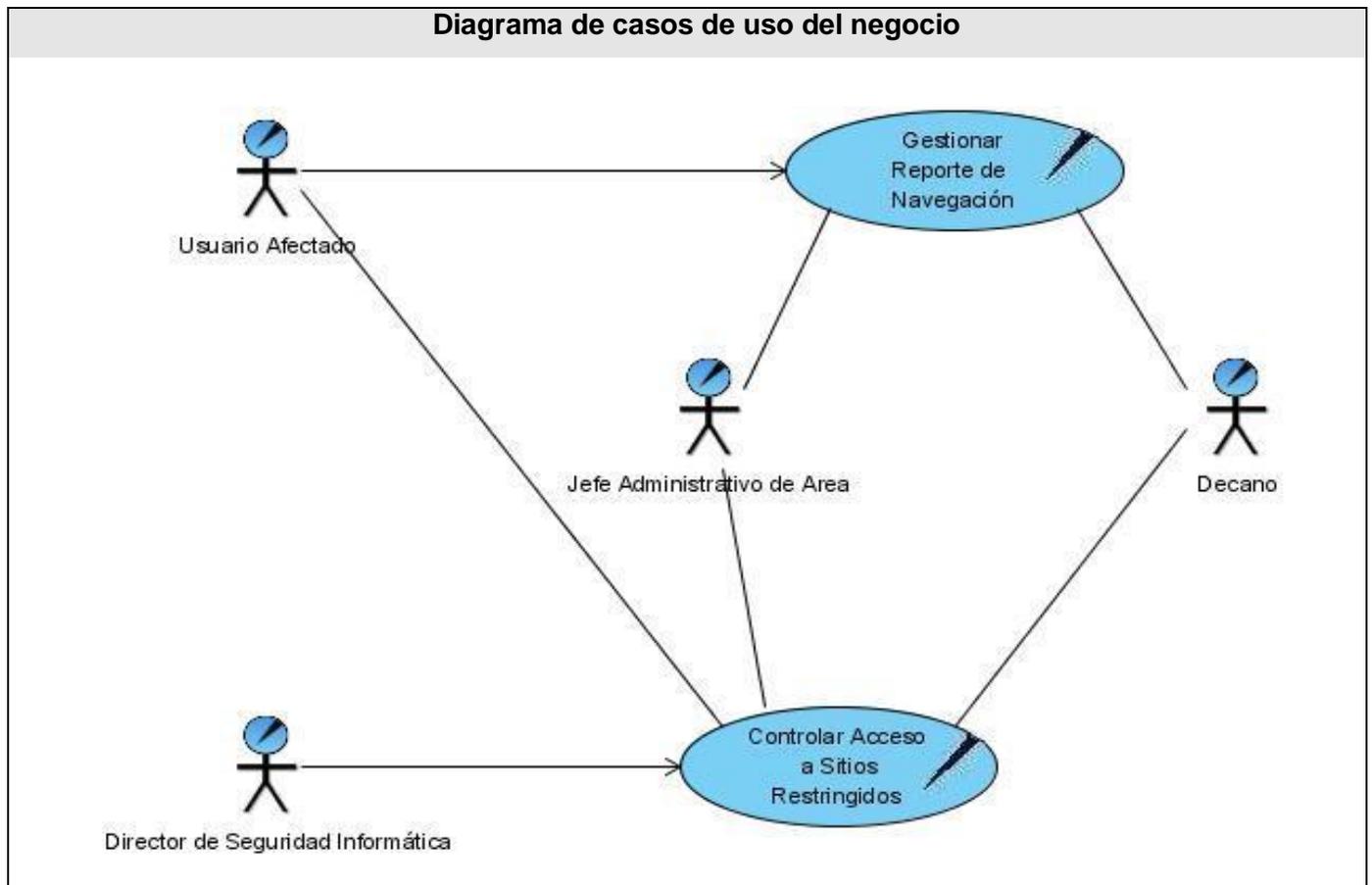
### 2.2.1. Representación de los actores y trabajadores del negocio

<b>Actor</b>	<b>Descripción</b>
Usuario Afectado	El actor usuario afectado inicializa el caso de uso Gestionar Reporte de Navegación, encargándose de informar los incidentes del estado de su cuenta de navegación por Internet.
Director de Seguridad Informática	El actor Director de Seguridad Informática inicializa el caso de uso Controlar Acceso a Sitios Restringidos, encargándose de llevar el control de los usuarios que accedieron a sitios prohibidos.
Decano	El actor Decano se asocia al caso de uso Controlar Acceso a Sitios Restringidos y al caso de uso Gestionar Reporte de Navegación encargándose de recibir notificación de los estudiantes y profesores que accedieron a Sitios Restringidos.
Jefe Administrativo de Area	El actor Jefe Administrativo de Área se asocia al caso de uso Controlar Acceso a Sitios Restringidos y al caso de uso Gestionar Reporte de Navegación encargándose de recibir notificación de los trabajadores que accedieron a Sitios Restringidos.

<b>Trabajador</b>	<b>Descripción</b>
Especialista	El trabajador Especialista es el encargado de atender las solicitudes realizadas por los usuarios afectados por un uso descontrolado de su cuenta de navegación por Internet y enviar la información necesaria al software especializado para obtener los reportes.

Sitio de Seguridad Informática	El trabajador Sitio de Seguridad Informática es uno de los encargados de archivar el reporte de incidencia.
Software Descompactador	El trabajador Software Descompactador es el encargado de descomprimir los ficheros compactados por día y concatenar los ficheros logs de los usuarios y generar un fichero log resultante.
Sawmill	El trabajador Sawmill es el encargado de analizar y procesar los ficheros resultantes, generar las estadísticas y obtener reporte.
Administrador de Redes	El trabajador Administrador de Redes es el encargado de suspender por dos meses el servicio de navegación de los usuarios que cometieron indisciplina.

### 2.2.2. Diagrama de casos de uso del negocio

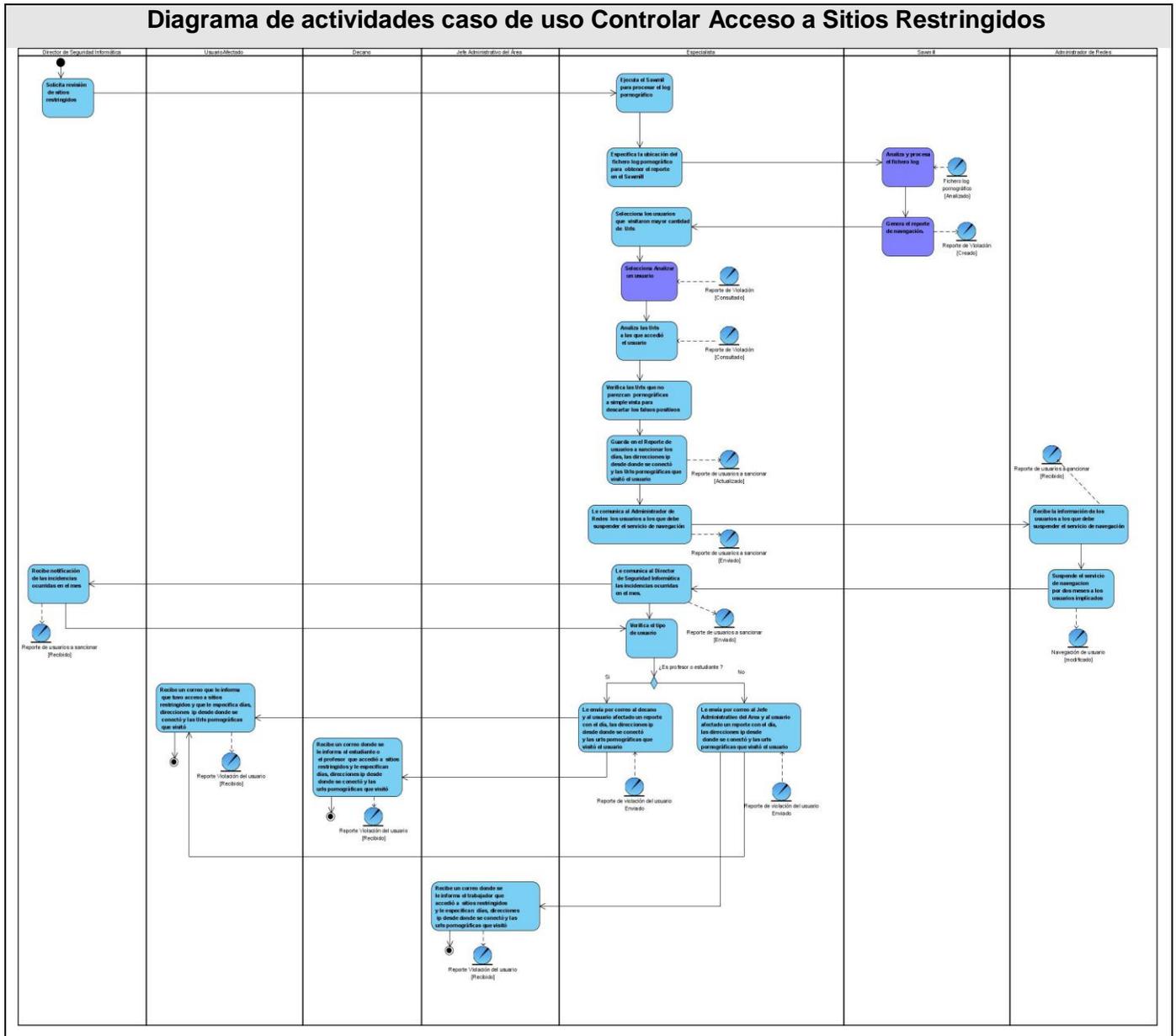


### 2.2.3. Diagramas de actividades del negocio

Un Diagrama de Actividades es una especialización del Diagrama de Estado. Un estado de actividad representa una actividad: un paso en el flujo de trabajo o la ejecución de una operación. Un grafo de actividades contiene estados de actividad que representan la ejecución de una secuencia en un procedimiento, o el funcionamiento de una actividad en un flujo de trabajo. Cuando la actividad termina, entonces la ejecución procede al siguiente estado de actividad dentro del diagrama. Los diagramas de actividades pueden contener bifurcaciones, así como actividades que se pueden realizar concurrentemente por los diversos objetos o personas. Las actividades concurrentes se pueden realizar simultáneamente o en cualquier orden. Los diagramas de actividades son como un organigrama tradicional, excepto que permite el control de concurrencia además del control secuencial.



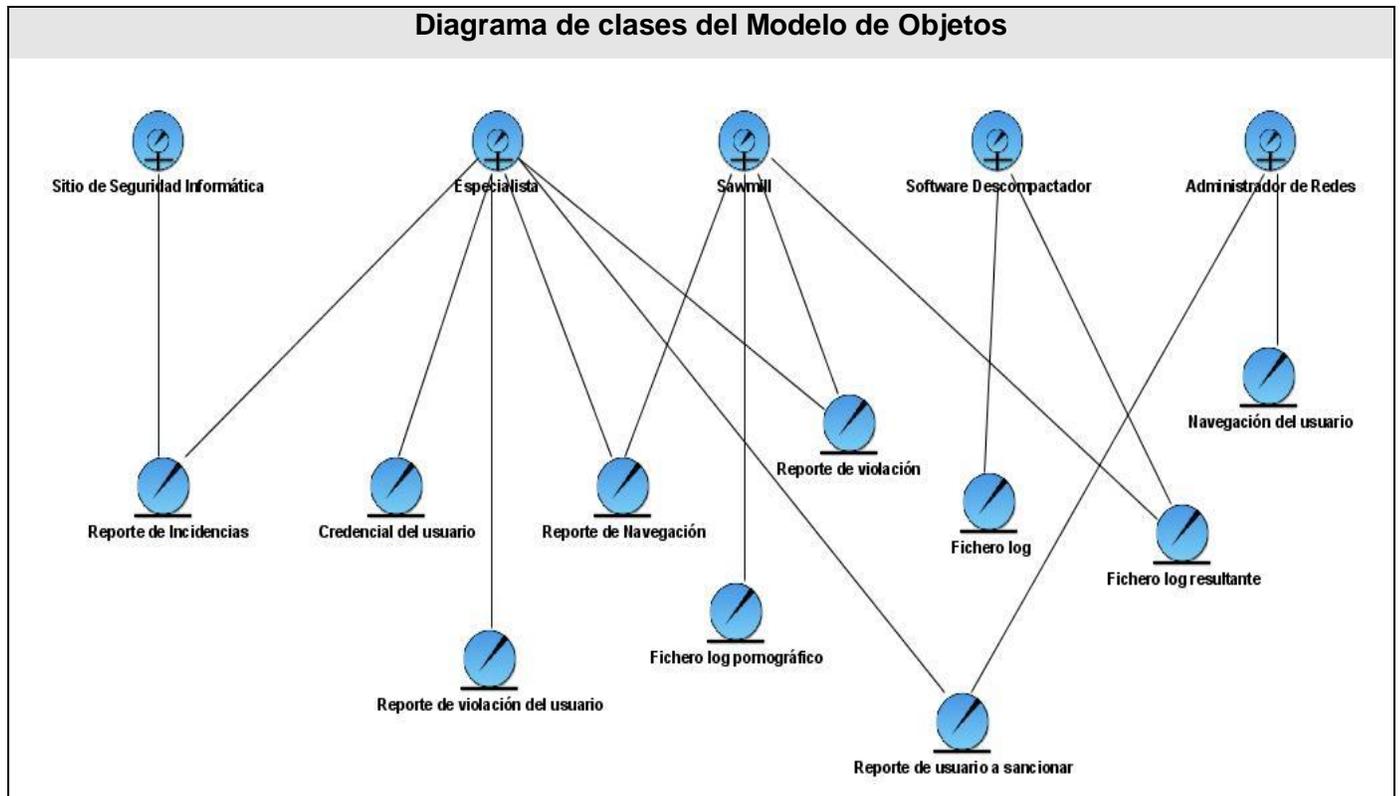
Diagrama de actividades caso de uso Controlar Acceso a Sitios Restringidos



### 2.2.4. Diagrama de clases del Modelo de Objetos

El diagrama de clases del modelo de objetos es el conjunto básico de objetos involucrados en el negocio, es la primera definición del modelo de estructura de objetos para la aplicación, puede realizarse simultáneo con el modelado de procesos de negocios, pero se recomienda comenzar modelando los

objetos ya que son la esencia de este enfoque. El alcance de los objetos del negocio puede involucrar objetos que no son relevantes al sistema, este modelo de objetos suele ser menos detallado que el modelo de estructura de objetos de la fase de análisis de requerimientos, usualmente son modelados solamente los objetos de la realidad.



### 2.3. Propuesta del Sistema de Reportes de la Navegación por Internet

El sistema en cuestión es una aplicación Web a la que todos los usuarios pueden acceder autenticándose. Los usuarios a través de reportes estadísticos podrán visualizar información referente a los sitios que han visitado, la hora de acceso, las descargas realizadas, las direcciones IP de los ordenadores desde donde se han conectado, etc. Para lograr que el sistema sea eficiente en su tiempo de respuesta este no obtendrá la información directamente de los ficheros logs, sino que la misma permanecerá en una base de datos, la cual se actualizará diariamente. El proceso de actualización de la base de datos necesita ser automatizado para que se lleve a cabo en una hora determinada.

Los directivos deben poder ver y controlar la navegación de sus subordinados para tomar inmediatamente y de forma más directa cualquier medida administrativa, sin depender de la Dirección de Redes y Seguridad Informática.

Hasta el momento en la UCI uno de los analizadores que se utiliza es Webalizer. Este software es de gran ayuda; pero generalmente desde un punto de vista administrativo; pues los reportes que generan son resúmenes muy generales. Además para la utilización de estos analizadores expertos es necesario tener acceso a los ficheros logs del servidor Proxy, esto impediría a un usuario el uso de los mismos, sin embargo la propuesta del sistema que se plantea es ajena a estos requerimientos, o sea que los usuarios solo tendrán que conectarse al sistema a través de un navegador para visualizar los detalles de su navegación.

### **2.4. Especificación de los requisitos de software**

Los requerimientos son muy importantes en el proceso de producción de software, ya que enfocan un área fundamental: la definición de lo que se desea producir. Su principal tarea consiste en la generación de especificaciones correctas que describan con claridad y sin ambigüedades el comportamiento del sistema.

#### **2.4.1. Requerimientos no Funcionales**

Los requerimientos no funcionales son los aspectos del sistema visibles para el usuario, que no están relacionados de forma directa con el comportamiento funcional del sistema. Especifican propiedades del sistema como restricciones de ambiente y desarrollo, rendimiento, dependencias de plataformas, mantenibilidad y confiabilidad. Los requerimientos de rendimiento imponen condiciones sobre los requerimientos funcionales como velocidad, tiempo de respuesta y uso de la memoria.

##### **2.4.1.1. Requerimientos de apariencia o interfaz externa**

Las personas que utilizarán el producto no todas tienen conocimientos profundos de informática, por lo que la interfaz debe ser amigable e intuitiva, además el sistema debe ser Interactivo.

### **2.4.1.2. Requerimientos de Usabilidad**

La aplicación tendrá gran aceptación por las personas que la utilicen porque tendrá una interfaz amigable e intuitiva y se solucionará uno de los problemas que se presentan actualmente de que los reportes de la navegación por Internet son enviados por correo electrónico y son reportes estáticos sin embargo los usuarios al utilizar la aplicación, que va a ser interactiva, podrán filtrar los datos por los que quieren obtener el reporte de navegación por Internet. Los usuarios tendrán privilegios por lo que sólo podrán realizar las acciones asignadas.

### **2.4.1.3. Requerimientos de Rendimiento**

El producto será más eficiente en la medida que se haga un mejor aprovechamiento de los recursos que se disponen en el modelo Cliente/Servidor, y la velocidad de las consultas en la Base de Datos. Además para lograr el rápido acceso a sus páginas se deben seguir las diferentes técnicas de elaboración en la Web para así alcanzar un funcionamiento óptimo. La aplicación propuesta debe ser rápida y el tiempo de respuesta a las solicitudes de los usuarios debe ser el mínimo posible.

### **2.4.1.4. Requerimientos de Soporte**

Para garantizar el soporte de esta herramienta, se mostrará una página de información al usuario que le permitirá a los mismos manejar cualquier duda que puedan tener respecto al uso de la aplicación, además en caso de surgir algún problema los usuarios pueden enviar un correo a los especialistas de seguridad informática solicitando que se le aclare algún tema que no les quede claro, los desarrolladores tendrán la posibilidad de proporcionar mantenimiento al sistema y darle solución a cualquier problema que se presente con la aplicación.

### **2.4.1.5. Requerimientos de Portabilidad**

La aplicación propuesta podrá ser usada bajo cualquier sistema operativo; el Servidor Web y la Base de Datos estarán en el mismo ordenador y corriendo sobre Sistema Operativo Linux. Para su implementación se usará como herramienta de desarrollo Eclipse y Gestor de Base de Datos Postgres.

### **2.4.1.6. Requerimientos de Seguridad**

#### Confidencialidad

La información manejada por el sistema debe estar protegida de acceso no autorizado y divulgación. En la aplicación se establecen permisos de control de la navegación y cada usuario podrá ver solamente la navegación de los usuarios a los que tiene acceso.

#### Integridad

La información utilizada por el sistema debe estar debidamente protegida contra la corrupción y estados inconsistentes. No se permite el libre acceso a los log generados por el servidor Proxy Squid.

#### Disponibilidad

La aplicación debe estar disponible en todo momento para aquellas personas con acceso a la información y los mecanismos utilizados para lograr la seguridad no deben ser un obstáculo a los usuarios para obtener los datos deseados en un momento dado.

### **2.4.1.7. Requerimientos de Software**

Para hacer uso de la aplicación los usuarios necesitan de una computadora que sólo requiere de un navegador Web (Internet Explorer o Mozilla Firefox), bajo cualquier sistema operativo; en el Servidor Web y de Base de Datos se utilizará la Distribución Red Hat Enterprise Linux 4. Para la implementación de la aplicación se utilizará el Gestor de Base de Datos Postgres versión 8.0 y como herramienta de desarrollo Eclipse versión 3.2.2. Se utilizará Apache Tomcat versión 5.5.

#### Requerimientos de Hardware.

Servidor: Pentium 3, CPU 1.00 GHz, 2GB de RAM.

Cliente: Como mínimo una computadora de 128 MB de RAM.

Todas las computadoras deben estar conectadas a una red y tener al menos 100 MB de velocidad.

### **2.4.1.8. Restricciones en el diseño y la implementación**

La calidad del sistema va a estar dada en gran medida por el tiempo de respuesta. La aplicación debe responder a las peticiones de los usuarios en el menor tiempo posible.

Para garantizar el desarrollo de la aplicación se utilizará como guía el Proceso Unificado de Desarrollo (RUP). Para visualizar, especificar, construir y documentar los artefactos se utilizará el lenguaje UML (Unified Modelling Language) y como herramienta de apoyo a este lenguaje de modelación se utilizará

Visual Paradigm SDE for Eclipse/WebSphere Interprise Edition. La aplicación será programada en Java utilizando el framework Spring. Para la gestión de configuración se utilizará Subversión y Tortoise.

### 2.4.2. Requerimientos Funcionales

Las capacidades o condiciones que el sistema debe cumplir son denominadas requerimientos funcionales.

- RF1 Permitir actualizar automáticamente en una Base de Datos la información contenida en los ficheros log.
  - RF1.1 Ubicar la información a actualizar en la base de datos.
  - RF1.2 Actualizar la información almacenada en los ficheros log.
- RF2 Permitir autenticar y verificar a los usuarios que tienen servicio de navegación. [Ver Anexo 2.](#)
  - RF2.1 Mostrar información del usuario (nombre del usuario).
- RF3 Gestionar permisos de supervisión de la navegación por Internet de otros usuarios a un supervisor(es un usuario).
- RF4 Permitir supervisar la navegación por Internet de los usuarios asignados a un supervisor.
- RF5 Mostrar reportes de la navegación por Internet de los usuarios a sitios permitidos o prohibidos.
  - RF5.1 Visualizar información referente al consumo por sitios que el usuario ha visitado. [Ver Anexo 5.](#)
  - RF5.2 Visualizar información referente al consumo por días de navegación. [Ver Anexo 4.](#)
  - RF5.3 Visualizar información referente al consumo por horas de navegación.
  - RF5.4 Visualizar información referente al consumo por direcciones IP desde las que se navegó. [Ver Anexo 4.](#)

### 2.5. Definición de los actores del Sistema

Un actor puede ser un humano, un software o un hardware que interactúe con el Sistema. Un actor no es parte del Sistema, sino un rol que se juega dentro del mismo. Este puede intercambiar información o puede ser un recipiente pasivo de la misma. Es una entidad externa al Sistema, por lo tanto, al identificar actores se delimita el Sistema y se define su alcance.

<b>Actor</b>	<b>Descripción</b>
Usuario	Persona que solicita los reportes de su navegación por Internet.
Supervisor	Persona que supervisa la navegación por Internet de los usuarios que le fueron asignados.
Administrador del Sistema	Persona encargada de establecer a un usuario el permiso de supervisión de la navegación por Internet de otros usuarios.
Reloj	Tarea programada del sistema encargada de procesar la información contenida en los ficheros logs y actualizar la base de datos centralizada del sistema.

## **2.6. Listado de los casos de uso del Sistema**

Los casos de uso son una técnica para capturar información de cómo un sistema o negocio trabaja, o de cómo se desea que trabaje, describen bajo la forma de acciones y reacciones el comportamiento de un sistema desde el punto de vista del usuario. Son descripciones de la funcionalidad del sistema independientes de la implementación.

CU-1	CU_Actualizar Información
Actor	Reloj
Descripción	Es una tarea programada del sistema, el caso de uso se inicia cuando el sistema automáticamente, todos los días a una hora escogida previamente se ejecuta y actualiza la información almacenada en los log en la Base de Datos.
Referencia	RF1

CU-2	CU_Autenticar
Actor	Usuario
Descripción	El caso de uso se inicia cuando el Usuario solicita entrar al Sistema, introduce su identificador de usuario y contraseña en la página de autenticación del sistema, si los datos son correctos el sistema le mostrará una página principal donde el usuario según

	los permisos que tenga asignados podrá ver detalles de su navegación.
Referencia	RF2

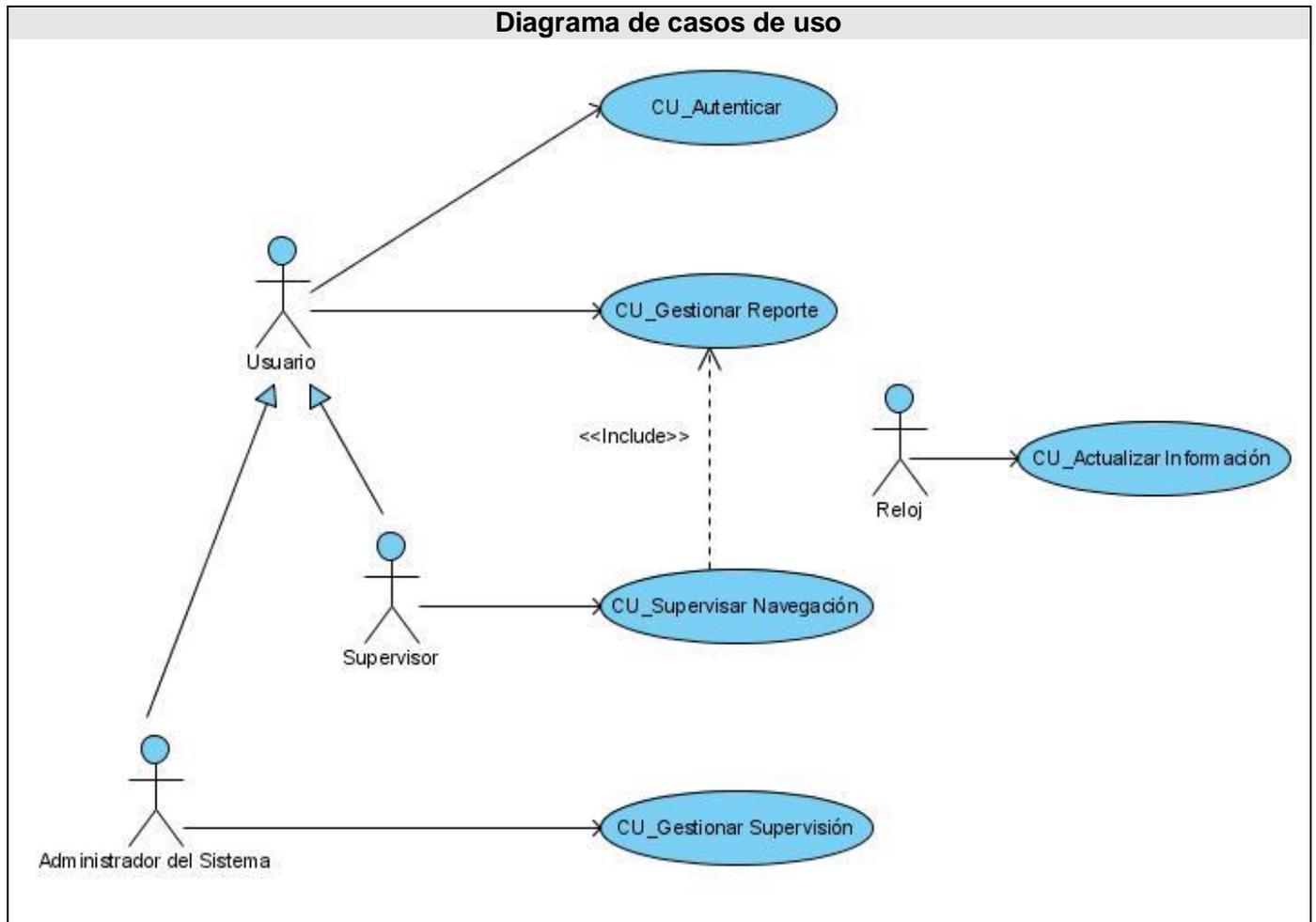
CU-3	CU_Gestionar Supervisión
Actor	Administrador del Sistema
Descripción	El caso de uso se inicia cuando el administrador del sistema decide establecer los permisos a los supervisores, que son los usuarios que podrán chequear la navegación de otros así como definir por cada supervisor que usuarios podrá controlar.
Referencia	RF3

CU-4	CU_Supervisar Navegación
Actor	Supervisor
Descripción	El caso de uso se inicia cuando un supervisor quiere chequear la navegación de un usuario, el sistema le muestra todos los usuarios a los que el tiene permiso para supervisarle su navegación y el supervisor escoge el usuario que quiere supervisar y el sistema le muestra los detalles de la navegación del usuario.
Referencia	RF4

CU-5	CU_Gestionar Reporte
Actor	Usuario
Descripción	El caso de uso se inicia cuando un usuario solicita ver los detalles de su navegación escogiendole previamente el rango de días que le interesa y el sistema le muestra los detalles de su navegación. El usuario podrá conocer detalles referente al consumo por URLs, al consumo por IP por días y consumo por horas.
Referencia	RF5

### 2.7. Diagrama de casos de uso del Sistema

Los diagramas de casos de uso especifican la funcionalidad y el comportamiento de un sistema mediante su interacción con los usuarios y/o otros sistemas. O sea, es un diagrama que representa los actores, los casos de uso y las relaciones que se establecen entre ellos.



## 2.8. Casos de uso expandidos

### 2.8.1. Caso de uso CU\_Autenticar

CU-1	CU_Autenticar	
<b>Propósito</b>	El caso de uso CU_Autenticar tiene como objetivo validar la entrada de usuarios al sistema.	
<b>Actores:</b> Usuario		
<b>Resumen:</b> El caso de uso se inicia cuando el Usuario solicita entrar al Sistema, el mismo introduce los datos de acceso (nombre de usuario y contraseña). El Usuario entra al Sistema o se le es denegado el acceso en caso de que los datos sean incorrectos.		
<b>Referencias</b>	RF1	
<b>Acción del actor</b>	<b>Respuesta del sistema</b>	
1. El caso de uso comienza cuando el usuario solicita acceder al Sistema.	2. El Sistema muestra la página de autenticación para que el usuario introduzca los datos de acceso.	
3. El usuario introduce su nombre de usuario y contraseña.		
4. El usuario selecciona el botón "Aceptar".	5. El Sistema verifica que los datos sean correctos.	
	6. El Sistema muestra la página principal.	
<b>Flujos alternativos</b>		
<b>Acción del actor</b>	<b>Respuesta del sistema</b>	
Acción 3	3.1 El sistema muestra el mensaje "No se permiten campos vacíos".	
3.2 El usuario selecciona el botón "OK".	3.3 Se pasa a la Acción 2 del caso de uso.	
Acción 5	5.1 El sistema muestra el mensaje "Usuario o contraseña incorrecta".	
	5.2 Se pasa a la Acción 3 del caso de uso.	

Flujos alternativos

**Acción 3:** Si el usuario selecciona el botón "Aceptar" sin llenar todos los campos requeridos para acceder al Sistema.

**Acción 5:** Si el Sistema verifica que los datos son incorrectos.

**2.8.2. Caso de uso CU\_Gestionar Reporte**

CU-2	CU_Gestionar Reporte	
<b>Propósito</b>	El caso de uso CU_Gestionar Reporte tiene como objetivo generar los reportes de navegación del usuario.	
<b>Actores:</b> Usuario		
<b>Resumen:</b> El caso de uso se inicia cuando el Usuario selecciona el intervalo de días para los que quiere obtener el reporte de navegación y la clasificación del tipo de URL que quiere ver. Luego el usuario selecciona el tipo de reporte y seguidamente el sistema lo muestra.		
<b>Referencias</b>	RF5	
<b>Acción del actor</b>	<b>Respuesta del sistema</b>	
1. El Usuario selecciona la opción “Rango Días”. <a href="#">Ver Anexo 3.</a>	2. El sistema muestra la interfaz correspondiente para seleccionar el rango de días para los que quiere obtener el reporte de navegación y el tipo de URL que el usuario desea consultar.	
3. El Usuario selecciona el rango de días y el tipo de URL.		
4. El Usuario selecciona la opción “Enviar”.	5. El sistema verifica que el rango de días sea correcto.	
	6. El sistema muestra el mensaje “Cambios salvados exitosamente”.	
7. El Usuario selecciona la opción “Reportes”.	8. El Sistema despliega las opciones: “Consumo x URL”, “Consumo x IP”, “Consumo x Días” y “Consumo x Horas”.	
9. Si el usuario selecciona la opción: a) “Consumo x URL”. Ver sección: “Consumo x URL” b) “Consumo x IP”. Ver sección:		

<p>“Consumo x IP”</p> <p>c)“Consumo x Días”. Ver sección: “Consumo x Días”</p> <p>d)“Consumo x Horas”. Ver sección: “Consumo x Horas”</p>	
<p><b>Sección “Consumo x URL”</b></p>	
	<p>10. El sistema muestra un reporte general de navegación del usuario dirigiéndose específicamente a las estadísticas de Consumo x URL.</p>
<p>11. El usuario filtra el reporte general seleccionando los vínculos que se encuentran en este.</p>	<p>12. El sistema muestra reportes estadísticos filtrados de acuerdo con la selección del usuario.</p>
<p><b>Sección “Consumo x IP”</b></p>	
	<p>13. El sistema muestra un reporte general de navegación del usuario dirigiéndose específicamente a las estadísticas de Consumo x IP.</p>
<p>14.El usuario filtra el reporte general seleccionando los vínculos que se encuentran en este.</p>	<p>15. El sistema muestra reportes estadísticos filtrados de acuerdo con la selección del usuario.</p>
<p><b>Sección “Consumo x Días”</b></p>	
	<p>16. El sistema muestra un reporte general de navegación del usuario dirigiéndose específicamente a las estadísticas de Consumo x Días.</p>
<p>17. El usuario filtra el reporte general seleccionando los vínculos que se</p>	<p>El sistema muestra reportes estadísticos filtrados de acuerdo con la selección del usuario.</p>

encuentran en este.	
<b>Sección “Consumo x Horas”</b>	
	18. El sistema muestra un reporte general de navegación del usuario dirigiéndose específicamente a las estadísticas de Consumo x Horas.
19. El usuario filtra el reporte general seleccionando los vínculos que se encuentran en este.	20. El sistema muestra reportes estadísticos filtrados de acuerdo con la selección del usuario.
<b>Flujos alternativos</b>	
<b>Acción del actor</b>	<b>Respuesta del sistema</b>
<b>Acción 1</b>	1.1 Se pasa a la Acción 2 del caso de uso.
<b>Acción 3</b>	3.1 El Sistema muestra el mensaje “No se permiten campos vacíos”.
3.2 El usuario selecciona el botón “OK”.	3.3 Se pasa a la Acción 2 del caso de uso.
<b>Acción 5 F1</b>	5 F1.1 El Sistema muestra el mensaje “Introduzca un rango de días correcto”.
5 F1.2 El usuario selecciona el botón “OK”.	5 F1.3 Se pasa a la Acción 2 del caso de uso.
<b>Acción 5 F2</b>	5 F2.1 El Sistema muestra el mensaje “El rango no puede exceder de una semana”.
5 F2.2 El usuario selecciona el botón “OK”.	5 F2.3 Se pasa a la Acción 2 del caso de uso.

Flujos alternativos

**Acción 1:** Si el Usuario selecciona realizar cualquier tipo de reporte sin haber seleccionado el rango de días previamente.

**Acción 3:** Si el Usuario selecciona la opción “Enviar” sin haber escogido el rango de días.

**Acción 5 F1:** Si el Sistema verifica que el día inicial es posterior al día final.

**Acción 5 F2:** Si el Sistema verifica que el rango seleccionado excede la cantidad de días permitidos para mostrar el reporte.

2.8.3. Caso de uso CU\_Gestionar Supervisión

CU-3	CU _Gestionar Supervisión	
<b>Propósito</b>	El caso de uso CU_Gestionar Supervisión tiene como objetivo asignar un usuario para que sea analizado por un supervisor y eliminar la asignación de uno o varios usuarios a un supervisor.	
<b>Actores:</b> Administrador del Sistema		
<b>Resumen:</b> El caso de uso se inicia cuando el Administrador del Sistema selecciona la opción “Administrar” y gestiona la información relacionada con los supervisores y los usuarios que estos tienen asignados.		
<b>Referencias</b>	RF3	
<b>Acción del actor</b>	<b>Respuesta del sistema</b>	
1. El Administrador del Sistema selecciona la opción Administrar.	2. El Sistema despliega las opciones “Asignar supervisión” y “Eliminar supervisión”.	
3. Si el Administrador del Sistema selecciona la opción: a) “Asignar supervisión”. Ver sección: “Asignar supervisión”. b) “Eliminar supervisión”. Ver sección: “Eliminar supervisión”.		
<b>Sección “Asignar supervisión”</b>		
	4. El sistema muestra la interfaz correspondiente para buscar el supervisor y el usuario que será supervisado.	
5. El Administrador del Sistema selecciona el botón para buscar el supervisor.	6. El Sistema muestra la interfaz para poder buscar al supervisor.	
7. El Administrador inserta el nombre de usuario del supervisor y selecciona la opción “Buscar”.	8. El Sistema verifica que exista el usuario.	

	9. El Sistema activa la opción "Agregar".
10. El Administrador del Sistema selecciona la opción "Agregar".	11. El Sistema muestra la interfaz de búsqueda principal con el campo de supervisor completado con el nombre de usuario escogido por el Administrador del Sistema.
12. El Administrador selecciona el botón para buscar el usuario que será supervisado.	13. El Sistema muestra la interfaz para poder buscar al usuario que será asignado.
14. El Administrador inserta el usuario que será analizado por el supervisor y selecciona la opción "Buscar".	15. El Sistema verifica que exista el usuario.
	16. El Sistema activa la opción "Agregar".
17. El Administrador del Sistema selecciona la opción "Agregar".	18. El Sistema muestra la interfaz de búsqueda principal con los campos supervisor y usuario completados con los nombres del supervisor y usuario respectivamente escogidos por el Administrador del Sistema.
19. El Administrador selecciona la opción "Enviar".	20. El sistema verifica que el campo de supervisor y usuario no contengan el mismo nombre de usuario.
	21. El Sistema asigna el usuario al supervisor.
	22. El sistema muestra el mensaje "Datos actualizados exitosamente".
<b>Sección "Eliminar supervisión"</b>	
<b>Acción del actor</b>	<b>Respuesta del sistema</b>
	23. El sistema muestra la interfaz correspondiente para buscar el supervisor.
24. El Administrador selecciona el botón para buscar el supervisor.	25. El Sistema muestra la interfaz para insertar el nombre de usuario del supervisor que se desea buscar.

26. El Administrador inserta el nombre de usuario del supervisor y selecciona la opción "Buscar".	27. El Sistema verifica que exista el usuario.
	28. El Sistema activa la opción "Agregar".
29. El Administrador del Sistema selecciona la opción "Agregar".	30. El Sistema muestra la interfaz de eliminar usuarios a un supervisor con el campo de supervisor completado con el nombre de usuario escogido por el Administrador del Sistema.
31. El usuario selecciona la opción "Enviar".	32. El Sistema muestra la interfaz con los datos de los usuarios que tiene asignados el supervisor seleccionado.
33. El Administrador del Sistema selecciona el/los usuario(s) que no desea sea(n) supervisado(s).	
34. El Administrador del Sistema selecciona el botón "Actualizar".	35. El Sistema elimina el derecho de supervisar a los usuarios seleccionados.
	36. El Sistema muestra el mensaje "Datos procesados exitosamente", además de los usuarios que le quedan asignados al supervisor.
<b>Flujos alternativos</b>	
<b>Acción del actor</b>	<b>Respuesta del sistema</b>
Acción 5 F1	5 F1.1 Se realizan las acciones 13-17 del caso de uso.
	5 F1.2 El Sistema muestra la interfaz de búsqueda principal con el campo de usuario completado con el nombre de usuario escogido por el Administrador del Sistema.

	5 F1.3 Se realizan las acciones 5-10 del caso de uso.
	5 F1.4 Se pasa a la Acción 18 del caso de uso.
Acción 5 F2	5 F2.1 El sistema muestra el mensaje “No se permiten campos vacíos”.
5 F2.2 El usuario selecciona el botón “OK”.	5 F2.3 Se pasa a la Acción 4 del caso de uso.
Acción 8	8.1 El sistema muestra el mensaje “El usuario no existe”.
	8.2 Se pasa a la Acción 7 del caso de uso.
Acción 10	10.1 Se pasa a la Acción 7 del caso de uso.
Acción 12	12.1 El sistema muestra el mensaje “No se permiten campos vacíos”.
12.2 El usuario selecciona el botón “OK”.	12.3 Se pasa a la Acción 11 del caso de uso.
Acción 15	15.1 El sistema muestra el mensaje “El usuario no existe”.
	15.2 Se pasa a la Acción 14 del caso de uso.
Acción 17	17.1 Se pasa a la Acción 14 del caso de uso.
Acción 20	20.1 El sistema muestra el mensaje “Usuarios iguales”.
20.2 El usuario selecciona el botón “OK”.	20.3 Se pasa a la Acción 18 del caso de uso.
Acción 24	24.1 El sistema muestra el mensaje “No se permiten campos vacíos”.
24.2 El usuario selecciona el botón “OK”.	24.3 Se pasa a la Acción 23 del caso de uso.
Acción 27	27.1 El sistema muestra el mensaje “El usuario no existe”.
	27.2 Se pasa a la Acción 25 del caso de uso.
Acción 29	29.1 Se pasa a la Acción 26 del caso de uso.
Acción 32	32.1 El sistema muestra el mensaje “No existen usuarios asignados”.

	32.2 Se pasa a la Acción 23 del caso de uso.
Acción 33	33.1 El sistema muestra el mensaje “Debe seleccionar al menos un usuario”.
33.2 El usuario selecciona el botón “OK”.	33.3 Se pasa a la Acción 32 del caso de uso.

Flujos alternativos

**Acción 5 F1:** Si el Administrador selecciona el botón para buscar el usuario que será supervisado.

**Acción 5 F2:** Si el Administrador del Sistema selecciona el botón “Enviar” sin buscar el identificador de usuario del supervisor o del usuario que será supervisado.

**Acción 8:** Si el Sistema verifica que el usuario especificado no existe.

**Acción 10:** Si el Administrador del Sistema desea buscar otro supervisor.

**Acción 12:** Si el Administrador del Sistema selecciona el botón “Enviar” sin buscar el nombre de usuario del usuario que será supervisado.

**Acción 15:** Si el Sistema verifica que el usuario especificado no existe.

**Acción 17:** Si el Administrador del Sistema desea buscar otro usuario.

**Acción 20:** Si el Sistema verifica que los identificadores de usuario son iguales.

**Acción 24:** Si el Administrador del Sistema selecciona el botón “Enviar” sin buscar el nombre de usuario del supervisor.

**Acción 27:** Si el Sistema verifica que el usuario especificado no existe.

**Acción 29:** Si el Administrador del Sistema desea buscar otro supervisor.

**Acción 32:** Si el supervisor no tiene usuarios asignados.

**Acción 33:** Si el Administrador del Sistema selecciona la opción “Actualizar” sin seleccionar ningún usuario.

#### 2.8.4. Caso de uso CU\_Supervisar Navegación

CU-4	CU_Supervisar Navegación
<b>Propósito</b>	El caso de uso CU_Supervisar Navegación tiene como objetivo supervisar la navegación por Internet de un usuario.
<b>Actores:</b> Supervisor	

<b>Resumen:</b> El caso de uso se inicia cuando un supervisor desea chequear la navegación de los usuarios que tiene asignados. El Sistema le muestra la lista de los usuarios a los que tiene permiso para supervisar su navegación y el supervisor selecciona el usuario que desea supervisar.	
<b>Referencias</b>	RF4
<b>Acción del actor</b>	<b>Respuesta del sistema</b>
1. El Supervisor del Sistema selecciona la opción Supervisar.	2. El Sistema despliega la opción "Usuarios".
3. El Supervisor del Sistema selecciona la opción "Usuarios".	4. El sistema muestra la lista de usuarios que el supervisor puede analizar.
5. El Supervisor del Sistema selecciona el usuario que desea analizar.	
<b>Puntos de extensión</b>	
Línea 5. El supervisor observa al reporte del usuario seleccionado. Ver caso de uso CU_Gestionar Reporte.	
<b>Flujos alternativos</b>	
<b>Acción del actor</b>	<b>Respuesta del sistema</b>
Acción 4	4.1 El sistema muestra el mensaje: "No hay usuarios asignados".
	4.2 Se termina el caso de uso.

Flujos Alternativos

**Acción 4:** Si el supervisor no tiene usuarios asignados.

### 2.8.5. Caso de uso CU\_Actualizar Información

CU-5	CU_Actualizar Información
<b>Propósito</b>	El caso de uso CU_Actualizar Información tiene como objetivo descomprimir los ficheros logs que contienen la navegación de los usuarios y guardar la información que estos contienen en la Base de Datos.
<b>Actores:</b> Reloj	

<b>Resumen:</b> El caso de uso se inicia cuando se ejecuta una tarea programada del Sistema.	
<b>Referencias</b>	RF4
<b>Acción del actor</b>	<b>Respuesta del sistema</b>
	1. Una tarea programada del Sistema se inicia y descomprime los ficheros que contienen los logs del acceso permitido a Internet de los usuarios.
	2. El Sistema descomprime los ficheros que contienen los logs del acceso prohibido a Internet de los usuarios.
	3. El Sistema actualiza la información contenida en los ficheros logs del acceso permitido a Internet por los usuarios en la Base de Datos.
	4. El Sistema actualiza la información contenida en los ficheros logs del acceso prohibido a Internet por los usuarios en la Base de Datos.

## **2.9. Conclusiones**

En este capítulo se trató el tema de cómo se llevan acabo los procesos de negocio en el Departamento de Seguridad Informática. Se realizó una descripción de la solución del problema dejando claro que la aceptación por parte de los usuarios va a estar condicionada en gran medida por la eficiencia en el tiempo de respuesta, explicando las medidas tomadas para alcanzar este objetivo. Se plantearon además los requerimientos funcionales y no funcionales que se deben cumplir para el desarrollo de la aplicación y el diagrama de los casos de uso del sistema.

## CAPÍTULO 3: ANÁLISIS Y DISEÑO

### 3.1. Introducción

Este capítulo tratará el flujo de trabajo Análisis y Diseño explicando como se cumplen los principales objetivos de este flujo de trabajo, como se transforman los requisitos al diseño del futuro sistema, como se desarrolla la arquitectura para el sistema y se expone además como adaptar el diseño para que sea consistente con el entorno de implementación, diseñado para el rendimiento, en fin, se traducen los requisitos a una especificación que describe cómo implementar el sistema.

### 3.2. Análisis

El análisis es una etapa fundamental dentro de la realización de una aplicación, el principal objetivo de esta etapa es entender el problema. Cuando se termina el análisis se tiene una mayor comprensión del problema, se conocen cuales son las abstracciones claves y se comienza entonces a estudiar como se desenvuelve la aplicación en el tiempo.

#### 3.2.1. Modelo de clases de análisis

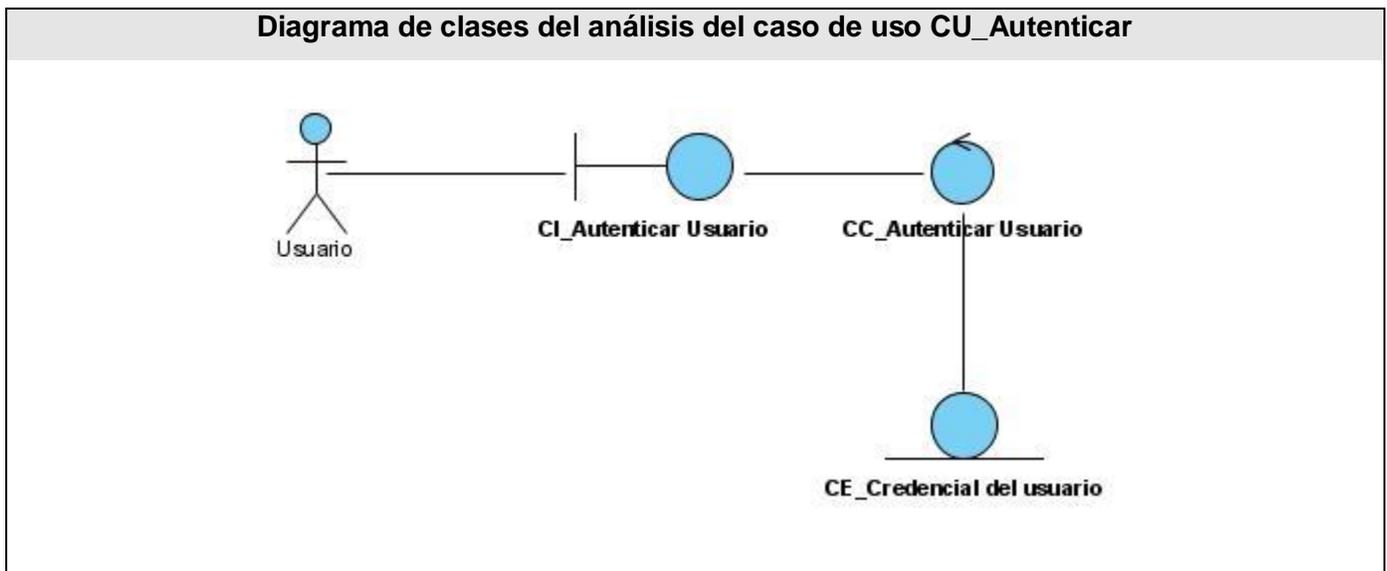


Diagrama de clases del análisis del caso de uso CU\_Gestionar Reporte

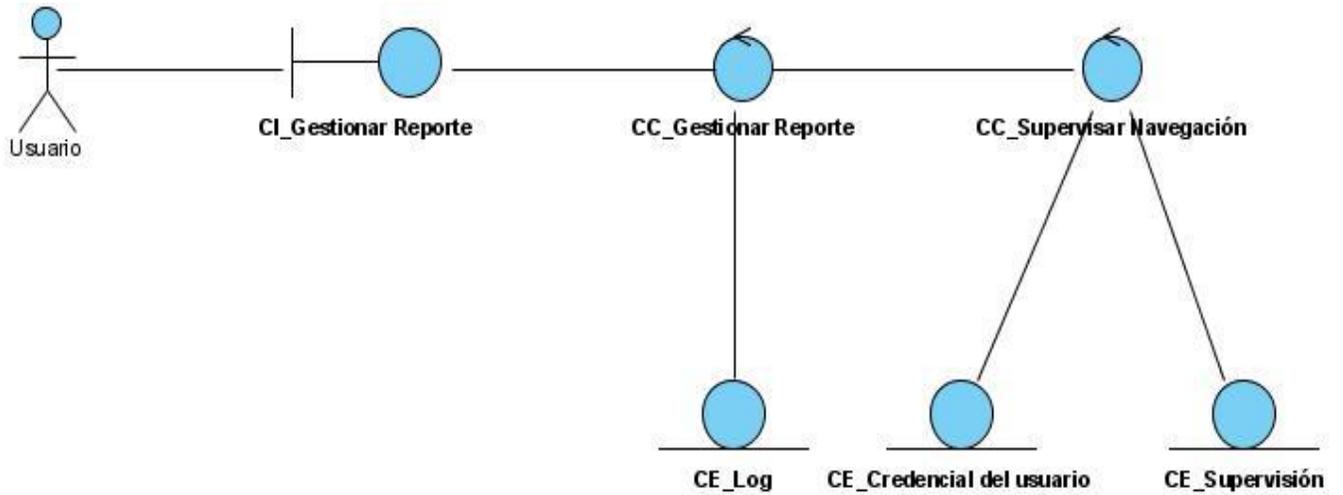
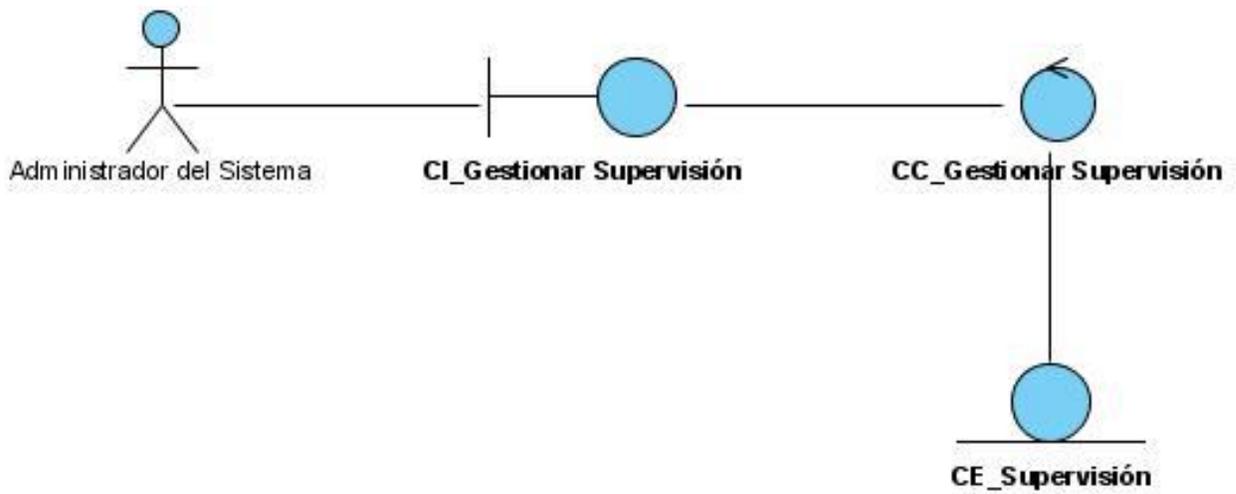
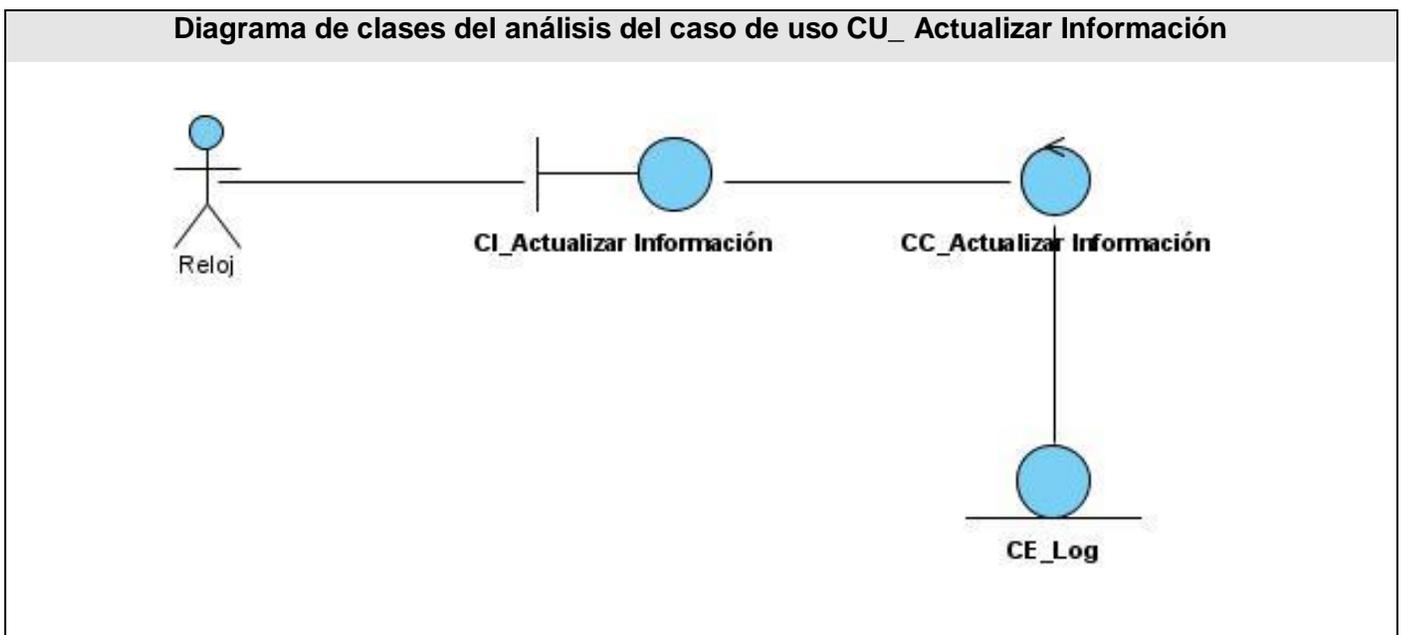
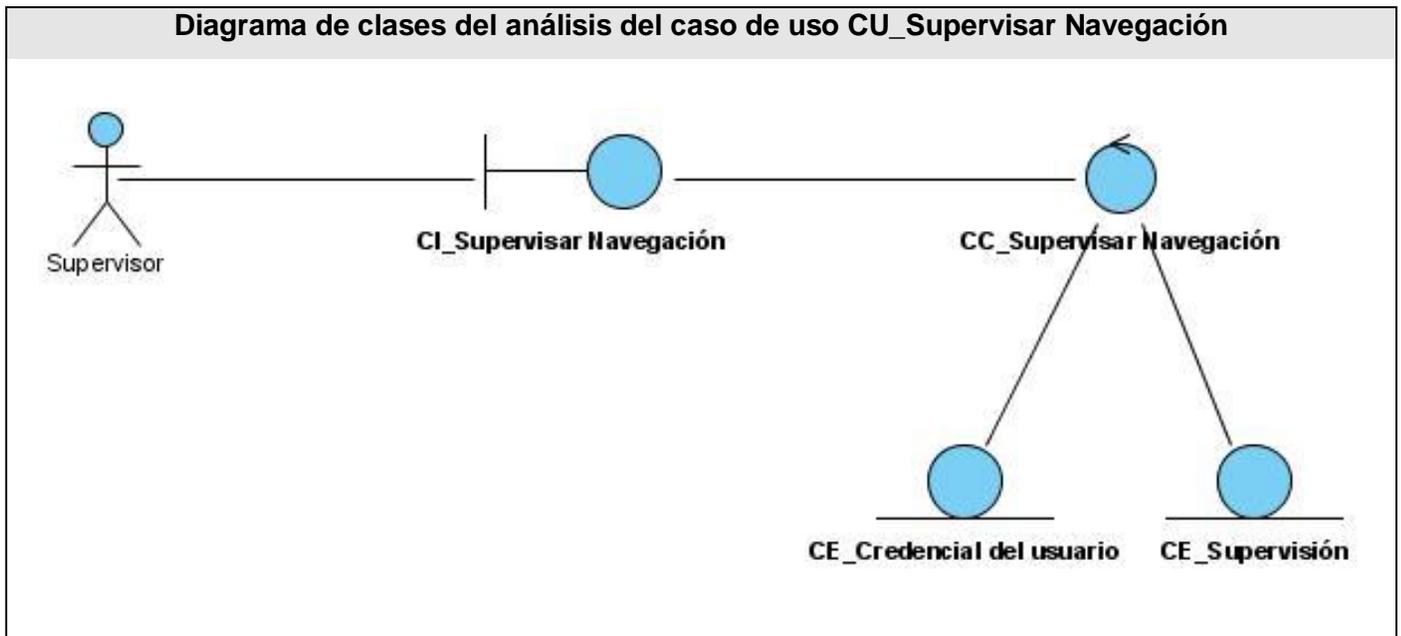


Diagrama de clases del análisis del caso de uso CU\_Gestionar Supervisión





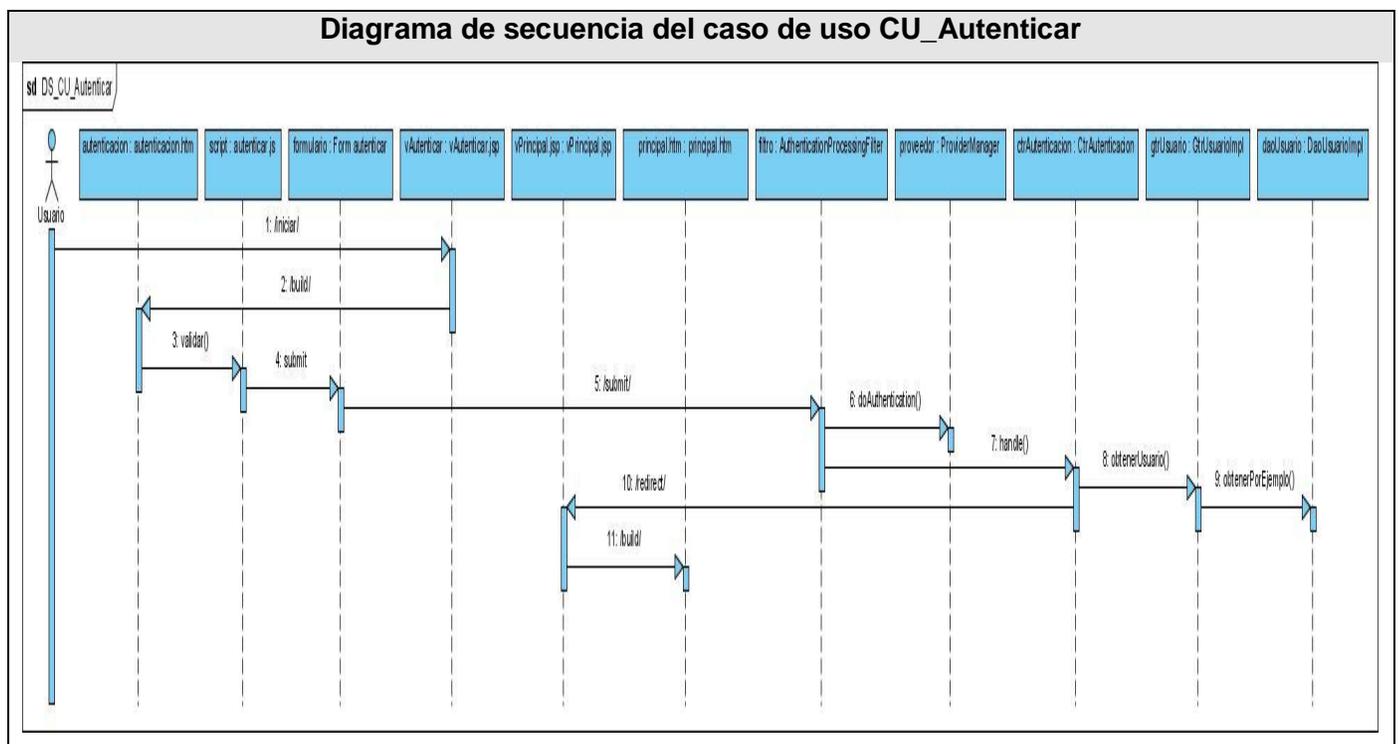
### 3.3. Diseño

El diseño debe definir una solución que satisfaga de modo efectivo y eficiente los requisitos especificados en el análisis, incorpora nuevos artefactos (nuevas clases, nuevos atributos y operaciones para las clases,

etc.) y se tiene en cuenta la plataforma tecnológica concreta sobre la que se construirá el sistema informático.

### 3.3.1. Diagramas de secuencia

Los diagramas de secuencia forman parte del modelado dinámico del sistema. Son representaciones que muestran, para un determinado escenario de un caso de uso, una interacción ordenada según la secuencia temporal de eventos, o sea muestran los objetos participantes en la interacción y los mensajes que intercambian ordenados según su secuencia en el tiempo. De manera general son útiles para documentar el flujo de lógica dentro de la aplicación, así como para observar la vida de los objetos en el sistema, identificar llamadas a realizar o posibles errores del modelado estático, que imposibiliten el flujo de información o de llamadas entre los componentes del sistema.



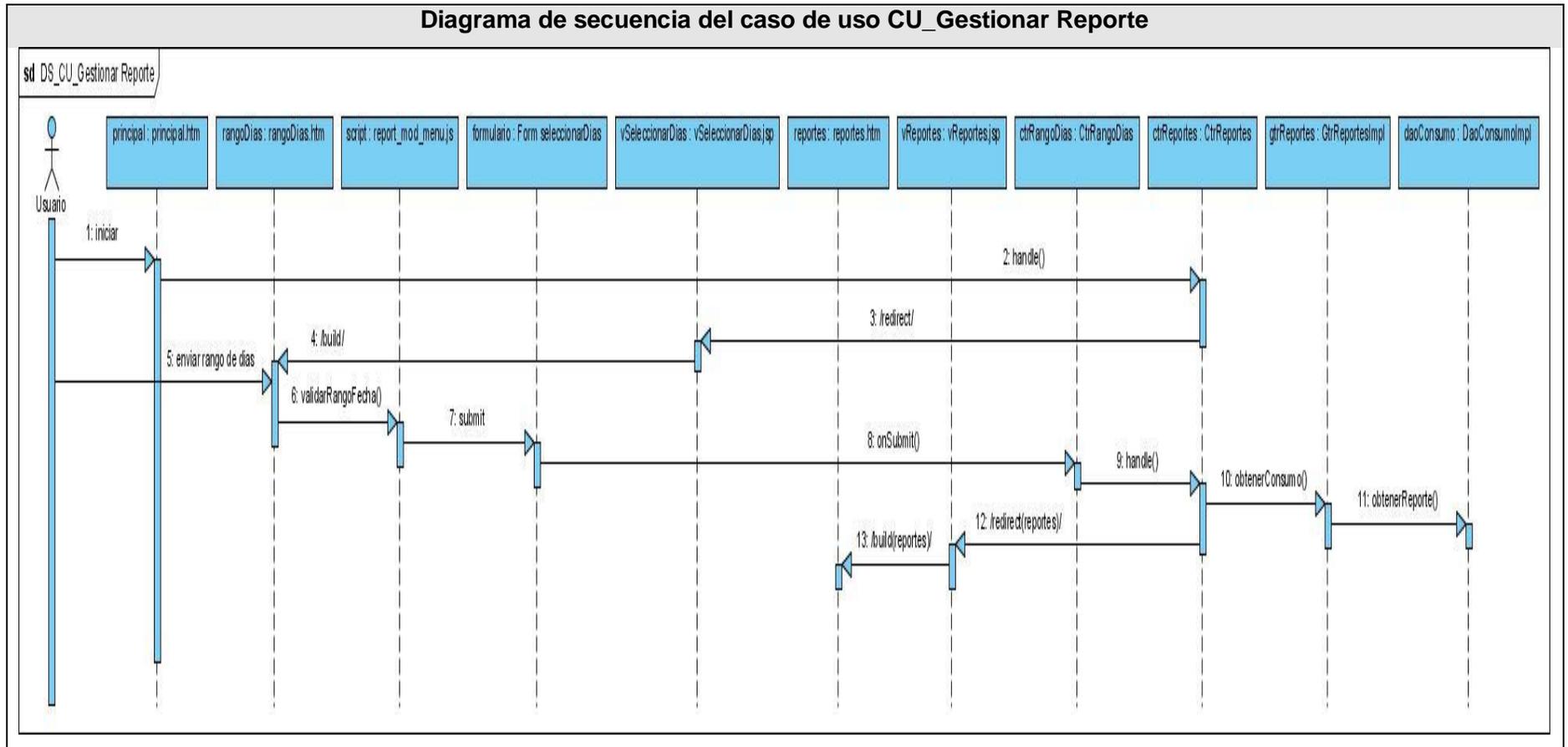
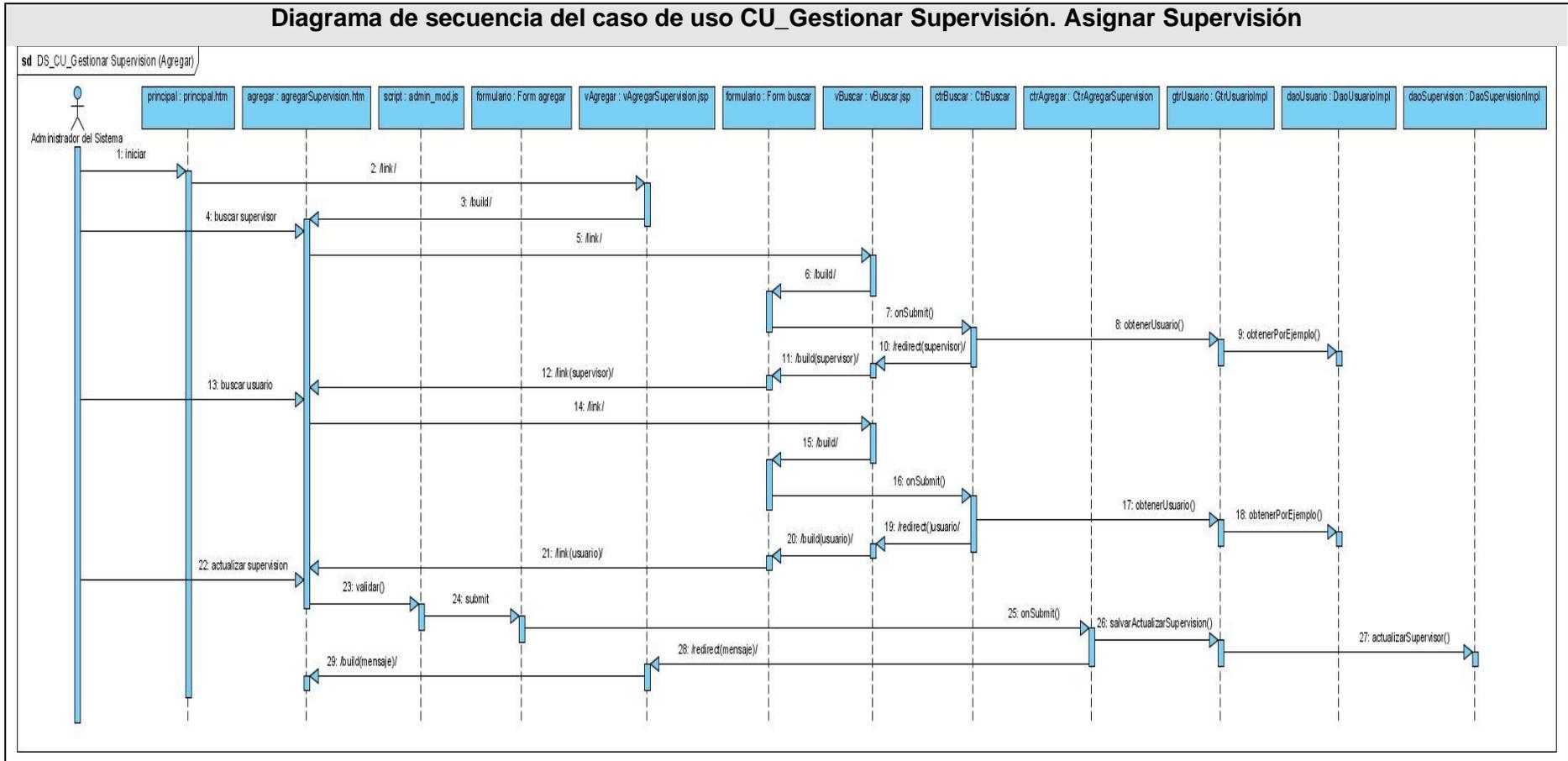
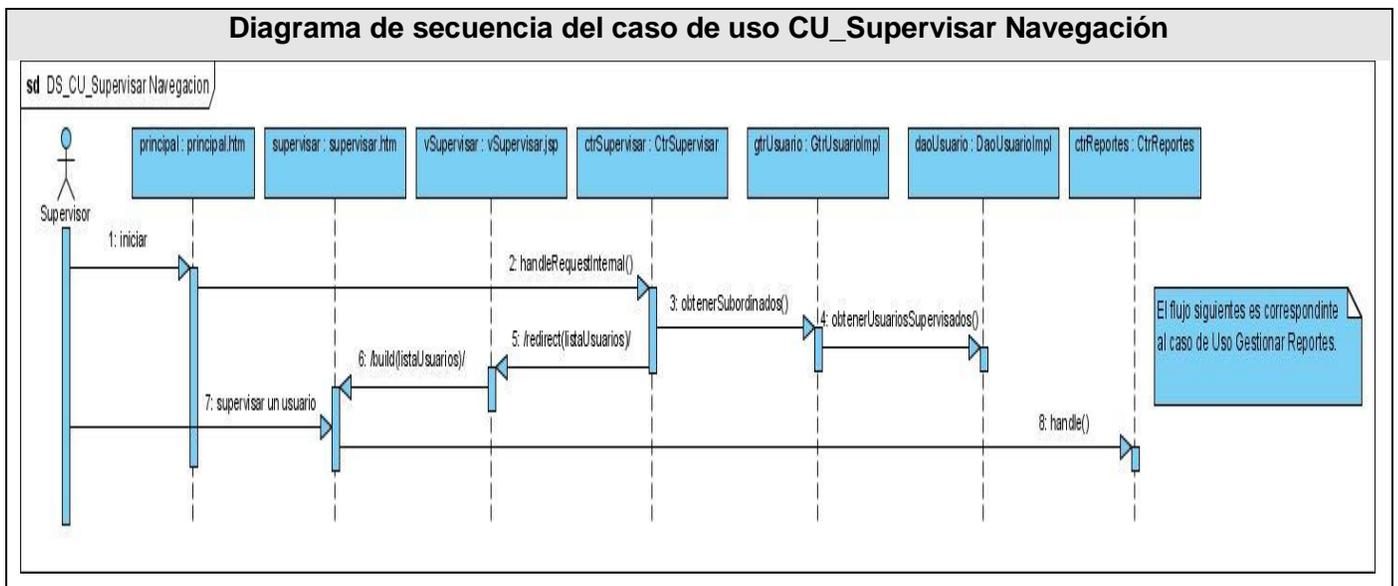
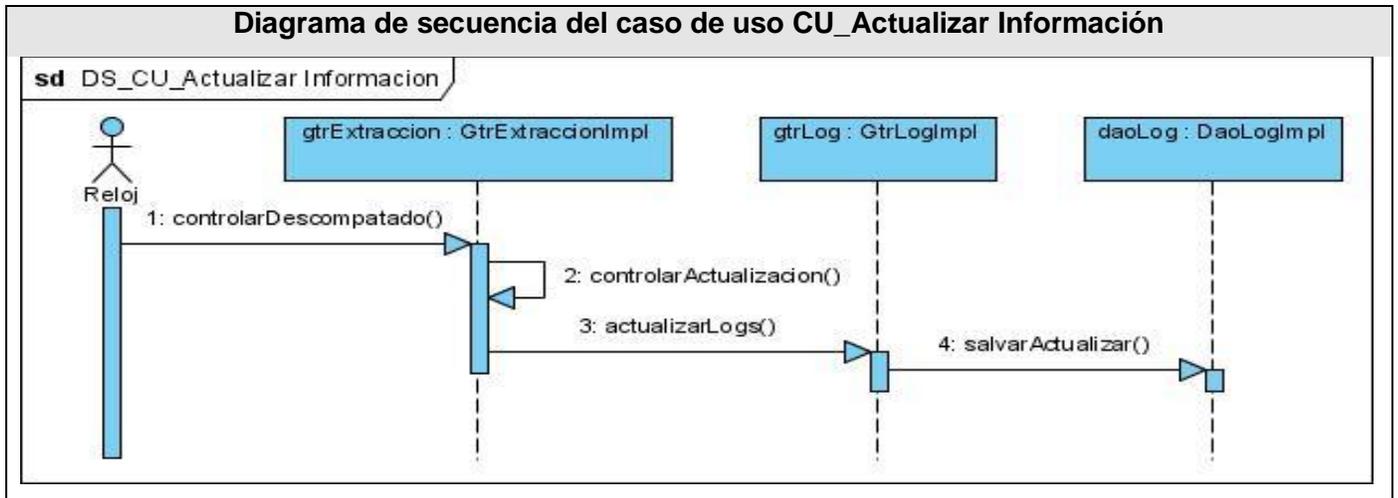


Diagrama de secuencia del caso de uso CU\_Gestionar Supervisión. Asignar Supervisión







### 3.3.2. Diagramas de clases Web

Los diagramas de clases son el diagrama principal para el flujo de trabajo análisis y diseño, se basan fundamentalmente en los diagramas de interacción. Presentan las clases del sistema con sus relaciones estructurales y de herencia. Dentro de las partes que conforman un diagrama de clases se encuentran: clases, relaciones y paquetes. Para desarrollar aplicaciones Web se establece una correspondencia entre

los elementos principales de las aplicaciones Web y los elementos propios de la orientación a objetos y se utilizan los estereotipos UML extendiendo la notación para describir los elementos que pueden aparecer en una aplicación Web.

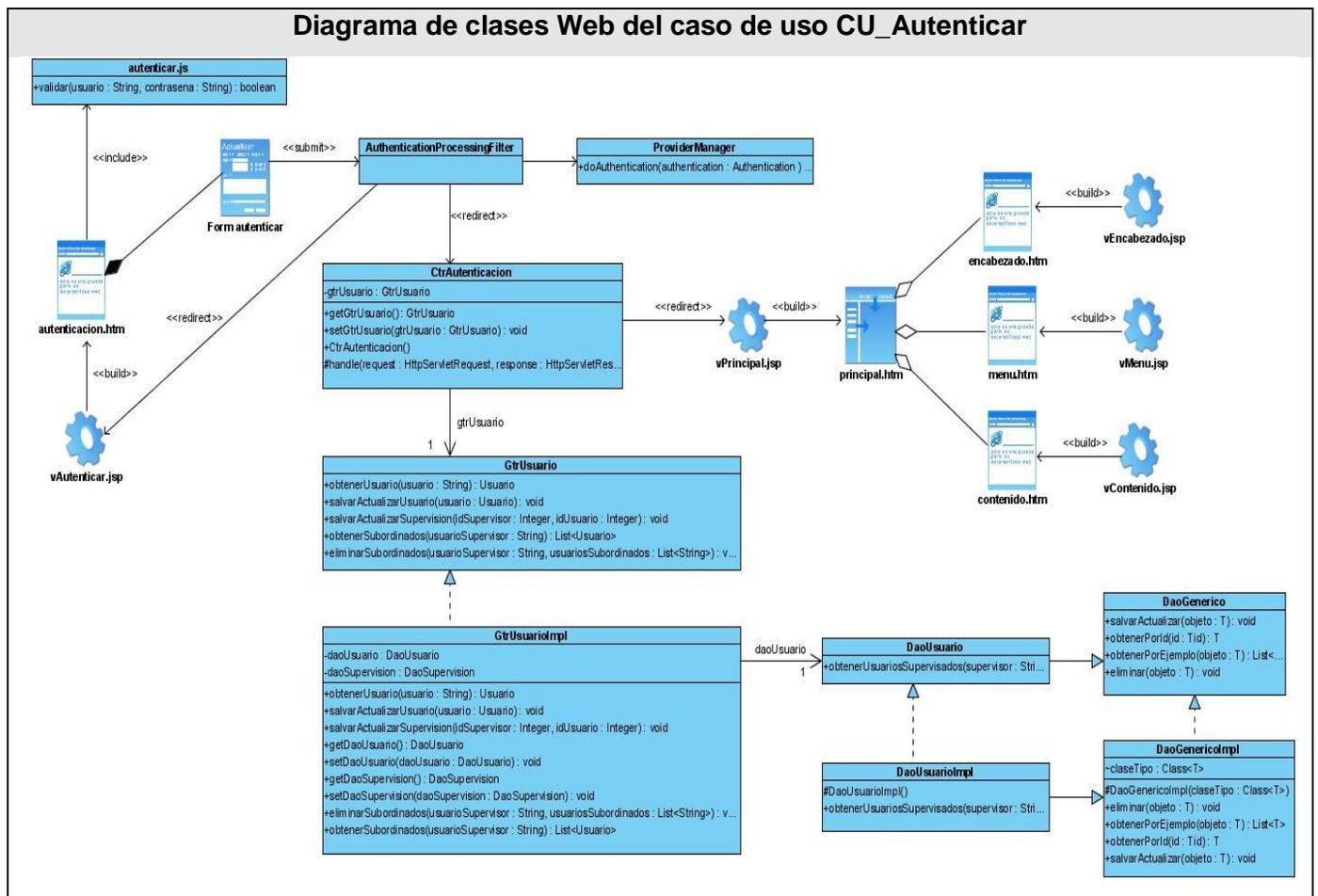


Diagrama de clases Web del caso de uso CU\_Gestionar Reporte

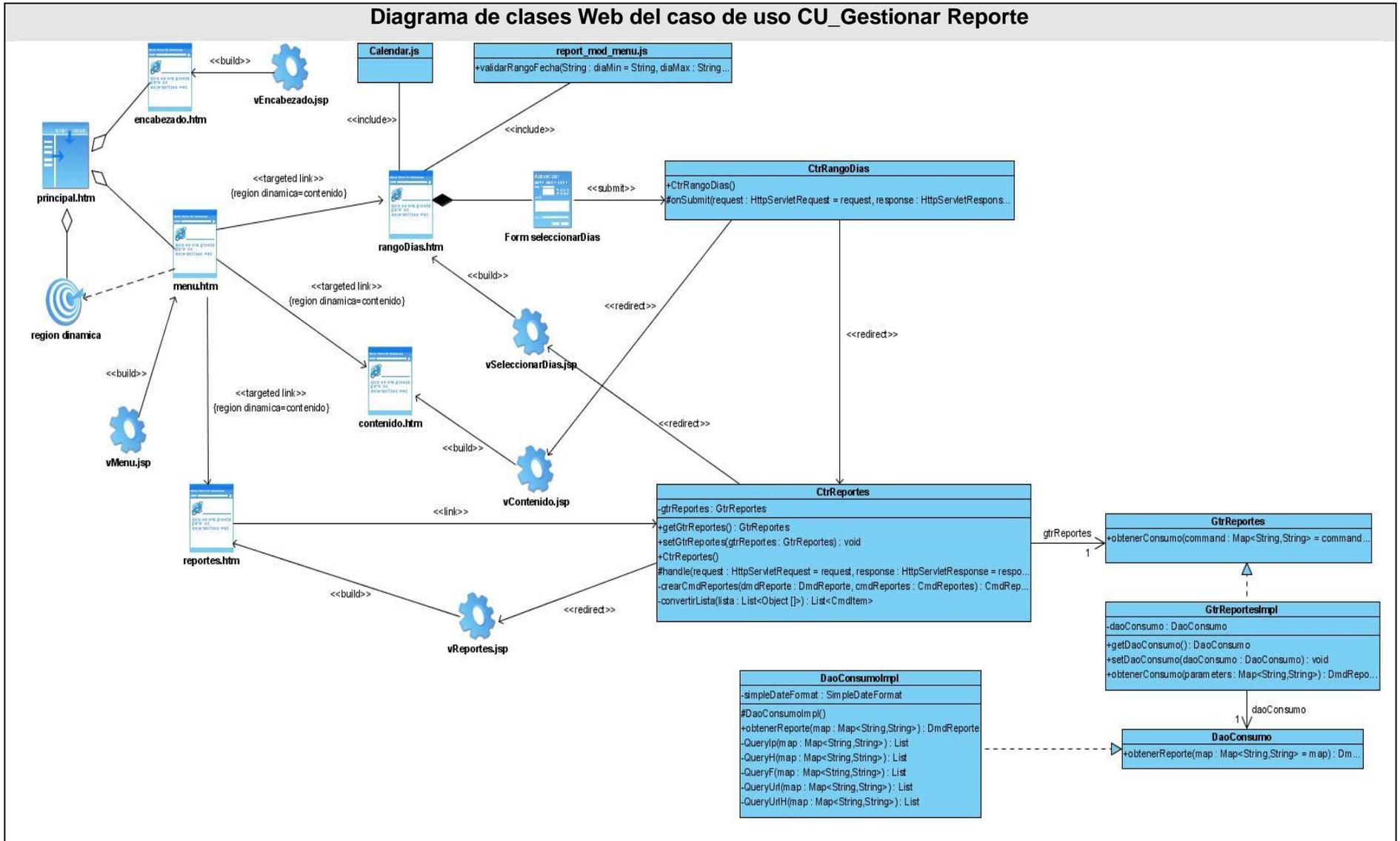


Diagrama de clases Web del caso de uso CU\_Gestionar Supervisión

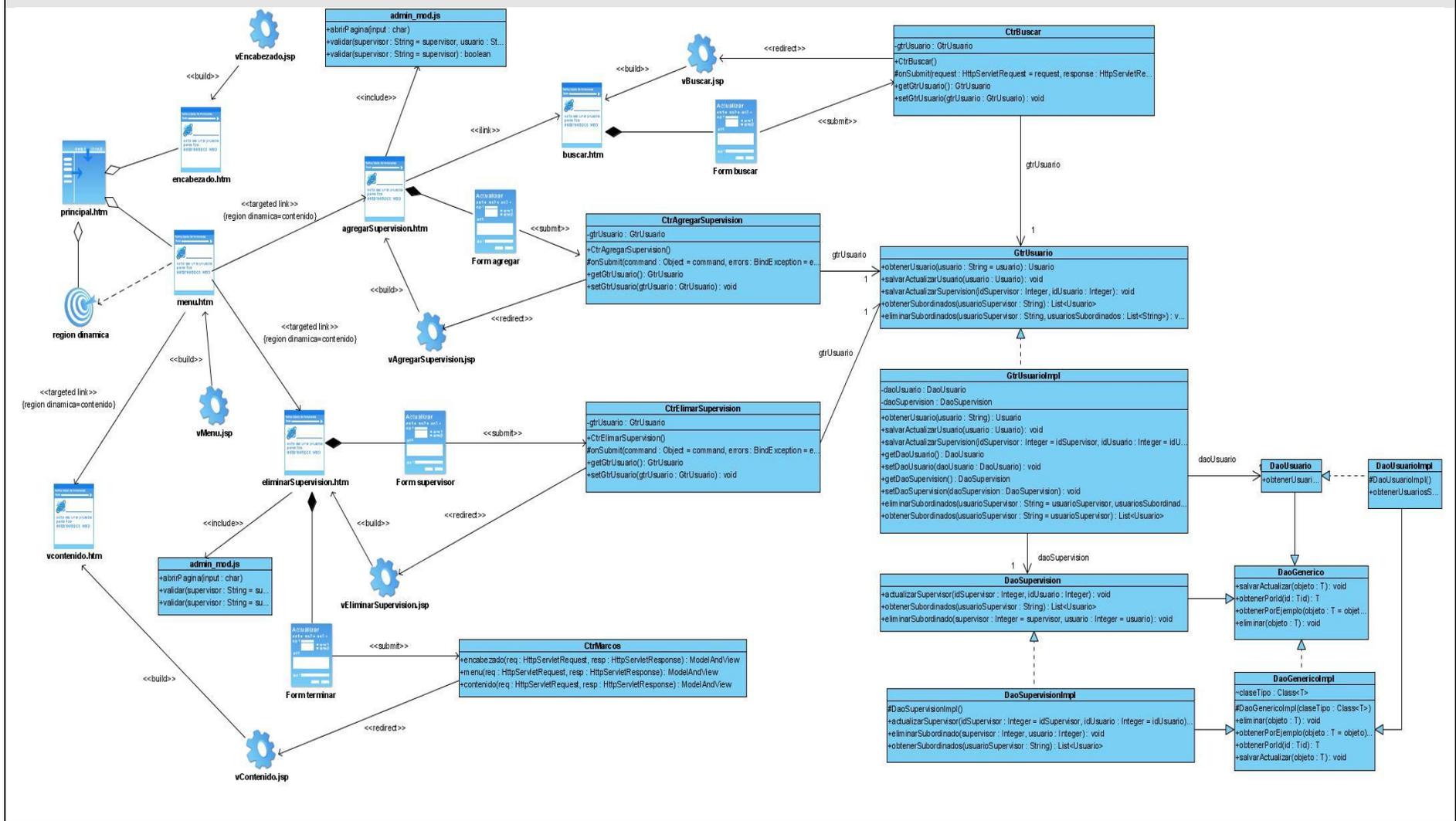


Diagrama de clases Web del caso de uso CU\_Supervisor Navegación

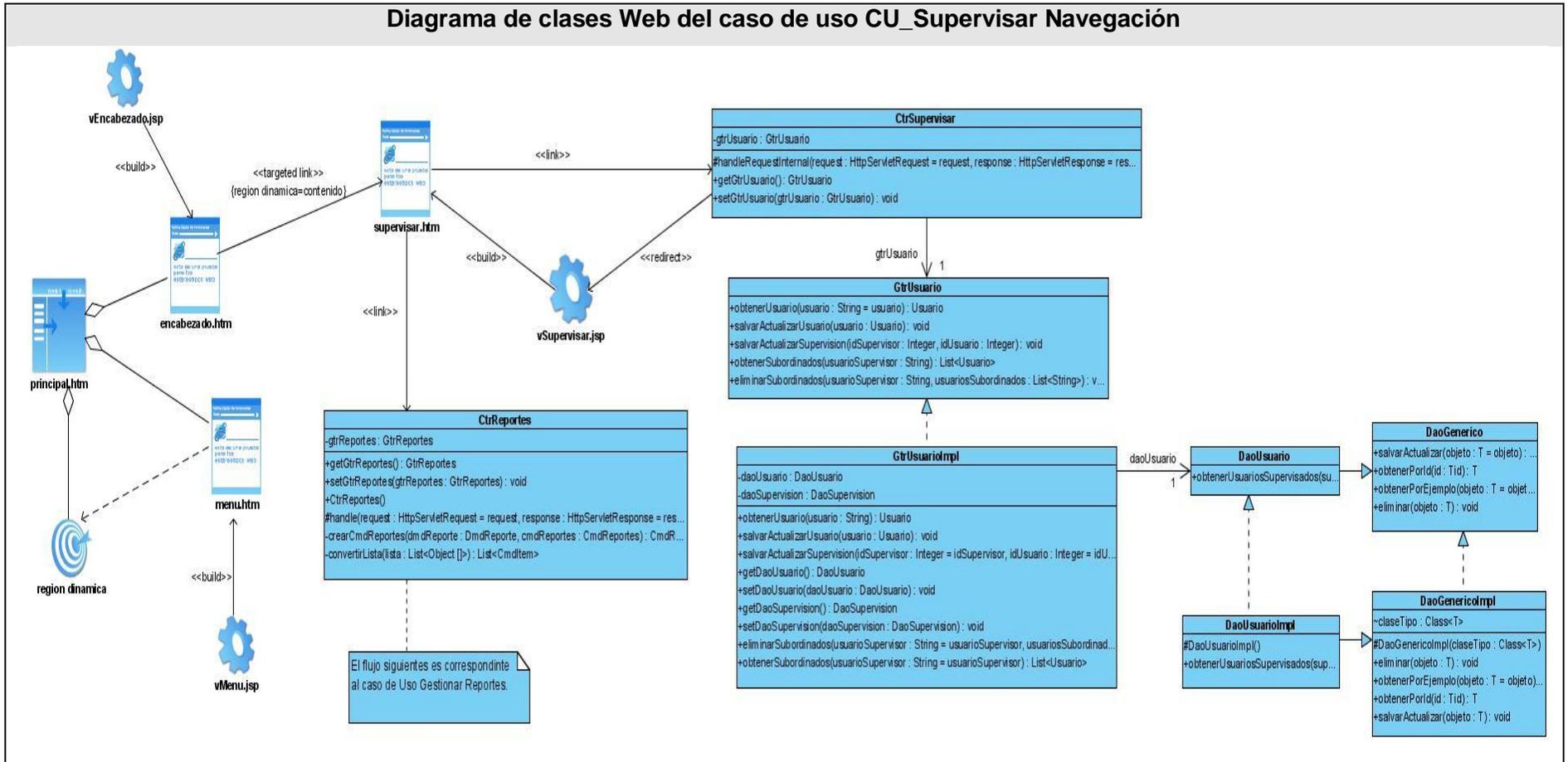
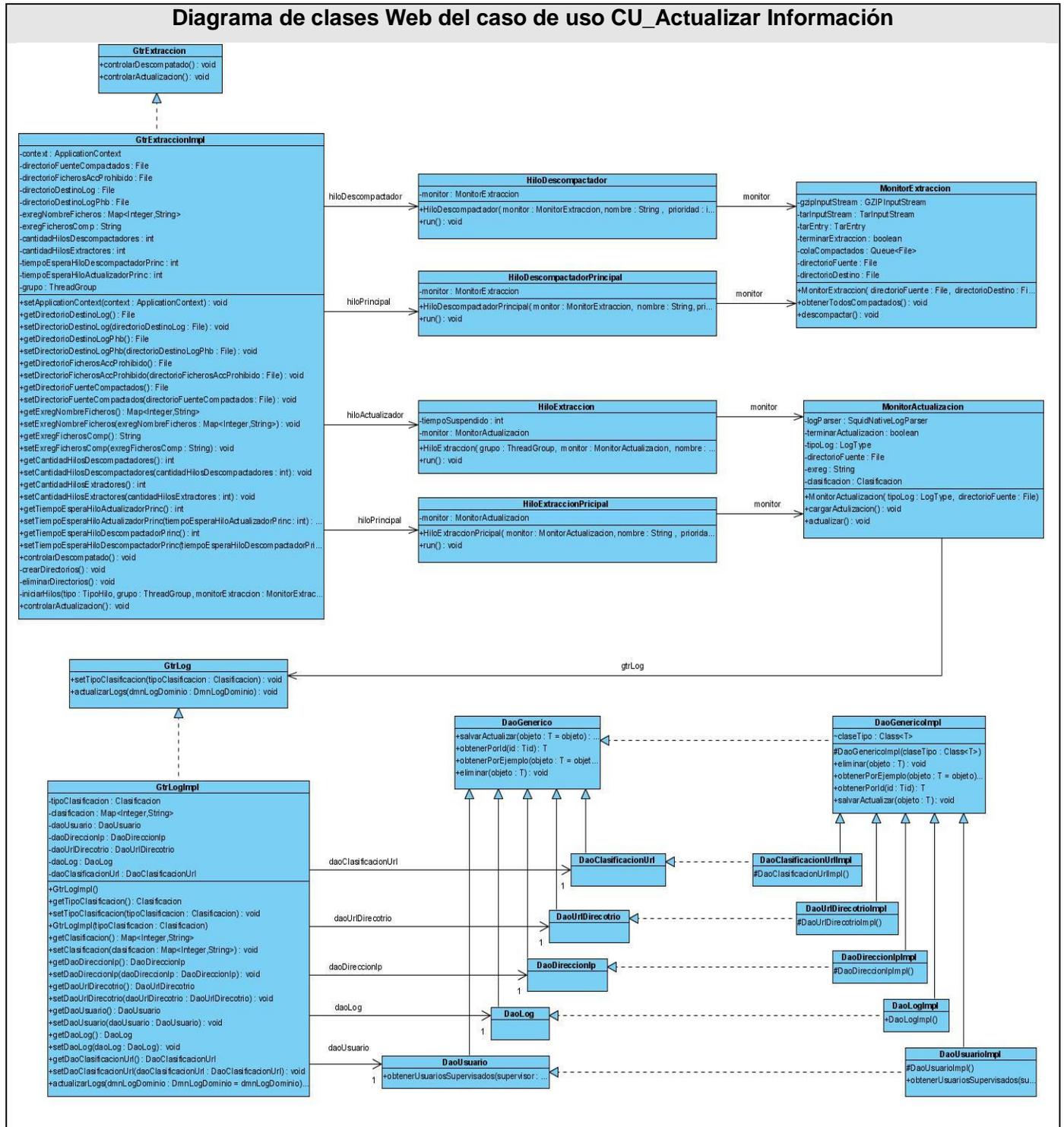


Diagrama de clases Web del caso de uso CU\_Actualizar Información



### 3.3.3. Descripción de las clases

Nombre:	vAgregarSupervision.jsp
Descripción:	Vista que contiene un formulario mediante el cual se le asigna un usuario a un supervisor.
Nombre:	vAutenticacion.jsp
Descripción:	Vista que contiene un formulario mediante el cual el usuario se autentica en el sistema.
Nombre:	buscar.jsp
Descripción:	Vista que contiene un formulario mediante el cual se puede hacer la búsqueda de un usuario.
Nombre:	vContenido.jsp
Descripción:	Vista que encierra el contenido inicial de la vista principal que se le mostrará al usuario.
Nombre:	vEliminarSupervision.jsp
Descripción:	Vista que contiene una lista de usuarios asignados a un supervisor mediante la cual se podrán eliminar dichos usuarios.
Nombre:	vEncabezado.jsp
Descripción:	Vista que contiene el encabezado de la vista principal.
Nombre:	vForward.jsp
Descripción:	Vista usada para redirigir de forma inmediata a la vista de vSeleccionarDias.jsp.
Nombre:	vMenu.jsp
Descripción:	Vista que contiene el menú de la vista principal.
Nombre:	vPrincipal.jsp
Descripción:	Vista principal del sistema que contiene el menú, el encabezado y el contenido principal que se le mostrará al usuario.
Nombre:	vReportes.jsp
Descripción:	Vista que contiene los reportes dinámicos de consumo de la navegación por Internet que se le mostrarán al usuario.
Nombre:	vSeleccionarDias.jsp
Descripción:	Vista que contiene el formulario mediante el cual se establece el rango de días en el

	que se quieren obtener los reportes.
Nombre:	vSupervisar.jsp
Descripción:	Vista que muestra la lista de usuarios asignados al supervisor, mediante la cual se podrá escoger un usuario para ser supervisado.

Nombre: CtrAgregarSupervision	
Tipo de clase: controladora	
Atributo	Tipo
gtrUsuario	GtrUsuario
Responsabilidades	
Nombre:	CtrAgregarSupervision()
Descripción:	Operación que devuelve una instancia de la clase.
Nombre:	getGtrUsuario()
Descripción:	Operación que devuelve una instancia de la clase GtrUsuario.
Nombre:	setGtrUsuario(GtrUsuario gtrUsuario)
Descripción:	Operación que establece una instancia de la clase GtrUsuario.
Nombre:	onSubmit(Object command, BindException errors)
Descripción:	Operación que intercepta las peticiones a la vista vAgregarSupervision.jsp procesa la petición y luego devuelve los datos a mostrar en dicha vista. Los datos enviados desde esta vista que contienen un supervisor y un usuario son interceptados por este método, el cual se encarga de hacerlos persistentes.

Nombre: CtrBuscar	
Tipo de clase: controladora	
Atributo	Tipo
gtrUsuario	GtrUsuario
Responsabilidades	
Nombre:	CtrBuscar()
Descripción:	Operación que devuelve una instancia de la clase.

Nombre:	getGtrUsuario()
Descripción:	Operación que devuelve una instancia de la clase GtrUsuario.
Nombre:	setGtrUsuario(GtrUsuario gtrUsuario)
Descripción:	Operación que establece una instancia de la clase GtrUsuario.
Nombre:	onSubmit(HttpServletRequest request, HttpServletResponse response, Object command, BindException errors)
Descripción:	Operación que intercepta las peticiones a la vista vBuscar.jsp procesa la petición y luego redirige la petición hasta la vista. Las peticiones desde esta vista también son interceptadas por este método, el cual se ocupa de buscar un usuario en la base de datos y devolverlo a la vista.

Nombre: CtrEliminarSupervision	
Tipo de clase: controladora	
Atributo	Tipo
gtrUsuario	GtrUsuario
Responsabilidades	
Nombre:	CtrEliminarSupervision ()
Descripción:	Operación que devuelve una instancia de la clase.
Nombre:	getGtrUsuario()
Descripción:	Operación que devuelve una instancia de la clase GtrUsuario.
Nombre:	setGtrUsuario(GtrUsuario gtrUsuario)
Descripción:	Operación que establece una instancia de la clase GtrUsuario.
Nombre:	onSubmit(Object command, BindException errors)
Descripción:	Operación que intercepta las peticiones a la vista vEliminarSupervision.jsp procesa la petición y luego devuelve una lista de usuarios asociadas al supervisor en cuestión. Las solicitudes desde esta vista también son atendidas por este método, el cual tiene la tarea de actualizar una lista de usuarios correspondientes al supervisor.

Nombre: CtrSupervisor	
Tipo de clase: controladora	
Atributo	Tipo
gtrUsuario	GtrUsuario
Responsabilidades	
Nombre:	CtrSupervisor ()
Descripción:	Operación que devuelve una instancia de la clase.
Nombre:	getGtrUsuario()
Descripción:	Operación que devuelve una instancia de la clase GtrUsuario.
Nombre:	setGtrUsuario(GtrUsuario gtrUsuario)
Descripción:	Operación que establece una instancia de la clase GtrUsuario.
Nombre:	handleRequestInternal(HttpServletRequest request, HttpServletResponse response)
Descripción:	Operación que intercepta las peticiones a la vista vSupervisor.jsp procesa la petición y devuelve una lista de usuarios que pueden ser supervisados por el supervisor en cuestión.

Nombre: CtrCredenciales	
Tipo de clase: controladora	
Atributo	Tipo
gtrUsuario	GtrUsuario
Responsabilidades	
Nombre:	CtrCredenciales ()
Descripción:	Operación que devuelve una instancia de la clase.
Nombre:	handleRequestInternal(HttpServletRequest arg0, HttpServletResponse arg1)
Descripción:	Operación que intercepta las peticiones a la vista vAutenticacion.jsp e instantáneamente redirige la petición hasta la vista.

Nombre: CtrAutenticacion	
Tipo de clase: controladora	
Atributo	Tipo
gtrUsuario	GtrUsuario
Responsabilidades	
Nombre:	CtrAutenticacion ()
Descripción:	Operación que devuelve una instancia de la clase.
Nombre:	getGtrUsuario()
Descripción:	Operación que devuelve una instancia de la clase GtrUsuario.
Nombre:	setGtrUsuario(GtrUsuario gtrUsuario)
Descripción:	Operación que establece una instancia de la clase GtrUsuario.
Nombre:	handle(HttpServletRequest request, HttpServletResponse response, Object command, BindException errors)
Descripción:	Operación que intercepta la petición que viene desde la vista de autenticación procesa las credenciales del usuario y las guarda en la Session del usuario actual. Una vez conocido el usuario esta operación se encarga de buscar los datos asociados al mismo para mostrarlos en la vista principal. Finalmente la petición realizada es redirigida a la vista principal.

Nombre: CtrMarcos	
Tipo de clase: controladora	
Atributo	Tipo
gtrUsuario	GtrUsuario
Responsabilidades	
Nombre:	CtrMarcos ()
Descripción:	Operación que devuelve una instancia de la clase.
Nombre:	encabezado(HttpServletRequest req, HttpServletResponse resp)
Descripción:	Operación que intercepta las peticiones a la vista vEncabezado.jsp y redirige la petición hasta dicha vista.

Nombre:	menu(HttpServletRequest req, HttpServletResponse resp)
Descripción:	Operación que intercepta las peticiones a la vista vMenu.jsp y redirige la petición hasta dicha vista.
Nombre:	contenido(HttpServletRequest req, HttpServletResponse resp)
Descripción:	Operación que intercepta las peticiones a la vista vContenido.jsp y redirige la petición hasta dicha vista.

Nombre: CtrRangoDias	
Tipo de clase: controladora	
Atributo	Tipo
Responsabilidades	
Nombre:	CtrRangoDias ()
Descripción:	Operación que devuelve una instancia de la clase.
Nombre:	onSubmit(HttpServletRequest request, HttpServletResponse response, Object commandParam, BindException errors)
Descripción:	Operación que intercepta las peticiones a la vista vSeleccionarDias.jsp y redirige la petición hasta dicha vista. Los datos de rango de días para un reporte enviados desde dicha vista son interceptados por este controlador e insertados en la Session.

Nombre: CtrReportes	
Tipo de clase: controladora	
Atributo	Tipo
gtrReportes	GtrReportes
Responsabilidades	
Nombre:	CtrReportes ()
Descripción:	Operación que devuelve una instancia de la clase.
Nombre:	getGtrReportes()
Descripción:	Operación que devuelve una instancia de la clase GtrReportes.
Nombre:	setGtrReportes(GtrReportes gtrReportes)

Descripción:	Operación que establece una instancia de la clase GtrReportes.
Nombre:	handle(HttpServletRequest request, HttpServletResponse response, Object command, BindException errors)
Descripción:	Operación encargada de procesar las peticiones dirigidas hacia la vista vReportes.jsp. Básicamente busca los reportes de consumo de la navegación por Internet de un usuario, los cuales son redirigidos hacia dicha vista.

Nombre: GtrExtraccionImpl	
Tipo de clase: gestora	
Atributo	Tipo
directorioFuenteCompactados	File
directorioFicherosAccProhibido	File
directorioDestinoLog	File
directorioDestinoLogPhb	File
exregNombreFicheros	Map<Integer, String>
exregFicherosComp	String
cantidadHilosDescompactadores	int
cantidadHilosExtractores	int
tiempoEsperaHiloDescompactadorPrinc	int
tiempoEsperaHiloActualizadorPrinc	int
Responsabilidades	
Nombre:	GtrExtraccionImpl()
Descripción:	Operación que devuelve una instancia de la clase.
Nombre:	setApplicationContext(ApplicationContext context)
Descripción:	Operación que se encarga de asignarle a la clase una instancia del contexto de la aplicación
Nombre:	getDirectorioDestinoLog()
Descripción:	Operación que devuelve el directorio donde se descomprimirán los ficheros logs de accesos permitidos que serán procesados.

Nombre:	setDirectorioDestinoLog(File directorioDestinoLog)
Descripción:	Operación que establece el directorio donde se descomprimirán los ficheros logs de accesos permitidos que serán procesados.
Nombre:	getDirectorioDestinoLogPhb()
Descripción:	Operación que devuelve el directorio donde se descomprimirán los ficheros logs de accesos prohibidos que serán procesados.
Nombre:	setDirectorioDestinoLogPhb(File directorioDestinoLogPhb)
Descripción:	Operación que establece el directorio donde se descomprimirán los ficheros logs de accesos prohibidos que serán procesados.
Nombre:	getDirectorioFuenteCompactados()
Descripción:	Operación que devuelve el directorio donde se encuentran comprimidos los ficheros logs de accesos permitidos que serán procesados.
Nombre:	setDirectorioFuenteCompactados(File directorioFuenteCompactados)
Descripción:	Operación que establece el directorio donde se encuentran comprimidos los ficheros logs de accesos permitidos que serán procesados.
Nombre:	getDirectorioFicherosAccProhibido()
Descripción:	Operación que devuelve el directorio donde se encuentran comprimidos los ficheros logs de accesos prohibidos que serán procesados.
Nombre:	setDirectorioFicherosAccProhibido(File directorioFicherosAccProhibido)
Descripción:	Operación que establece el directorio donde se encuentran comprimidos los ficheros logs de accesos prohibidos que serán procesados.
Nombre:	getExregNombreFicheros()
Descripción:	Operación que devuelve una expresión regular para los nombres de los ficheros logs descomprimidos.
Nombre:	setExregNombreFicheros(Map<Integer, String> exregNombreFicheros)
Descripción:	Operación que establece una expresión regular para los nombres de los ficheros logs descomprimidos.
Nombre:	getExregFicherosComp()
Descripción:	Operación que devuelve una expresión regular que define los nombres de los ficheros

	logs comprimidos.
Nombre:	setExregFicherosComp(String exregFicherosComp)
Descripción:	Operación que establece una expresión regular que define los nombres de los ficheros logs comprimidos.
Nombre:	getCantidadHilosDescompactadores()
Descripción:	Operación que devuelve la cantidad de hilos encargados de descomprimir los ficheros logs.
Nombre:	setCantidadHilosDescompactadores(int cantidadHilosDescompactadores)
Descripción:	Operación que establece la cantidad de hilos encargados de descomprimir los ficheros logs.
Nombre:	getCantidadHilosExtractores()
Descripción:	Operación que devuelve la cantidad de hilos encargados de actualizar la base de datos con la información contenida en los ficheros logs descomprimidos.
Nombre:	setCantidadHilosExtractores(int cantidadHilosExtractores)
Descripción:	Operación que establece la cantidad de hilos encargados de actualizar la base de datos con la información contenida en los ficheros logs descomprimidos.
Nombre:	getTiempoEsperaHiloActualizadorPrinc()
Descripción:	Operación que devuelve el tiempo de espera del hilo principal actualizador de la base de datos por los demás hilos actualizadores.
Nombre:	setTiempoEsperaHiloActualizadorPrinc(int tiempoEsperaHiloActualizadorPrinc)
Descripción:	Operación que establece el tiempo de espera del hilo principal actualizador de la base de datos por los demás hilos actualizadores.
Nombre:	getTiempoEsperaHiloDescompactadorPrinc()
Descripción:	Operación que devuelve el tiempo de espera del hilo principal encargado de descomprimir los ficheros por los demás hilos actualizadores.
Nombre:	setTiempoEsperaHiloDescompactadorPrinc(int tiempoEsperaHiloDescompactadorPrinc)
Descripción:	Operación que establece el tiempo de espera del hilo principal encargado de descomprimir los ficheros por los demás hilos actualizadores.
Nombre:	crearDirectorios()

Descripción:	Operación que asegura la creación de todos los directorios de ficheros establecidos.
Nombre:	eliminarDirectorios()
Descripción:	Operación que asegura que todos los directorios establecidos sean vaciados al final del proceso.
Nombre:	iniciarHilos(TipoHilo tipo, ThreadGroup grupo, MonitorExtraccion monitorExtraccion, MonitorActualizacion monitorActualizacion, int prioridad, int cantHilos)
Descripción:	Operación que inicia todos los hilos necesarios para el proceso de descomprimir los ficheros y los hilos necesarios para el proceso de actualizar la base de datos con la información obtenida de los logs.
Nombre :	controlarDescompatado()
Descripción:	Operación que controla el proceso de descomprimir los ficheros logs.
Nombre :	controlarActualizacion()
Descripción:	Operación que controla el proceso de actualización de la base de datos con la información obtenida de los ficheros logs.

Nombre: GestorLogImpl	
Tipo de clase: gestora	
Atributo	Tipo
tipoClasificacion	Clasificacion
clasificacion	Map<Integer, String>
daoUsuario	DaoUsuario
daoDireccionIp	DaoDireccionIp
daoUrlDirecotrio	DaoUrlDirecotrio
daoUrlSubDirectorio	DaoUrlSubDirectorio
daoUrlCompleta	DaoUrlCompleta
daoLog	DaoLog
daoClasificacionUrl	DaoClasificacionUrl
Clasificacion	enum
Responsabilidades	

Nombre:	GestorLogImpl()
Descripción:	Operación que devuelve una instancia de la clase.
Nombre:	GestorLogImpl(Clasificacion tipoClasificacion)
Descripción:	Constructor de la clase que recibe por parámetros un tipo de clasificación de log y devuelve una instancia de la clase.
Nombre:	getTipoClasificacion()
Descripción:	Operación que devuelve el tipo de clasificación del fichero que se va a procesar.
Nombre:	setTipoClasificacion(Clasificacion tipoClasificacion)
Descripción:	Operación que establece el tipo de clasificación del fichero que se va a procesar.
Nombre:	getClasificacion()
Descripción:	Operación que devuelve los nombres de las posibles clasificaciones que pueden tener los ficheros logs.
Nombre:	setClasificacion(Map<Integer, String> clasificacion)
Descripción:	Operación que establece los nombres de las posibles clasificaciones que pueden tener los ficheros logs.
Nombre:	getDaoDireccionIp()
Descripción:	Operación que devuelve una instancia de la clase DaoDireccionIpImpl.
Nombre:	setDaoDireccionIp(DaoDireccionIp daoDireccionIp)
Descripción:	Operación que establece una instancia de la clase DaoDireccionIpImpl.
Nombre:	getDaoUrlDirecotrio()
Descripción:	Operación que devuelve una instancia de la clase DaoUrlDirecotrioImpl.
Nombre:	setDaoUrlDirecotrio(DaoUrlDirecotrio daoUrlDirecotrio)
Descripción:	Operación que establece una instancia de la clase DaoUrlDirecotrioImpl.
Nombre:	getDaoUsuario()
Descripción:	Operación que devuelve una instancia de la clase DaoUsuarioImpl.
Nombre:	setDaoUsuario(DaoUsuario daoUsuario)
Descripción:	Operación que establece una instancia de la clase DaoUsuarioImpl.
Nombre:	getDaoLog()
Descripción:	Operación que devuelve una instancia de la clase DaoLogImpl.

Nombre:	setDaoLog(DaoLog daoLog)
Descripción:	Operación que establece una instancia de la clase DaoLogImpl.
Nombre:	getDaoClasificacionUrl()
Descripción:	Operación que devuelve una instancia de la clase DaoClasificacionUrlImpl.
Nombre:	setDaoClasificacionUrl(DaoClasificacionUrl daoClasificacionUrl)
Descripción:	Operación que establece una instancia de la clase DaoClasificacionUrlImpl.
Nombre:	actualizarLogs(LogDominio logDominio)
Descripción:	Operación que actualiza los logs de navegación en la base de datos.

Nombre: GtrUsuarioImpl	
Tipo de clase:gestora	
Atributo	Tipo
daoUsuario	DaoUsuario
daoSupervision	DaoSupervision
Responsabilidades	
Nombre:	obtenerUsuario(String usuario)
Descripción:	Operación que devuelve los datos almacenados en la base de datos referentes a un usuario.
Nombre:	salvarActualizarUsuario(Usuario usuario)
Descripción:	Operación que salva o actualiza los datos de un usuario en la base de datos.
Nombre:	getDaoUsuario()
Descripción:	Operación que devuelve una instancia de la clase DaoUsuarioImpl.
Nombre:	setDaoUsuario(DaoUsuario daoUsuario)
Descripción:	Operación que establece una instancia de la clase DaoUsuarioImpl.
Nombre:	getDaoSupervision()
Descripción:	Operación que devuelve una instancia de la clase DaoSupervisionImpl.
Nombre:	setDaoSupervision(DaoSupervision daoSupervision)
Descripción:	Operación que establece una instancia de la clase DaoSupervisionImpl.
Nombre:	eliminarSubordinados(String usuarioSupervisor, List<String> usuariosSubordinados)

Descripción:	Operación que elimina una lista de usuarios a un supervisor. Ambas informaciones son pasadas por parámetro a la operación.
Nombre:	obtenerSubordinados(String usuarioSupervisor)
Descripción:	Operación que devuelve una lista de subordinados referentes al supervisor pasado por parámetros.

Nombre: GtrReportesImpl	
Tipo de clase: gestora	
Atributo	Tipo
daoConsumo	DaoConsumo
Responsabilidades	
Nombre:	getDaoConsumo()
Descripción:	Operación que devuelve una instancia de la clase DaoConsumo.
Nombre:	setDaoConsumo(DaoConsumo daoConsumo)
Descripción:	Operación que establece una instancia de la clase DaoConsumo.
Nombre:	obtenerConsumo(Map<String, String> parameters)
Descripción:	Operación que devuelve una instancia de la clase DmdReporte que contiene estadísticas de consumo del usuario.

Nombre: ClasificacionUrl	
Tipo de clase: entidad	
Atributo	Tipo
serialVersionUID	Long
idClasificacion	Integer
clasificacion	String
urlDirectorios	List<UrlDirectorio>
Responsabilidades	
Nombre:	ClasificacionUrl()
Descripción:	Operación que devuelve una instancia de la clase.

Nombre:	ClasificacionUrl(Integer idClasificacion, String clasificacion)
Descripción:	Operación que devuelve una instancia de la clase con el nombre de la clasificación.
Nombre:	ClasificacionUrl(Integer idClasificacion, String clasificacion, List<UrlDirectorio> urlDirectorios)
Descripción:	Operación que devuelve una instancia de la clase con el nombre de la clasificación y una lista de las URLs que tienen dicha clasificación asociados a la misma.
Nombre:	getIdClasificacion()
Descripción:	Operación que devuelve el identificador correspondiente al nombre de la clasificación.
Nombre:	setIdClasificacion(Integer idClasificacion)
Descripción:	Operación que establece el identificador correspondiente al nombre de la clasificación.
Nombre:	getClasificacion()
Descripción:	Operación que devuelve el nombre de la clasificación.
Nombre:	setClasificacion(String clasificacion)
Descripción:	Operación que establece el nombre de la clasificación.
Nombre:	getUrlDirectorios()
Descripción:	Operación que devuelve la lista de Urls que contienen la clasificación establecida en esta clase.
Nombre:	setUrlDirectorios(List<UrlDirectorio> urlDirectorios)
Descripción:	Operación que establece la lista de Urls que contienen la clasificación establecida en esta clase.

Nombre: DireccionIp	
Tipo de clase: entidad	
Atributo	Tipo
serialVersionUID	Long
idIp	Integer
Ip	String
Logs	List<Log>
Responsabilidades	

Nombre:	DireccionIp()
Descripción:	Operación que devuelve una instancia de la clase.
Nombre:	DireccionIp(Integer idIp, String ip)
Descripción:	Operación que devuelve una instancia de la clase con una dirección IP asociada a la misma.
Nombre:	DireccionIp(Integer idIp, String ip, List<Log> logs)
Descripción:	Operación que devuelve una instancia de la clase con una dirección IP y una lista de logs relacionados a esta IP asociados a la misma.
Nombre:	Integer getIdIp()
Descripción:	Operación que devuelve el identificador asociado a la dirección IP establecida en la clase.
Nombre:	setIdIp(Integer idIp)
Descripción:	Operación que establece el identificador asociado a la dirección IP.
Nombre:	getIp()
Descripción:	Operación que devuelve la dirección IP asociada a la clase.
Nombre:	setIp(String ip)
Descripción:	Operación que establece la dirección IP asociada a la clase.
Nombre:	getLogs()
Descripción:	Operación que devuelve la lista de logs asociados a esta IP.
Nombre:	setLogs(List<Log> logs)
Descripción:	Operación que establece la lista de logs asociados a esta IP.

Nombre: Log	
Tipo de clase: entidad	
Atributo	Tipo
serialVersionUID	Long
Id	Integer
urlDirectorio	UrlDirectorio
direccionIp	DireccionIp

usuario	Usuario
Dia	Date
consumo	Long
Responsabilidades	
Nombre:	Log()
Descripción:	Operación que devuelve una instancia de la clase.
Nombre:	Log(Integer id, UrlDirectorio urlDirectorio, DireccionIp direccionIp, Usuario usuario, Date dia, Long consumo)
Descripción:	Operación que devuelve una instancia de la clase con una URL, una dirección IP, un usuario y un día de acceso asociados a la misma.
Nombre:	getId()
Descripción:	Operación que devuelve el identificador del log.
Nombre:	setId(Integer id)
Descripción:	Operación que establece el identificador del log.
Nombre:	UrlDirectorio getUrlDirectorio()
Descripción:	Operación que devuelve la URL asociada al log.
Nombre:	setUrlDirectorio(UrlDirectorio urlDirectorio)
Descripción:	Operación que establece la URL asociada al log.
Nombre:	getDireccionIp()
Descripción:	Operación que devuelve la dirección IP asociada al log.
Nombre:	setDireccionIp(DireccionIp direccionIp)
Descripción:	Operación que establece la dirección IP asociada al log.
Nombre:	getUsuario()
Descripción:	Operación que devuelve el usuario asociado al log.
Nombre:	setUsuario(Usuario usuario)
Descripción:	Operación que establece el usuario asociado al log.
Nombre:	getDia()
Descripción:	Operación que devuelve el día de acceso a la URL asociada al log.
Nombre:	setDia(Date dia)

Descripción:	Operación que establece el día de acceso a la URL asociada al log.
Nombre:	getConsumo()
Descripción:	Operación que devuelve el consumo asociado a la URL relacionada con el log.
Nombre:	setConsumo(Long consumo)
Descripción:	Operación que establece el consumo asociado a la URL relacionada con el log.

Nombre: UrlDirectorio	
Tipo de clase: entidad	
Atributo	Tipo
serialVersionUID	Long
idDirectorio	Integer
urlDirectorio	UrlDirectorio
clasificacionUrl	ClasificacionUrl
url	String
urlDirectorios	List<UrlDirectorio>
Logs	List<Log>
Responsabilidades	
Nombre:	UrlDirectorio()
Descripción:	Operación que devuelve una instancia de la clase.
Nombre:	UrlDirectorio(Integer idDirectorio, ClasificacionUrl clasificacionUrl, String url)
Descripción:	Operación que devuelve una instancia de la clase con una URL y una clasificación asociada a la misma.
Nombre:	UrlDirectorio(Integer idDirectorio, UrlDirectorio urlDirectorio, ClasificacionUrl clasificacionUrl, String url, List<UrlDirectorio> urlDirectorios, List<Log> logs)
Descripción:	Operación que devuelve una instancia de la clase con una URL, una clasificación asociada a la misma, una URL principal, una lista de subdirecciones correspondientes a la URL principal y una lista de logs asociados a la misma.
Nombre:	getIdDirectorio()
Descripción:	Operación que devuelve un identificador asociado a la URL establecida.

Nombre:	setIdDirectorio(Integer idDirectorio)
Descripción:	Operación que establece un identificador asociado a la URL establecida.
Nombre:	getUrlDirectorio()
Descripción:	Operación que devuelve la URL principal asociada la URL.
Nombre:	setUrlDirectorio(UriDirectorio urlDirectorio)
Descripción:	Operación que establece la URL principal asociada la URL.
Nombre:	getClasificacionUrl()
Descripción:	Operación que devuelve la clasificación asociada a la URL establecida.
Nombre:	setClasificacionUrl(ClasificacionUrl clasificacionUrl)
Descripción:	Operación que establece la clasificación asociada a la URL establecida.
Nombre:	getUrl()
Descripción:	Operación que devuelve la URL asociada a la clase.
Nombre:	setUrl(String url)
Descripción:	Operación que establece la URL asociada a la clase.
Nombre:	getUrlDirectorios()
Descripción:	Operación que devuelve la lista de subdirecciones asociadas a la URL.
Nombre:	setUrlDirectorios(List<UriDirectorio> urlDirectorios)
Descripción:	Operación que establece la lista de subdirecciones asociadas a la URL.
Nombre:	getLogs()
Descripción:	Operación que devuelve la lista de logs asociados a la URL establecida.
Nombre:	setLogs(List<Log> logs)
Descripción:	Operación que establece la lista de logs asociados a la URL establecida.

Nombre: Usuario	
Tipo de clase: entidad	
Atributo	Tipo
serialVersionUID	Long
idUsuario	Integer
usuario	String

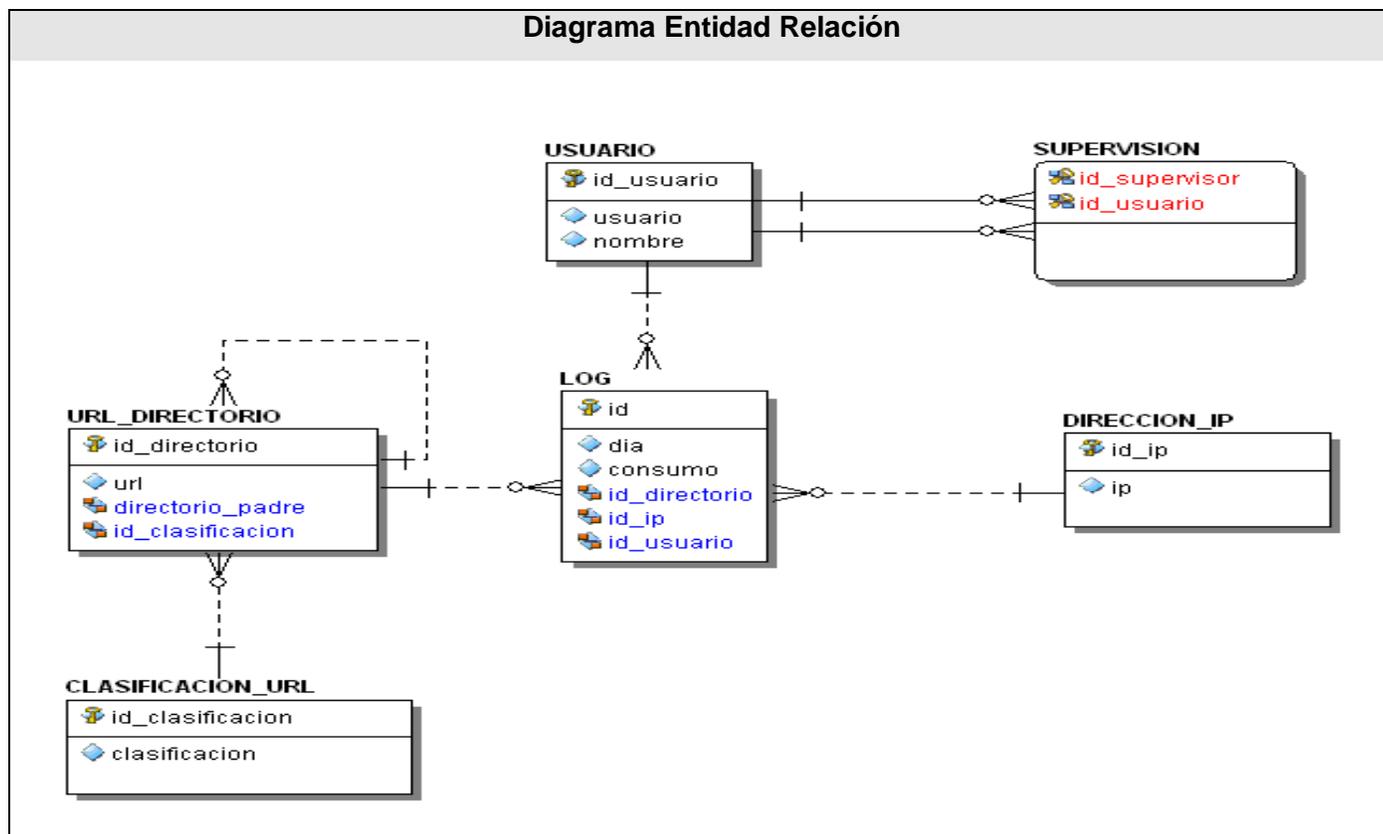
nombre	String
usuariosForIdUsuario	List<Usuario>
usuariosForIdSupervisor	List<Usuario>
Logs	List<Log>
Responsabilidades	
Nombre:	Usuario()
Descripción:	Operación que devuelve una instancia de la clase.
Nombre:	Usuario(Integer idUsuario, String usuario, String nombre)
Descripción:	Operación que devuelve una instancia de la clase con un usuario y un nombre del usuario asociado a la misma.
Nombre:	Usuario(Integer idUsuario, String usuario, String nombre, List<Usuario> usuariosForIdUsuario, List<Usuario> usuariosForIdSupervisor, List<Log> logs)
Descripción:	Operación que devuelve una instancia de la clase con un usuario, un nombre del usuario, una lista de usuarios que serán supervisados por este usuario, una lista de supervisores y una lista de logs asociados a la misma.
Nombre:	getIdUsuario()
Descripción:	Operación que devuelve el identificador asociado al usuario establecido.
Nombre:	setIdUsuario(Integer idUsuario)
Descripción:	Operación que establece el identificador asociado al usuario establecido.
Nombre:	getUsuario()
Descripción:	Operación que devuelve el usuario asociado a la clase.
Nombre:	setUsuario(String usuario)
Descripción:	Operación que establece el usuario asociado a la clase.
Nombre:	getNombre()
Descripción:	Operación que devuelve el nombre del usuario asociado a la clase.
Nombre:	setNombre(String nombre)
Descripción:	Operación que establece el nombre del usuario asociado a la clase.
Nombre:	getUsuariosForIdUsuario()
Descripción:	Operación que devuelve la lista de usuarios que serán supervisados por el usuario

	asociado a la clase.
Nombre:	setUsuariosForIdUsuario(List<Usuario> usuariosForIdUsuario)
Descripción:	Operación que establece la lista de usuarios que serán supervisados por el usuario asociado a la clase.
Nombre:	getUsuariosForIdSupervisor()
Descripción:	Operación que devuelve la lista de supervisores del usuario asociado a la clase.
Nombre:	setUsuariosForIdSupervisor(List<Usuario> usuariosForIdSupervisor)
Descripción:	Operación que establece la lista de supervisores del usuario asociado a la clase.
Nombre:	getLogs()
Descripción:	Operación que devuelve la lista de logs asociados al usuario asignado a la clase.
Nombre:	setLogs(List<Log> logs)
Descripción:	Operación que establece la lista de logs asociados al usuario asignado a la clase.

#### **3.3.4. Diseño de la BD**

El diseño de una base de datos es un proceso complejo que se puede controlar si se descompone el problema en subproblemas y se resuelve cada uno de estos por separado, el diseño de una base de datos se descompone en diseño conceptual, diseño lógico y diseño físico. Un buen diseño de base de datos debe reflejar la estructura del problema en el mundo real, debe ser capaz de representar todos los datos esperados y mantener la integridad de los mismos incluso con el paso del tiempo, además tiene que evitar el almacenamiento de información redundante así como proporcionar un acceso eficaz a los datos.

### 3.3.4.1. Diagrama Entidad Relación de la Base de Datos



### 3.3.4.2. Descripción de las tablas

Nombre: clasificacion_url		
Descripción: Tabla que contiene la información de las clasificaciones de las URLs.		
Atributo	Tipo	Descripción
id_clasificacion	integer	Identificador de la clasificación de la URL.
Clasificacion	varchar(50)	Nombre de la clasificación de las URLs.

Nombre: direccion_ip		
Descripción: Tabla que contiene la información de las direcciones IP.		
Atributo	Tipo	Descripción
id_ip	integer	Identificador de la dirección IP

ip	varchar(50)	Dirección IP desde donde el usuario accede a Internet.
----	-------------	--

<b>Nombre:</b> usuario		
<b>Descripción:</b> Tabla que contiene la información de los usuarios.		
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
id_usuario	integer	Identificador del usuario
usuario	varchar(50)	Usuario del dominio
nombre	varchar(50)	Nombre de la persona

<b>Nombre:</b> log		
<b>Descripción:</b> Tabla que contiene la información de los logs de navegación por Internet de los usuarios.		
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
id	integer	Identificador del log.
día	timestamp	Instante de tiempo en que se registra el log de acceso a la URL.
consumo	Bigint	Consumo de bytes en la URL accedida.
id_directorio	integer	Identificador de la URL a la que accedió el usuario.
id_ip	integer	Identificador de la dirección IP desde donde accedió el usuario a Internet.
id_usuario	integer	Identificador del usuario que realizó el acceso a la URL registrada.

<b>Nombre:</b> supervisión		
<b>Descripción:</b> Tabla que contiene la información de los supervisores y los usuarios que estos tienen asignados.		
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
id_supervisor	integer	Identificador del supervisor.

id_usuario	integer	Identificador del usuario que será supervisado.
------------	---------	---

Nombre: url_directorio		
Descripción: Tabla que contiene la información de las URLs a las que accedió el usuario.		
Atributo	Tipo	Descripción
id_directorio	integer	Identificador de la URL visitada por el usuario.
url	varchar(1000)	La URL se divide en dos cadenas. La primera contiene la URL principal del sitio accedido por el usuario, y las segunda el resto de la URL. En este campo se guarda una de las dos cadenas.
directorio_padre	integer	En el caso de que se haya guardado en el campo anterior el resto de la URL en lugar de la URL principal, este campo contiene el id de un registro que contiene la URL principal del sitio correspondiente al resto de la URL que se almacena en el registro actual.
id_clasificacion	integer	Identificador de la clasificación correspondiente a la URL guardada.

### 3.4. Conclusiones

En este capítulo se ubica al lector de cómo el análisis consiste en obtener una visión del sistema, de modo que sólo se interesa por los requisitos funcionales. Por otro lado el diseño es un refinamiento del análisis que tiene en cuenta los requisitos no funcionales, o sea cómo cumple el sistema sus objetivos. Se construyeron los diagramas de interacción mediante diagramas de secuencia. Se logró alcanzar el resultado final más importante de este flujo de trabajo que es el modelo de diseño que consiste en colaboraciones de clases, que pueden ser agrupadas en paquetes y subsistemas.

## **CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBA**

---

### **4.1. Introducción**

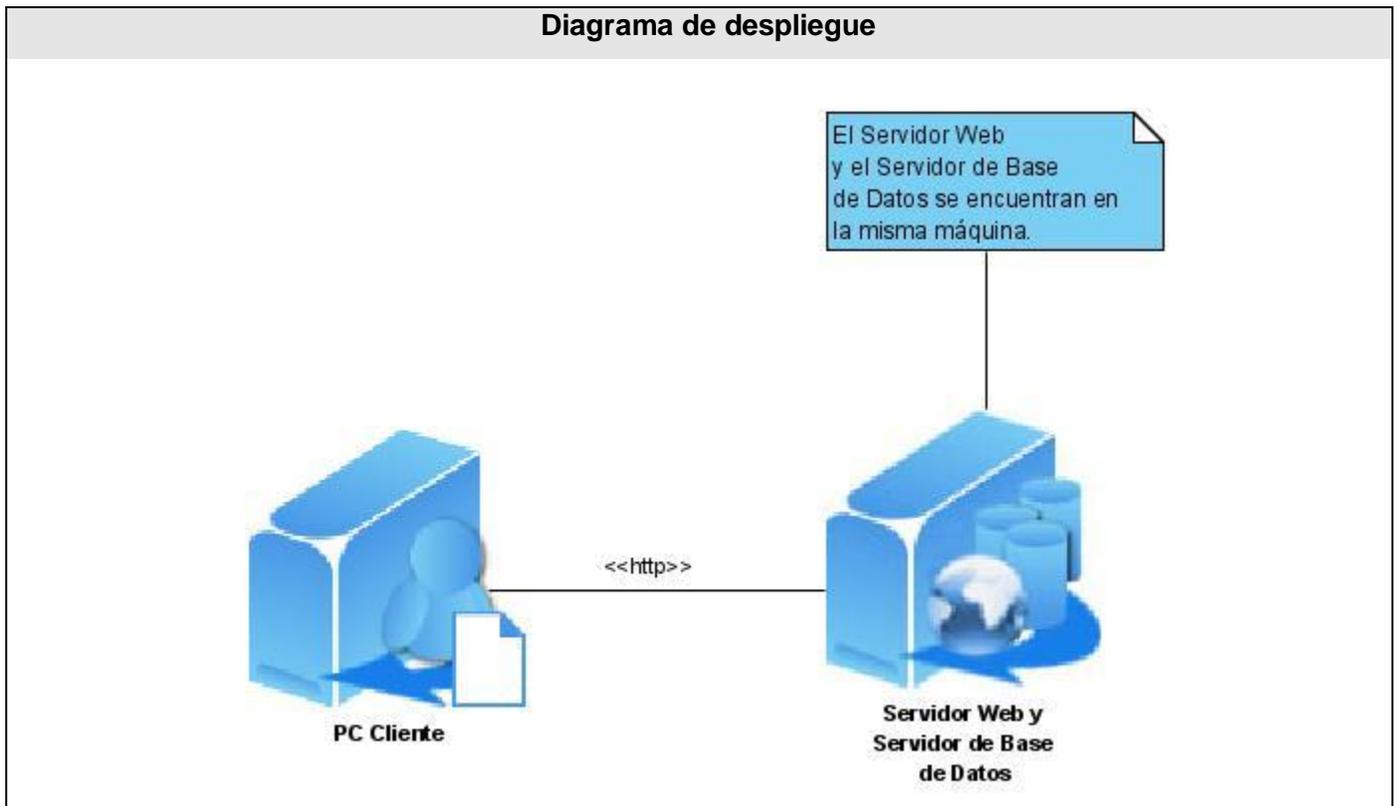
La implementación comienza con el resultado del diseño. Este capítulo trata sobre los flujos de trabajo Implementación y Prueba, se representan los diagramas de despliegue y de componentes de los cuales se brinda además un sencillo concepto que facilite un mejor entendimiento al lector y se explican algunos detalles relacionados con las pruebas como por ejemplo que el modelo de implementación es la entrada principal de las etapas de prueba que siguen a la implementación, que en la etapa de prueba cada construcción generada durante la implementación es sometida a pruebas de integración y es posible que a pruebas de sistema.

### **4.2. Implementación**

En el flujo de trabajo Implementación se implementan las clases y objetos en ficheros fuente, binarios y ejecutables, y se realizan las pruebas de unidad. Todos los elementos implementados forman el modelo de implementación, se integran de forma incremental con el objetivo de que en caso que ocurran fallos sean más fáciles de detectar y los componentes se prueban más a fondo. El resultado final de este flujo de trabajo es un sistema ejecutable.

#### **4.2.1. Diagrama de Despliegue**

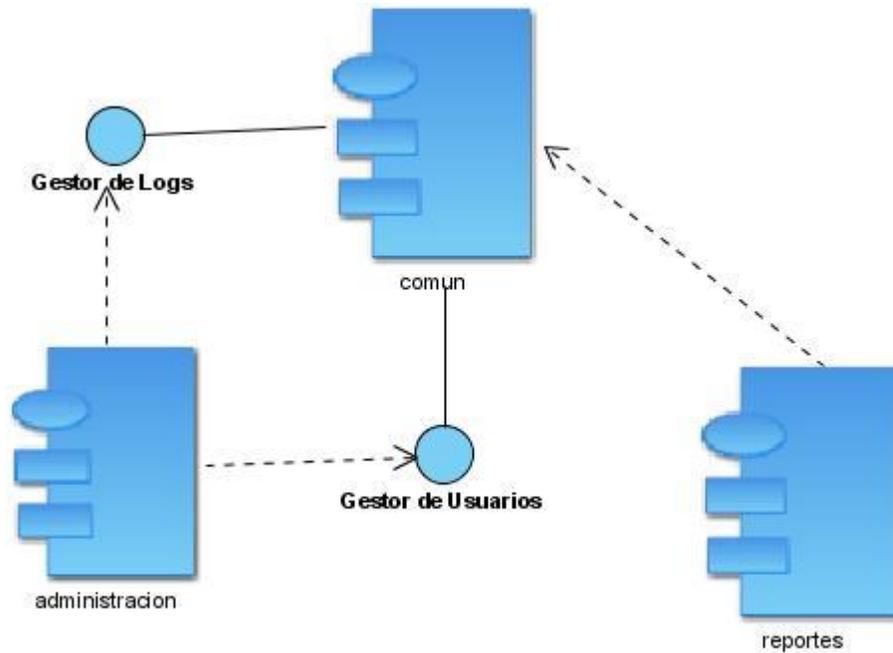
El diagrama de despliegue se utiliza para modelar la vista de despliegue estática de un sistema, muestra las relaciones físicas entre los componentes hardware y software, es decir, la configuración de los elementos de procesamiento en tiempo de ejecución y los componentes software. El diagrama de despliegue muestra además los nodos que participan en la ejecución y las relaciones de dependencia y asociación que se establecen entre ellos. Los nodos pueden ser servidores, procesadores o dispositivos.

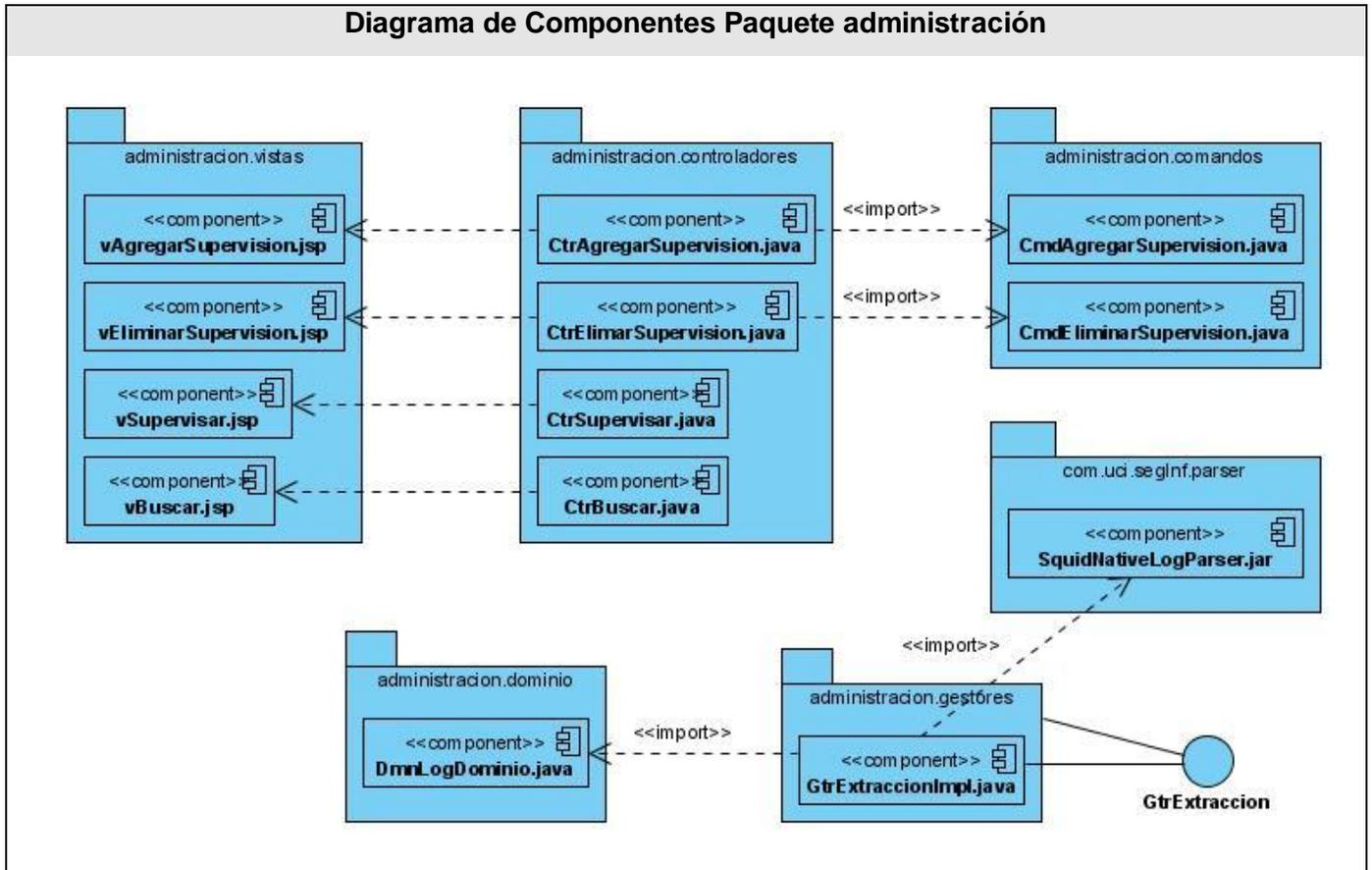


#### 4.2.2. Diagrama de Componentes

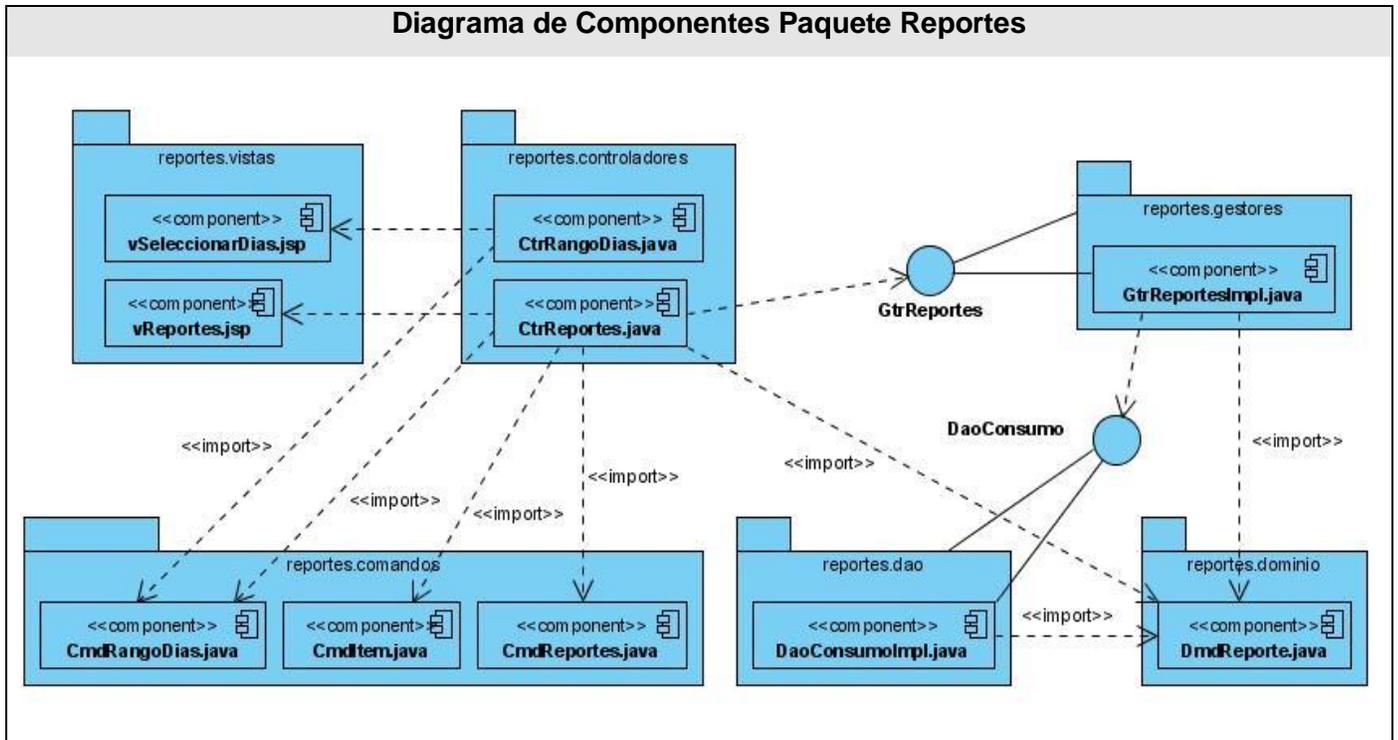
Los diagramas de componentes describen los elementos físicos del sistema y sus relaciones. Muestran las opciones de realización incluyendo código fuente, binario y ejecutable. Los componentes representan todos los tipos de elementos software que entran en la fabricación de aplicaciones informáticas. Desde el punto de vista del diagrama de componentes se tienen en consideración los requisitos relacionados con la facilidad de desarrollo, la gestión del software, la reutilización, y las restricciones impuestas por los lenguajes de programación y las herramientas utilizadas en el desarrollo. Un diagrama de componentes muestra clasificadores de componentes, las clases definidas en ellos, y las relaciones entre ellas.

Diagrama de Componentes Subsistema Control de Navegación









### 4.3. Modelo de Prueba

El objetivo de las pruebas del sistema es verificar el software para comprobar si el mismo cumple los requisitos. Se pueden desarrollar varios tipos de pruebas en función de los objetivos de las mismas, entre ellas, las pruebas funcionales de aplicación con interfaces gráficas que verifican si el software ofrece a los actores la funcionalidad recogida en su especificación.

#### 4.3.1. Descripción de los casos de prueba

##### Caso de uso: CU\_Autenticar

Entrada	Resultados	Condiciones
El usuario inserta el nombre de usuario, la contraseña correcta y selecciona el botón "Aceptar".	El Sistema muestra la página principal de entrada al usuario.	El usuario debe tener una cuenta en el directorio activo de Windows (uci.cu).

Entrada	Resultados	Condiciones
El usuario inserta su nombre de usuario y una contraseña incorrecta y selecciona el botón "Aceptar".	El Sistema muestra el mensaje "usuario o contraseña incorrecta".	El usuario debe haber insertado su nombre de usuario y una contraseña incorrecta.

Entrada	Resultados	Condiciones
El usuario inserta el nombre de usuario incorrecto y la contraseña correcta.	El Sistema muestra el mensaje "usuario o contraseña incorrecta".	El usuario debe haber insertado un nombre de usuario incorrecto y su contraseña.

Entrada	Resultados	Condiciones
El usuario inserta su nombre de usuario y selecciona el botón "Aceptar".	El Sistema muestra el mensaje "No se permiten campos vacíos".	El usuario debe haber dejado el campo de contraseña vacío.

Entrada	Resultados	Condiciones
El usuario selecciona el botón "Aceptar".	El Sistema muestra el mensaje "No se permiten campos vacíos".	El usuario debe haber dejado vacíos todos los campos requeridos.

**Caso de uso: CU\_Gestionar Reporte**

Entrada	Resultados	Condiciones
El usuario selecciona la opción "Rango Días".	El Sistema muestra la interfaz correspondiente para seleccionar el rango de días	El usuario debe estar autenticado en el Sistema.

	para los que quiere obtener el reporte de navegación.	
El Usuario selecciona el rango de días y la opción "Enviar".	El Sistema muestra el mensaje "Cambios salvados exitosamente".	El rango de días especificado por el usuario es correcto.
El Usuario selecciona la opción "Reportes".	El Sistema despliega las opciones: "Consumo x URL", "Consumo x IP", "Consumo x Días" y "Consumo x Horas".	
El usuario selecciona la opción "Consumo x URL".	El Sistema muestra el reporte de navegación del usuario de acuerdo al criterio "Consumo x URL".	El usuario debe haber seleccionado un rango de días válido.
El sistema navega por los diferentes criterios del reporte obtenido.		

Entrada	Resultados	Condiciones
El usuario selecciona la opción "Rango Días".	El Sistema muestra la interfaz correspondiente para seleccionar el rango de días para los que quiere obtener el reporte de navegación.	El usuario debe estar autenticado en el Sistema.
El usuario selecciona la opción "Enviar" sin haber escogido el rango de días.	El Sistema muestra el mensaje "No se permiten campos vacíos".	El usuario debe haber seleccionado la opción "Enviar" sin haber escogido el rango de días.

Entrada	Resultados	Condiciones
El usuario selecciona la opción "Rango Días".	El Sistema muestra la interfaz correspondiente para seleccionar el rango de días para los que quiere obtener el reporte de navegación.	El usuario debe estar autenticado en el Sistema.
El usuario selecciona el rango de días y la opción "Enviar".	El Sistema muestra el mensaje "El rango no puede exceder de una semana".	El usuario debe haber escogido un rango de días superior a una semana.

Entrada	Resultados	Condiciones
El usuario selecciona la opción "Rango Días".	El Sistema muestra la interfaz correspondiente para seleccionar el rango de días para los que quiere obtener el reporte de navegación.	El usuario debe estar autenticado en el Sistema.
El usuario selecciona el rango de días y la opción "Enviar".	El Sistema muestra el mensaje "Introduzca un rango de días correcto".	El usuario debe haber escogido un día inicial posterior al día final.

**Caso de uso: CU\_Gestionar Supervisión**

Entrada	Resultados	Condiciones
El Administrador inserta el nombre de usuario del supervisor que desea buscar y selecciona la opción "Buscar".	El Sistema muestra la opción "Agregar".	El Administrador debe estar autenticado en el Sistema y haber seleccionado la opción "Asignar supervisión".
El Administrador selecciona la opción "Agregar".	El Sistema muestra la interfaz de búsqueda principal con el	El Administrador debe haber escrito correctamente el nombre de usuario del

	campo de supervisor completado con el nombre de usuario escogido por el Administrador del Sistema.	supervisor y haber seleccionado la opción "Buscar".
El Administrador inserta el nombre de usuario del usuario que desea sea supervisado y selecciona la opción "Buscar".	El Sistema muestra la opción "Agregar".	El Administrador debe haber escrito correctamente el nombre de usuario y haber seleccionado la opción "Buscar".
El Administrador selecciona la opción "Agregar"	El Sistema muestra la interfaz de búsqueda principal con el campo de usuario completado con el nombre de usuario escogido por el Administrador del Sistema.	El Administrador debe haber seleccionado la opción "Buscar".
El Administrador marca la opción "Enviar".	El Sistema muestra el mensaje "Datos guardados exitosamente".	El Administrador debe haber buscado nombres de usuarios diferentes en los campos supervisor y usuario.

Entrada	Resultados	Condiciones
El Administrador selecciona la opción "Enviar"	El Sistema muestra el mensaje "No se permiten campos vacíos".	El Administrador debe estar autenticado al Sistema, haber seleccionado la opción "Asignar supervisión" y no haber rellenado los campos de supervisor y usuario.

Entrada	Resultados	Condiciones
El Administrador inserta un	El Sistema muestra el	El Administrador debe haber

nombre de usuario incorrecto y selecciona la opción "Buscar".	mensaje "El usuario no existe".	especificado un nombre de usuario incorrecto.
---	---------------------------------	---

Entrada	Resultados	Condiciones
El Administrador inserta el nombre de usuario del Supervisor que desea buscar y selecciona la opción "Buscar "	El Sistema muestra la opción "Agregar".	El Administrador debe haber especificado un nombre de usuario correcto.
El Administrador selecciona la opción "Agregar"	El Sistema muestra la interfaz de búsqueda principal con el campo de supervisor completado con el nombre de usuario escogido por el Administrador del Sistema.	El Administrador debe haber seleccionado la opción "Buscar".
El Administrador cuando busca el nombre de usuario que desea sea supervisado inserta un usuario incorrecto y selecciona la opción "Buscar".	El Sistema muestra el mensaje "El usuario no existe".	El Administrador debe haber especificado un nombre de usuario incorrecto.

Entrada	Resultados	Condiciones
El Administrador inserta el nombre de usuario del supervisor que desea buscar y selecciona la	El Sistema muestra la opción "Agregar".	El Administrador debe haber especificado un nombre de usuario correcto.

opción “Buscar ”		
El Administrador selecciona la opción “Agregar”	El Sistema muestra la interfaz de búsqueda principal con el campo de supervisor completado con el nombre de usuario escogido por el Administrador del Sistema.	El Administrador debe haber seleccionado la opción “Buscar”.
El Administrador selecciona la opción “Enviar”.	El Sistema muestra el mensaje “No se permiten campos vacíos”.	El Administrador debe haber seleccionado la opción enviar sin haber rellenado el campo de usuario.

Entrada	Resultados	Condiciones
El Administrador inserta el nombre de usuario del supervisor que desea buscar y selecciona la opción “Buscar”.	El Sistema muestra la opción “Agregar”.	El Administrador debe estar autenticado al Sistema y haber seleccionado la opción “Eliminar supervisión”. Además debe haber especificado un nombre de usuario correcto.
El Administrador selecciona la opción “Agregar”	El Sistema muestra la interfaz de búsqueda principal con el campo de supervisor completado con el nombre de usuario escogido por el Administrador del Sistema.	El Administrador debe haber seleccionado la opción “Buscar”.
El Administrador selecciona la opción “Enviar”	El Sistema muestra la lista de usuarios que el supervisor tiene asignados.	El Supervisor debe tener asignados usuarios para supervisar.
El Administrador selecciona los usuarios que no desea sean analizados por el	El Sistema elimina los usuarios seleccionados de la lista de usuarios asignados al	El Administrador debe haber seleccionado al menos un usuario.

Supervisor y selecciona la opción "Actualizar".	Supervisor y muestra el mensaje "Datos procesados exitosamente".	
---	--	--

Entrada	Resultados	Condiciones
El Administrador inserta el nombre de usuario del supervisor que desea buscar y selecciona la opción "Buscar".	El Sistema muestra la opción "Agregar".	El Administrador debe estar autenticado al Sistema y haber seleccionado la opción "Eliminar supervisión". Además debe haber especificado un nombre de usuario correcto.
El Administrador selecciona la opción "Agregar"	El Sistema muestra la interfaz de búsqueda principal con el campo de supervisor completado con el nombre de usuario escogido por el Administrador del Sistema.	El Administrador debe haber seleccionado la opción "Buscar".
El Administrador selecciona la opción "Enviar"	El Sistema muestra el mensaje "No existen usuarios asignados".	El Supervisor no debe tener usuarios asignados para supervisar.

Entrada	Resultados	Condiciones
El Administrador inserta un usuario incorrecto y selecciona la opción "Buscar".	El Sistema muestra el mensaje "El usuario no existe".	El Administrador debe estar autenticado al Sistema y haber seleccionado la opción "Eliminar supervisión". Además el administrador debe haber especificado un nombre de usuario incorrecto.

Entrada	Resultados	Condiciones
El Administrador inserta el	El Sistema muestra la opción	El Administrador debe estar autenticado

nombre de usuario del supervisor que desea buscar y selecciona la opción "Buscar".	"Agregar".	al Sistema y haber seleccionado la opción "Eliminar supervisión". Además el administrador debe haber especificado un nombre de usuario correcto.
El Administrador selecciona la opción "Agregar"	El Sistema muestra la interfaz de búsqueda principal con el campo de supervisor completado con el nombre de usuario escogido por el Administrador del Sistema.	El Administrador debe haber seleccionado la opción "Buscar".
El Administrador selecciona la opción "Enviar"	El Sistema muestra la lista de usuarios que el supervisor tiene asignados.	El Supervisor debe tener asignados usuarios para supervisar.
El Administrador selecciona la opción "Actualizar".	El Sistema muestra el mensaje "Debe seleccionar al menos un usuario".	El Administrador selecciona la opción "Actualizar" sin antes seleccionar al menos un usuario de la lista mostrada.

**Caso de uso: CU\_Supervisar Navegación**

Entrada	Resultados	Condiciones
El Supervisor selecciona la opción "Supervisar".	El Sistema despliega la opción "Usuarios".	El Supervisor del Sistema debe estar autenticado en el Sistema como un Supervisor.
El Supervisor selecciona la opción "Usuarios".	El Sistema muestra la lista de usuarios que el Supervisor puede analizar.	El Supervisor debe tener usuarios asignados.
El Supervisor selecciona el usuario que desea analizar.	Ver caso de uso CU_Gestionar Reporte.	

Entrada	Resultados	Condiciones
El Supervisor selecciona la opción "Supervisar".	El Sistema despliega la opción "Usuarios".	El Supervisor del Sistema debe estar autenticado en el Sistema como un Supervisor.
El Supervisor selecciona la opción "Usuarios".	El Sistema muestra el mensaje: "No hay usuarios asignados".	El Supervisor no tiene usuarios asignados.

#### **4.4. Conclusiones**

En este capítulo se describe de forma detallada como se llevaron a cabo los flujos de Implementación y Prueba descritos por RUP mediante la utilización de los casos de prueba, los diagramas de componentes y de despliegue. Después de leer este capítulo el lector puede tener una idea de la importancia de las pruebas, que son una de las herramientas para asegurar la calidad de un producto y que ofrecen un gran valor al mismo descubriendo los errores que tienen los programas por mucho cuidado que se tenga a la hora de desarrollar la aplicación.

## **CONCLUSIONES**

---

Con la realización de este trabajo se ha logrado desarrollar un sistema informático que permite a los usuarios de las tecnologías de información y comunicaciones de la Universidad de las Ciencias Informáticas, consultar las trazas de la navegación en Internet de una manera eficiente y sencilla.

El Sistema de Reportes de la Navegación por Internet (SRNI), como se le ha denominado a este software, es capaz de extraer la información contenida en los ficheros log generados por el servidor Proxy de la Universidad y almacenarla en una base de datos que se actualiza diariamente de manera automática. El programa permite, a través de una Interfaz amigable, de forma rápida y sencilla que los usuarios puedan obtener reportes de cómo ha sido su navegación por Internet seleccionando previamente el período de tiempo que les interese. Los reportes que se realizan son dinámicos posibilitando que los usuarios obtengan detalles a fondo de cómo ha sido su navegación filtrando el tipo de reporte que quieren obtener y los tiempos de respuesta a las peticiones de los usuarios están optimizados para que la respuesta sea en el menor tiempo posible. El software brinda además la posibilidad de establecer permisos de supervisión, lo cual permitirá que los directivos de las diferentes áreas puedan revisar la navegación de los usuarios de su área, contribuyendo de esta manera a descentralizar el control que realiza la Dirección de Redes y Seguridad Informática sobre la navegación en Internet.

La implantación de este sistema debe disminuir la efectividad del robo de contraseñas, que es uno de los incidentes de seguridad informática más problemáticos en la UCI, debido a que el usuario podrá revisar constantemente las estadísticas de su navegación, pudiendo detectar a tiempo cualquier anomalía. Deben también disminuir considerablemente las visitas a los sitios Web restringidos tipificados en el Código de Ética para el uso de las Tecnologías en la UCI. El hecho de que los usuarios de la Universidad puedan consultar las trazas de su navegación en Internet y conozcan que las mismas pueden ser revisadas en cualquier momento por un superior, debe influir directamente en un mejor uso de los recursos informáticos que la Universidad ha puesto en manos de sus estudiantes y trabajadores.

Finalmente es válido destacar que el software desarrollado es multiplataforma debido a que puede instalarse en cualquier sistema operativo y que aunque el trabajo ha sido realizado para su aplicación en la Universidad de las Ciencias Informáticas, es completamente adaptable a otras instituciones para el control del acceso a Internet.

## **RECOMENDACIONES**

---

El acceso a Internet es un pilar importante en las instituciones pero a la vez el descontrol de la navegación en esta inmensa red es un tema preocupante para las empresas y en especial las universidades. La actividad del control de la navegación por Internet puede estar infiltrada o distribuida en muchas áreas de la informática, y con el surgimiento de las nuevas tecnologías cada vez se hace más extenso este campo. El presente trabajo solo ha rosado una parte de este tema. El Sistema de Reportes de la Navegación por Internet (SRNI) está diseñado para mostrar estadísticas de consumo de la navegación por Internet de los usuarios de la Universidad de las Ciencias Informáticas, aunque puede ser desplegado en otras instituciones. A pesar del dinamismo de sus reportes, pudiera pensarse en una solución más amigable de los mismos, mostrando resúmenes de la navegación a través de gráficos, que podrían ser interpretados más fáciles por un usuario final. El sistema no ofrece información en tiempo real debido a que los ficheros logs del servidor Squid utilizados en la actualización de la base de datos de SRNI solo pueden ser procesados hasta el día anterior, es por eso que recomendamos que en próximas versiones del sistema, la base de datos que contiene la fuente de información para la generación de reportes pueda ser actualizada en tiempo real por alguna herramienta que se integre con el Squid.

El estado de la cuota del usuario es una información importante que brinda el sistema inetCuotas de la UCI, información que está muy relacionada con nuestro sistema, por tanto sería bueno pensar en una integración futura de ambos sistemas.

Finalmente hay tareas que podrían realizarse en un futuro desde el SRNI; por ejemplo, la supervisión de la navegación de los usuarios es una de las funcionalidades que brinda el sistema, esta opción permite a cualquier supervisor revisar la navegación de los usuarios que les fueron asignados, una vez que un supervisor verifique que algún usuario haya accedido a sitios restringidos por la Universidad, podría dársele la posibilidad al supervisor de bloquear el acceso del usuario a Internet. En próximas versiones el sistema debe seguirse perfeccionando hasta alcanzar todas estas funcionalidades que se recomiendan.

## **REFERENCIAS BIBLIOGRÁFICAS**

---

BECERRIL, F. *Java a su alcance*, Litográfica Ingramex, 1998.

CYBERCENTER. *Analizador Estadístico para Sitios Web*, 2007]. Disponible en:

<http://www.cybercenter.cl/webtrends/log.html>

HTTP-PERU, C. *Soluciones TI para su Empresa "Llegó la hora del PostgreSQL"*, 2006. [2007]. Disponible en: <http://www.http-peru.com/postgresql1.php>

INFORMÁTICA, C. T. D. *Arquitectura de desarrollo*, 2006. [2007]. Disponible en:

[http://tic.csic.es/documentos/Arquitectura\\_desarrollo\\_framework\\_csic\\_feb06.pdf](http://tic.csic.es/documentos/Arquitectura_desarrollo_framework_csic_feb06.pdf)

JALÓN, J. G. D. *Aprenda Java como si estuviera en primero*, 1999.

MOTA, A. G. *Instalando y usando Webalizer*, 2006.

NETWORKS, X. *Estadísticas Webalizer*, 2007.

PALENZUELA, O. B. y D. G. PÉREZ. *Monitoreando el tráfico de Internet. Una solución al alcance de la mano.*, 2006. [2007]. Disponible en:

[http://www.informaticahabana.com/evento\\_virtual/files/seg034.doc](http://www.informaticahabana.com/evento_virtual/files/seg034.doc)

PALOS, J. A. *Persistencia de Objetos Java utilizando Hibernate*, 2007. [2007]. Disponible en:

SEGURMATICA. *¿Que es AAInternet?*, Segurmatica, 2007.

SOFTONIC, E. D. *Genera estadísticas a partir de logs*, 2007. [2007]. Disponible en:

<http://sawmill.softonic.com/>

SUÁREZ, J. A. *Análisis de ficheros log en GNU/Linux*.

VERA, K. L. *Ingeniería de Software – RUP - UML* 2006.

YANOVER, D. A. *Analiza a fondo el tráfico de tu sitio online.*, Edición 16: 47-48.

## **BIBLIOGRAFÍA**

---

- AGULLÓ, P. java.io: operaciones de Entrada y Salida en Java – Parte I, 2004. [2007]. Disponible en:  
<http://www.cricava.com/java>
- . java.io: operaciones de Entrada y Salida en Java – Parte II, 2004. [2007]. Disponible en:  
<http://www.cricava.com/java>
- ALLAMARAJU, S.; C. BEUST, et al. Professional Java Server Programming J2EE 1.3 Edition. United States, 2001. p. 1-861005-37-7
- BARRETT, B. The Webalizer 2006
- BAUER, C. y G. KING. Java Persistence with Hibernate. New York, Manning Publications Co., 2007.
- COOPER, J. Logging, Reporting, and Monitoring Facilities, 2005. [2007]. Disponible en:  
<http://www.swelltech.com/support/procyonguide/ch17.html>
- ENAMORADO, I. A. G.; I. S. R. MORALES, et al. Sistema Adaptativo de Filtrado de Contenidos, 2006.  
[Disponible en:
- FPEOPLE. AWStats ofrece estadísticas gratuitas y completas para sitios Web, 2007.
- JACOBSON, I.; G. BOOCH, et al. El Proceso Unificado de Desarrollo de Software. Felix Varela. La Habana, 2004. p. The Addison-Wesley Object Technology Series.
- MARIÑÁN, M. P. Cachés, concurrencia e Hibernate, 2007. [2007]. Disponible en:
- MUSCIANO, C. y B. KEMEDY. HTML La Guía Completa. Mexico, D.F., McGRAW-HILL INTERAMERICANA EDITORES, S.A. de C.V., 1999. 2007.
- ORALLO(, E. H. El Lenguaje Unificado de Modelado (UML), 2006. [2007]. Disponible en:  
<http://www.disca.upv.es/enheror/pdf/ActaUML.PDF>
- SEGURMATICA. Analizador de acceso a Internet, SegurmatICA, 2007. [2007]. Disponible en:  
<http://www.segurmatICA.co.cu/laboratorio/lab4.jsp>
- WALLS, C. y R. BREIDENBACH. Spring In Action. Greenwich, 2005. p. 1-932394-35-4

## ANEXOS

## Anexo 1: Formato de los ficheros logs del servidor Proxy Squid.

```

1164939270.379 1411 10.32.14.31 TCP_MISS/302 730 GET http://redir.metaservices.microsoft.com/redir/allservices/? l
1164939271.549 1279 10.32.14.31 TCP_MISS/200 1308 GET http://tools.google.com/googletalk/google-talk-versioncheck.
1164939272.289 2088 10.32.14.31 TCP_REFRESH_HIT/200 310 GET http://filetransferenabled.mail.google.com/images/clea
1164939273.602 2187 10.32.14.31 TCP_MISS/200 5089 GET http://onlinestores.metaservices.microsoft.com/serviceswitch
1164939273.925 29 10.32.14.31 TCP_IMS_HIT/200 1218 GET http://wmp.emusic.com/images/frameworks/wmp10/logo-15x15.
1164939273.963 30 10.32.14.31 TCP_IMS_HIT/304 212 GET http://images.metaservices.microsoft.com/svcs/switch/wm_com_
1164939274.028 24 10.32.14.31 TCP_IMS_HIT/304 212 GET http://www.passalong.com/onlinestore/images/menuimage.gif
1164939274.689 645 10.32.14.31 TCP_MISS/302 606 GET http://swapstart.walmart.com/i/swap/v10/wmp10_musicd_icon_16x
1164939274.871 797 10.32.14.31 TCP_IMS_HIT/304 212 GET http://xmro.xradio.com/images/wmp/sm_menu_logo.png lmarti
1164939275.124 55 10.32.14.31 TCP_IMS_HIT/304 251 GET http://downloads.walmart.com/i/swap/v10/wmp10_musicd_icon_
1164939275.124 1402 10.32.14.31 TCP_MISS/302 702 GET http://go.microsoft.com/fwlink/? lmartin DIRECT/207.46.250.10
1164939278.172 2894 10.32.14.31 TCP_MISS/200 1885 GET http://store.urge.com/sitewide/wmp/mtvn-service.jhtml? lmart
1164939285.106 1280 10.32.14.227 TCP_MISS/302 458 GET http://www.google.com/ lmartin DIRECT/72.14.203.99 text/html
1164939288.105 733 10.32.14.227 TCP_MISS/200 1711 GET http://www.google.com/cu/ lmartin DIRECT/72.14.203.99 text/
1164939289.134 655 10.32.14.227 TCP_REFRESH_HIT/200 8341 GET http://www.google.com/cu/intl/en_com/images/logo_pla
1164939291.658 2469 10.32.14.227 TCP_MISS/200 4697 GET http://www.google.com/ac.js lmartin DIRECT/72.14.203.99 tex
1164939293.479 1435 10.32.14.227 TCP_MISS/200 510 GET http://www.google.com/cu/complete/search? lmartin DIRECT/72.
1164939294.181 618 10.32.14.227 TCP_REFRESH_HIT/200 1678 GET http://www.google.com/cu/favicon.ico lmartin DIRECT/
1164939310.434 3517 10.32.14.31 TCP_MISS/200 4287 CONNECT v5.windowsupdate.microsoft.com:443 lmartin DIRECT/64.4.2
1164939315.293 4535 10.32.14.227 TCP_MISS/200 25970 GET http://www.kaspersky.com/ lmartin DIRECT/81.176.69.74 text
1164939317.527 2036 10.32.14.227 TCP_HIT/200 11643 GET http://www.kaspersky.com/style.css lmartin NONE/- text/css
1164939319.378 898 10.32.14.227 TCP_HIT/200 3071 GET http://images.kaspersky.com/en/draft/fl_logo.gif lmartin NON
1164939320.349 194 10.32.14.227 TCP_HIT/200 355 GET http://images.kaspersky.com/en/draft/fl_bul_r.gif lmartin NON
1164939321.224 736 10.32.14.227 TCP_HIT/200 355 GET http://images.kaspersky.com/en/draft/fl_search.gif lmartin NO
1164939321.753 158 10.32.14.227 TCP_HIT/200 1206 GET http://images.kaspersky.com/en/promo/v6_read.gif lmartin NON

```

## Anexo 2: Interfaz de autenticación del sistema.

**SRNI** Sistema de Reportes de la Navegación por Internet



**Autenticación**

Usuario:

Contraseña:

Copyright © 2007 Seg Inf

**Anexo 3:** Interfaz para seleccionar rango de días de los reportes.

**Anexo 4:** Interfaz que muestra reportes de consumo por IP y consumo por día.

Consumo x IP	
2 items found, displaying all items.1	
IP	Consumo(Kb)
10.3.3.49	5819831
10.3.3.240	3548697

Consumo x Dia	
2 items found, displaying all items.1	
Dia	Consumo(Kb)
15/06/2007	6068082
14/06/2007	3300446

**Anexo 5:** Interfaz que muestra reportes de consumo por Urls.

The screenshot displays the SRNI (Sistema de Reportes de la Navegación por Internet) web interface. The header includes the SRNI logo, the system name, and the user 'luis orlando martin alvarez' with a 'Salir' link. A left-hand menu is visible with options: Administrar, Supervisar, Rango Dias, Reportes, and Consumo. The main content area shows a table titled 'Consumo x Directorios' with 116 items found, displaying the first 10. The table has two columns: 'Directorios' and 'Consumo(Bytes)'. The data rows are as follows:

Directorios	Consumo(Bytes)
www.dooffy.com	2269517
cocina.cuba.cu	1371529
i.tuning.cz	776637
wallpapers.tuning.cz	599473
zulia.colciencias.gov.co:8098	303867
foobar.cl	266560
es.wikipedia.org	251439
www.facusanfrancisco.com.ar	224565
www.ilustrados.com	222192
www.google.com.cu	169825

## GLOSARIO

---

**Ancho de banda:** Es como el ancho de la tubería por la que pasa la información: a mayor ancho de banda en nuestra línea de conexión, mayor rapidez de transmisión.

**Analizadores de Log:** Proporcionan información estadística sobre las visitas a un sitio Web a partir de los ficheros "log" que registran los servidores Web.

**CGI:** Common Gateway Interface (en castellano «Interfaz Común de Pasarela»).

**Código de Ética:** Los códigos de ética tratan del comportamiento según principios éticos, su normatividad no es más que mostrar una declaración de intenciones sobre la "misión" de una institución y la limitación real con que se imponen es pequeña, aunque en algunos casos se incluyen expulsiones de la asociación en cuestión.

**Dirección IP (Internet Protocol):** Es el identificador único asociado a cualquier equipo que se encuentre dentro de una red (local o Internet).

**EJB:** Enterprise Java Beans. Es una de las API que forman parte del estándar de construcción de aplicaciones empresariales J2EE de Sun Microsystems. Su especificación detalla cómo los servidores de aplicaciones proveen objetos desde el lado del servidor que son, precisamente, los EJBs. (INFORMÁTICA 2006)

**(ICID):** Instituto Central de Investigación Digital.

**IIS:** Internet Information Server.

**JNDI:** Interfaz de Nombrado y Directorio Java, es una Interfaz de Programación de Aplicaciones para servicios de directorio.

**JSP:** Java Server Pages. Es la tecnología para generar páginas Web de forma dinámica en el servidor. Permite a los programadores generar dinámicamente páginas HTML.

**Log:** El log ofrece que información están solicitando los usuarios, que les resulta más interesante, cuando la necesita, rutas de información entre otras. Técnicamente el "log de visitas" es un archivo creado por el servidor donde se registran las acciones que los usuarios generan en la Web.

**MVC:** Modelo Vista Controlador. Es un patrón de diseño de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos posibilitando que las modificaciones a un componente de la vista ocasionen un mínimo impacto en el componente del modelo de datos.

**URL (Uniform Resource Locator):** Localizador uniforme de recurso. URL una cadena de caracteres con la cual se asigna dirección única a cada uno de los recursos de información disponibles en internet.

**ORM:** Mapeador Objeto-Relacional. Consiste en un “traductor” entre dos paradigmas, el orientado a objetos y el modelo relacional, también se conoce como motor de persistencia.

**(TIC):** Tecnologías de la Informática y las Comunicaciones.

**Tráfico Web:** En Internet, el tráfico hace referencia a la cantidad de visitantes, visitantes únicos, hits, megabytes transferidos o cualquier otra forma de medida, que se produce en un servidor Web o en un sitio Web específico en un determinado período de tiempo.