Instituto Superior Politécnico "José Antonio Echeverría"

Centro de Estudios de Ingeniería y Sistemas

Universidad de las Ciencias Informáticas

TRABAJO DE DIPLOMA PARA OPTAR POR EL TÍTULO DE

INGENIERÍA INFORMÁTICA

AUTORES: Yoenis Pantoja Zaldívar
Yurisbel Vega Ortiz

TUTORES: Ing. Madelín Haro Pérez
Lic. David Silva Barreras

Ciudad de La Habana, Junio 2006

DECLARACIÓN DE AUTORÍA

Declaramos que somos los únicos autores de este trabajo y autorizamos a la Universidad de las Ciencias Informáticas (UCI) y al Centro de Estudios de Ingeniería y Sistemas (CEIS) para que utilicen este trabajo según estimen pertinente.

Para que así conste firmamos la presente a los _____ días del mes de junio de 2006.

Yoenis Pantoja Zaldívar	Yurisbel Vega Ortiz
Tutor: Ing. Madelín Haro Pérez	Tutor: Lic. David Silva Barreras

OPINIÓN DEL USUARIO DEL TRABAJO DE DIPLOMA

El Trabajo de Diploma, titulado “Sistema Genérico de Seguridad, adaptación para el Sistema de Gestión del INSMET”, fue realizado en la Universidad de Ciencias Informáticas (UCI) en el marco del proyecto “Meteorología” de la Facultad #9 “Software Educativo, Multimedia, Meteorología y Televisión”. Se considera que, en correspondencia con los objetivos trazados, el trabajo realizado le satisface:

 Totalmente

 Parcialmente en un ____ %

Los resultados de este Trabajo de Diploma le reportan a esta entidad los beneficios siguientes:

_____

_____

_____

_____

_____

_____

Como resultado de la implantación de este trabajo se reporta un efecto económico que asciende a _____.

Y para que así conste, se firma la presente a los ____ días del mes de _____ del año _____

_____

_____

Nombre del representante de la entidad Cargo

_____

Firma Cuño

Agradecimientos

A todos los que invirtieron parte de sus días en nuestra formación profesional a lo largo de estos cinco años.

A la Ing. Madelín Haro Pérez por recibirnos en la puerta del camino final y llevarnos de la mano.

Al Ing. Febe Ángel Ciudad Ricardo, Jefe de Departamento de la Especialidad, Facultad 9, por tendernos la mano en momentos de urgencia.

Al Ing. Sasha Valdés Jiménez, por aclararnos en los días en que nos tornábamos confundidos.

Al colectivo de profesores y directivos de la Facultad de Informática y Matemática (FACINF) de la Universidad de Holguín “Oscar Lucero Moya” (UHo), nuestra casa de altos estudios durante 4 decisivos años.

A nuestros padres y familiares, por su apoyo y ejemplo en esta batalla diaria que es la vida.

A nuestros enemigos, por obligarnos a ser cada día mejores.

A nuestros amigos de siempre: Mandy, Elián, Chamakito, Yubi, Renier, César, Katia, Dayana y el resto del piquete por estar siempre al alcance de nuestras manos.

A nuestra Revolución y a Fidel.

Dedicatoria

A nuestros padres, quienes nos han brindado amor incondicional y siempre han impulsado a superarnos profesionalmente ofreciéndonos aliento constante para lograrlo.

Al resto de nuestros familiares, que nos apoyan siempre y ayudan a conseguir nuestros anhelos.

Resumen

El desarrollo acelerado de las Tecnologías de la Información y las Comunicaciones (TIC) ha impuesto al mundo empresarial, a instituciones y organizaciones este ritmo para lograr mejores resultados y buenos niveles de competitividad. Es por esto que todos optan hoy en día por incorporar sistemas automatizados que gestionen su información de manera eficiente con altos grados de disponibilidad. Sucede que en la mayoría de los casos la información que se manipula tiene un carácter sensible y se necesita protegerla o restringir su uso.

El proyecto SeguriNet consiste en el análisis, diseño e implementación de un sistema genérico de seguridad de aplicaciones Web y las acciones a tomar para su adaptación al “Sistema de Gestión de la Información” del Instituto Nacional de Meteorología, actualmente en desarrollo. Pretende garantizar de manera eficiente, íntegra y consistente la seguridad de sus datos así como la creación, configuración, autenticación y control de usuarios, reportes de accesibilidad, historial e información de las operaciones del sistema.

Con su puesta en marcha se derivan ventajas como ahorro de tiempo y código en el desarrollo de los demás módulos, aprovechando la genericidad de su implementación, siendo reutilizable para varios sistemas, independiente en cuanto a lenguaje y estructura, permitiendo la centralización de las tareas de administración lo que posibilita al sistema general convertirse en un producto robusto, fiable y extensible, logrando la integridad y consistencia de la información procesada y abaratando su costo.

Índice

Índice de figuras

Introducción

Con el amplio desarrollo de las tecnologías de la información y de las Telecomunicaciones en general, la Web se ha consolidado como un medio de comunicación e intercambio de datos masivo, cada vez más útil y próspero, y a la vez accesible a un creciente número de personas. La información fluye libremente de un ordenador a otro en décimas de segundo con un solo clic, sin importar apenas la distancia que los separa. Todos se benefician de este hecho: empresas y particulares, entidades docentes, de investigación, gubernamentales o no, asociaciones, organizaciones, etcétera; las cuales confían la gestión de su información a sistemas informáticos que aceleran y perfeccionan su manipulación, recopilación y almacenamiento.

Dependiendo del tipo de información y grado de confidencialidad, se hace necesario controlar de alguna forma su flujo, asegurando que sólo sea recibida o revisada por quien se autorice; autenticando al manipulador de la información para evitar que un impostor pueda alterar la consistencia e integridad de los datos que pueden ser privilegiados, privados o confidenciales.

El Instituto Nacional de Meteorología (INSMET) opera información bajo estas condiciones; datos que necesitan reflejar las condiciones ambientales o atmosféricas, conservarse íntegramente y con precisión pues a partir de ellos se hacen importantes estudios y pronósticos que tienen una repercusión directa en la toma de decisiones de personas, entidades y organismos de dirección de nuestro país, principalmente bajo situaciones excepcionales de carácter meteorológico.

Se solicitó a la Universidad de las Ciencias Informáticas (UCI) la creación y desarrollo de un sistema que garantice la automatización e informatización de todos sus procesos. Como parte de ese sistema, denominado como producto: “Sistema de Gestión de la Información del INSMET”, la universidad identificó la necesidad de crear módulos complementarios que garantizaran la completa funcionalidad y seguridad del sistema general. Todo esto acompañado de las características de la manipulación de los datos del Instituto, el cual no cuenta con:

- un mecanismo o proceso de control de acceso a la información que garantice que los diferentes usuarios manipulen la información que les es accesible por regla,

- una forma de auditar las violaciones en el acceso a la información y lo que es peor, las infracciones en la manipulación de esta,
- la manera de analizar comportamientos estadísticos sobre los recursos que más se solicitan o los usuarios habituales,
- la posibilidad de realizar la administración centralizada de los perfiles de usuarios para el posible conjunto de aplicaciones con las que contará la entidad en un futuro,
- la forma de evitar que en el futuro cada sistema tenga que implementar sus propios elementos de seguridad, revirtiéndose esto en un mayor costo y tiempo de desarrollo, además de no permitir concentrar todos los esfuerzos de seguridad en un solo punto;

crea la necesidad de un nuevo subsistema o módulo que realice estas funciones.

Dicho módulo debe además posibilitar el control general de los usuarios y las claves de los usuarios para facilitarles el acceso a los distintos subsistemas con los que debe contar el sistema general.

El proyecto SEGURINET surge entonces como necesidad de dar solución a las situaciones antes expuestas; identificando como problema:

¿Cómo flexibilizar la seguridad de las aplicaciones Web que conformarán el Sistema de Gestión de la Información del INSMET para conformar un nuevo sistema adaptable a las condiciones del Instituto?

Actualmente en el Instituto no se cuenta con una herramienta potente dedicada a la gestión de su información, solo algunas aplicaciones aisladas que se utilizan independientemente en alguna que otra área y que no cuentan con seguridad ni administran y controlan usuarios, elementos mínimos fundamentales a tener en cuenta en los sistemas informáticos, por lo que en la entidad no existen antecedentes de un trabajo como el que se propone.

Se ha identificado como **objeto de estudio** para nuestra labor *la seguridad en aplicaciones Web* y como campo de acción *las técnicas de seguridad de aplicaciones*

Web, las principales fallas en cuanto a seguridad de las aplicaciones Web y, por supuesto, *el Sistema de Gestión de la Información del INSMET*.

Como idea a defender consideramos:

Con la obtención de un sistema genérico de seguridad de aplicaciones Web se lograría implementar la protección de la información en los sistemas de gestión del INSMET.

El **objetivo general** de nuestro trabajo será:

- Desarrollar un módulo flexible de seguridad para aplicaciones Web.
- Adaptar el módulo a las condiciones del Sistema de Gestión de la Información del INSMET.

En correspondencia con esta propuesta desglosamos a continuación los **objetivos específicos**:

- Implementar un mecanismo de autenticación de usuarios.
- Desarrollar un proceso de control de acceso a los recursos disponibles.
- Obtener información del acceso no autorizado al sistema.
- Analizar la información relacionada a los usuarios y los recursos en cuanto a la accesibilidad realizada.
- Lograr la flexibilidad que permita un proceso de administración centralizada de grupos de clientes del sistema.

Para darle cumplimiento a estas metas propuestas se desarrollarán un grupo de actividades:

- Búsqueda de información sobre sistemas de seguridad para aplicaciones Web.
- Selección de las herramientas a utilizar para la implementación del módulo.
- Presentación del perfil de proyecto para su aprobación.
- Entrevistas a especialistas que trabajan la seguridad en la UCI.
- Realizar el levantamiento de requerimientos del sistema.
- Elaborar el análisis y diseño de la aplicación.

- Desarrollar el sistema genérico
- Realizar pruebas de adaptabilidad a las condiciones del INSMET.

El cumplimiento de los objetivos y la puesta en marcha del sistema resultante beneficiarán en gran parte al proceso de informatización del Instituto de Meteorología, funcionando de manera segura y permitiendo auditar todas las actividades de los usuarios en los sistemas, hacer análisis estadísticos sobre los accesos y limitar la disponibilidad de un recurso según se necesite. Además de garantizar la seguridad de información sensible, se disminuye el costo, esfuerzo y tiempo de desarrollo en futuros proyectos, tanto para la entidad como para el productor, la UCI, cumpliendo con el principio de la administración centralizada de usuarios, permitiendo concentrar todos los esfuerzos de seguridad en un solo lugar.

El presente trabajo se estructura en cuatro capítulos:

El primer capítulo describe el objeto de estudio, se expone una valoración del estado del arte y se analizan las tendencias y tecnologías actuales referentes a las diferentes áreas del conocimiento del tema de la seguridad de las aplicaciones Web. Además se fundamenta el uso de las herramientas, lenguajes y metodologías a utilizar.

El segundo capítulo refleja la descripción de los principales procesos involucrados en el objeto de estudio. Se definen los conceptos en un Modelo del Dominio para capturar correctamente los requisitos y poder construir un sistema consistente, se enumeran los requisitos funcionales y no funcionales que debe tener el sistema que se propone. Además incluye las descripciones de los actores y casos de uso del sistema, así como los diagramas que representan a estos últimos.

El tercer capítulo plantea todas las líneas de descripción del diseño y construcción de la solución propuesta, basadas en el futuro funcionamiento del sistema a través de la representación de diagramas de clases del diseño, de clases persistentes, del modelo de datos, de componentes y despliegue. Además se fundamenta los principios de diseño que sustentan el entorno gráfico de SEGURINET.

El cuarto y último capítulo describe la estimación de costos del sistema propuesto y sus beneficios, basado en las técnicas de Análisis de Puntos de Casos de Uso.

Capítulo 1

Fundamentación Teórica

1.1 Introducción

En este capítulo se hace una descripción general del objeto de estudio con el apoyo de las componentes de las diferentes áreas del conocimiento en el tema de la seguridad de las aplicaciones Web, se expone una valoración del estado del arte y se analizan las tendencias y tecnologías actuales referentes al entorno de la investigación.

Además se fundamenta el uso de las herramientas, lenguajes y metodologías a utilizar, lo que constituye el basamento teórico del proyecto SEGURINET.

1.2 Seguridad de Aplicaciones

En pos de conseguir un software seguro, se debe dejar claro qué se entiende por seguridad, para así luego poder establecer requisitos mínimos que deben satisfacer un sistema que pretenda ser considerado seguro. La seguridad de un sistema de software es un concepto multi-dimensional. Las múltiples dimensiones de la seguridad son:

- ***Autenticación***: Proceso de validación de la identidad de un usuario para permitir o denegar una solicitud. Habitualmente, consiste en aceptar el nombre de usuario y su contraseña y validar, posteriormente, estos datos en una base de datos de seguridad. Además de este caso típico, los procesos de autenticación pueden ser mucho más complejos. Una vez verificada y validada la identidad del usuario, se completa la solicitud de recurso. Las solicitudes posteriores que realice el mismo usuario, en teoría, no se someten al proceso de autenticación hasta que el usuario salga de la aplicación Web.
- ***Autorización***: En este proceso se comprueba que los usuarios con identidad válida tengan entrada solamente a aquellos recursos a los que tienen derechos de acceso. En otras palabras, la autorización es una simple revisión que se ejecuta en todas las fases del procesamiento de solicitudes en el servidor Web. De esta forma se garantiza que únicamente se acceda a los recursos permitidos.

- ***Personalización***: Este proceso permite que una aplicación asuma la identidad del invocador y que, en su nombre, realice solicitudes a otros recursos. El acceso se concede o deniega en función de la identidad que se adopte. Si esta identidad dispone de permisos para un recurso, la aplicación que la adopta también dispondrá de los mismos permisos de acceso.
- ***Control de acceso***: mecanismo que en función de la identificación ya autentificada permite acceder a datos o recursos.
- ***Auditoría***: un registro cronológico de los eventos relevantes a la seguridad de un sistema. Este registro puede luego examinarse para reconstruir un escenario en particular.
- ***Confidencialidad***: la propiedad de que cierta información no esté disponible a ciertas entidades.
- ***Integridad***: la propiedad de que la información no sea modificada en el trayecto fuente-destino.
- ***Disponibilidad***: la propiedad de que el sistema sea accesible a las entidades autorizadas.
- ***No repudio***: la propiedad que ubica la confianza respecto al desenvolvimiento de una entidad en una comunicación.[1]

1.3 Seguridad de Aplicaciones Web

La seguridad es un aspecto crítico de las aplicaciones Web. Según numerosos estudios, este tema es el que más limita el crecimiento de la actividad económica en la red debido a que los sistemas de información en línea y de realización de transacciones económicas a través de medios telemáticos se enfrentan a numerosas amenazas. Los ataques de piratas contra las tecnologías de la información son cada vez más frecuentes y sus efectos más devastadores. Por estos motivos resulta de crucial importancia poseer la capacidad de detectar y bloquear ataques en tiempo real contra los propios sistemas de información.

Términos como inyección de SQL, Cross Site Scripting, envenenamiento de cookies o falsificación de parámetros son relativamente novedosos para este nuevo conjunto de

amenazas a los servicios Web y que necesitan de una protección adicional a la normalmente ofrecida por los sistemas de protección en red.

No existe un sistema único de protección para la tecnología Web, sino que es necesario abordarlo en tres frentes: *control de accesos* (mediante cortafuegos de red), *control del tráfico* (mediante sistemas de prevención de intrusiones en red) y *control de las transacciones* (mediante cortafuegos de aplicaciones Web).

Según un estudio realizado durante los últimos cuatro años por Data Security for the Data (IMPERVA), tan solo un 10 por ciento de las aplicaciones Web pueden considerarse seguras ante cualquier tipo de ataque. Los estudios realizados han concluido que al menos un 92% de las aplicaciones Web son vulnerables a algún tipo de ataque.[2]

1.4 Ataques a la Web

Durante los últimos años los servidores Web se han convertido en una excelente fuente de diversión para piratas: cualquier empresa que se precie, desde las más pequeñas a las grandes multinacionales, tiene una página Web en las que al menos vende su imagen corporativa. Si hace unos años un pirata que quisiera atacar a una empresa (y no a todas, pues muy pocas tenían representación en la red) tenía que agenciárselas para obtener primero información de la misma y después buscar errores de configuración más o menos comunes de sus sistemas, hoy les basta con teclear el nombre de su objetivo en un navegador y añadir la coletilla “.com” detrás del mismo para contactar con al menos una de sus máquinas: su servidor Web. [3]

Los ataques a las páginas Web de una organización son casi siempre los más “vistosos” que la misma puede sufrir: en cuestión de minutos piratas de todo el mundo se enteran de cualquier problema que la página Web principal de una empresa más o menos grande pueda estar sufriendo, y si se trata de una modificación de la misma incluso existen recopilatorios de páginas “hackeadas”. Por supuesto, la noticia de la modificación salta inmediatamente a los medios, que gracias a ella pueden rellenar alguna cabecera sensacionalista sobre “los piratas de la red”, y así se consigue que la imagen de la empresa atacada caiga notablemente y la del grupo de piratas suba entre la comunidad “underground” nacional o internacional.

La mayor parte de estos ataques tiene éxito gracias a una configuración incorrecta del servidor o a errores de diseño del mismo: si se trata de grandes empresas, los servidores Web suelen ser bastante complejos y difíciles de administrar correctamente, mientras que si la empresa es pequeña, es muy posible que haya elegido un servidor Web simple en su instalación y administración pero en el cual es casi imposible garantizar una mínima seguridad. La cuestión es que cada día es más sencillo para un pirata ejecutar órdenes de forma remota en una máquina, o al menos modificar contenidos de forma no autorizada, gracias a los servidores Web que un sistema pueda albergar.[3]

1.5 Tipos de ataques

A nivel de Sistema

Es el nivel de acceso que representa más riesgo. Es lo que todo intruso desea lograr en una máquina: tener el control total de los recursos y adueñarse completamente de la máquina con los mismos privilegios que el propio administrador.

El acceso a nivel de sistema implica el acceso al mismo sistema operativo, en última instancia mediante una terminal remota.

Estos accesos ilegales se logran a través de algún servicio mal configurado, o de aplicaciones vulnerables que permitan la ejecución de código arbitrario en el servidor. Por ejemplo: si tenemos abierto un servicio telnet, o un SSH vulnerable, estamos dando al atacante una terminal de acceso para que corrompa la escala de privilegios del sistema.

Otra posible entrada a un sistema es la explotación de servicios vulnerables que permitan desbordamientos de búfers que nos permitan, aún sin tener una terminal, ejecutar comandos en el sistema operativo.

A nivel de Aplicación

Los ataques a nivel de aplicación son aquellos que se realizan explotando vulnerabilidades de aplicaciones que permitan modificar los datos que la propia aplicación manipula, pero sin la posibilidad de ejecución de comandos sobre el sistema operativo.

En este tipo de ataques el intruso puede cambiar lo que desee en nuestro sitio Web, o en nuestras bases de datos. Pero no se puede considerar que el servidor esté comprometido, ni que el intruso haya entrado efectivamente en el sistema.

Los ataques a nivel de aplicación son los más comunes y visibles (y los más populares entre los chicos traviesos aficionados a romper cosas). De modo que el servidor -a pesar de estos ataques- permanece intacto. La seriedad y la gravedad de estos ataques depende de la importancia de la aplicación Web atacada: no es lo mismo un ataque de este tipo en una galería de fotos online, que en una aplicación de procesamiento de pagos y transferencias financieras.[4]

A nivel de hardware

Las estaciones de trabajo modernas y las computadoras del hogar tienen BIOS que controlan los recursos del sistema a nivel del hardware. Los usuarios de las estaciones de trabajo pueden establecer contraseñas administrativas dentro del BIOS para prevenir a los usuarios maliciosos de acceder o arrancar el sistema.

Sin embargo, si el usuario malicioso roba la computadora personal (PC) (el caso más común de robo entre los viajeros frecuentes que llevan portátiles y otros dispositivos móviles) y la lleva a una ubicación donde ellos pueden desmontar la PC, la contraseña del BIOS no previene al atacante de remover el disco duro, instalarlo en otra PC sin la restricción del BIOS y montar el disco duro para leer los contenidos en él. En estos casos, se recomienda que las estaciones de trabajo tengan seguros para restringir el acceso al hardware interno. Se pueden conectar a las PC y portátiles dispositivos de seguridad especializados, tales como cables de acero asegurables, para de esta manera prevenir robos, así como también instalar seguros en el chasis mismo para prevenir el acceso interno. Este tipo de hardware está ampliamente disponible desde fabricantes como Kensington y Targus. [5]

Accesos no autorizados

El acceso no autorizado consiste en que personas que no deberían utilizar los servicios de sus sistemas acceden a ellos de forma ilícita.

Estos accesos pueden producirse de diferentes formas:

- Explotando una -vulnerabilidad conocida de un programa.

- Suplantando identidades.
- Capturando los datos de acceso (nombres de usuario y contraseña), entre otras.

Contraseñas inseguras

Las contraseñas son las llaves de un sistema. Gracias a ellas se puede evitar el acceso no autorizado de otros usuarios a nuestros sistemas.

Cada día son más las aplicaciones y servicios que requieren de un nombre de usuario y contraseña para autenticarse. La tendencia general es elegir una misma contraseña para todos los lugares y que esta contraseña sea corta para recordarla más fácilmente. Pero la elección de una contraseña frágil puede poner en peligro su sistema.

El software para averiguar contraseñas utiliza cada vez métodos más perfeccionados:

- Suposiciones inteligentes
- Ataques de diccionario
- Herramientas automatizadas que intentan todas las combinaciones posibles de caracteres, etcétera.

Estos métodos pueden permitir a un atacante averiguar la contraseña y tomar el control del sistema afectado.

Ingeniería social

Los métodos tradicionales de engaños también triunfan en Internet. Existen estafadores, embaucadores que utilizan la red para engañar a los usuarios. La Ingeniería Social se convierte en la mejor herramienta para llevar a cabo toda clase de estafas, fraudes, timos o engaños sobre los usuarios más confiados.

Así, los virus informáticos emplean estas técnicas para lograr que el usuario ejecute un archivo determinado al prometerle que este contiene, por ejemplo, un vídeo de su cantante favorito.

La clave para evitar que los atacantes puedan servirse de la ingeniería social para acceder a los datos es contar con un buen sistema de formación:

- Formación y sensibilización de usuarios
- Formación técnica y de administración

Pérdida de datos

La información se ha convertido en un activo muy valioso para las empresas. La pérdida de datos sensibles amenaza la continuidad de su negocio. Este problema puede tener diferentes orígenes:

- Fallo físico (rotura de disco duro, incendio)
- Fallo lógico (corrupción de los datos)
- Error humano (borrado accidental)
- Virus y ataques externos

Phishing

Los ciberdelicuentes utilizan esta técnica para engañar a los usuarios y conseguir datos sobre sus cuentas bancarias o tarjetas de crédito.

El phishing es un intento de estafa que busca obtener información confidencial sobre un individuo, grupo u organización mediante el uso de correos electrónicos falseados o sitios Web fraudulentos. Estos mecanismos intentan incitar a una posible víctima a proporcionar información sensible como números de tarjetas de crédito, nombres de usuario y contraseña de bancos electrónicos u otros servicios.

La técnica habitual para realizar este tipo de ataques consiste en el uso de correos electrónicos falseados (mediante la alteración de elementos identificativos como pueden ser las cabeceras de correo electrónico) que aparentan llegar desde un dominio real vinculado a una organización, habitualmente financiera, en la que los usuarios confían. Con estos correos, donde se afirma que por motivos de seguridad el usuario debe actualizar sus datos, se intenta redirigir a los usuarios de Internet a un sitio Web fraudulento que simula la Web original de la organización financiera (mediante técnicas como falsear la dirección en la barra de direcciones o imitar la apariencia del sitio Web).

Spam

Spam es todo tipo de comunicación no solicitada realizada por vía electrónica. El spam ha sido considerado por varias entidades como uno de los principales problemas al que tiene que hacer frente Internet hoy día.

El envío de correo por la Red cuesta dinero al usuario que lo recibe, tanto en la conexión como en el uso de la red (tiempo/productividad).

Troyanos

El troyano es un programa que aparentemente realiza una tarea inofensiva y sin que el usuario se dé cuenta está realizando otra completamente diferente y dañina.

El troyano crea una puerta trasera en la máquina de la víctima que le proporcionará la posibilidad de acceder a ese sistema sin autorización.

Los atacantes utilizan este tipo de programas por ejemplo para disponer de máquinas para lanzar denegaciones de servicio.

Un buen antivirus y una buena formación pueden ser elementos clave para combatir este tipo de amenaza.

Virus

Los virus son programas que se pueden introducir en nuestros sistemas de formas muy diversas: unidades de disco extraíble, Internet (e-mail, sitios Web), redes locales, etcétera. El peligro de este tipo de programas es que puede producir efectos nocivos en la máquina infectada: borrar información, modificar datos, detener su funcionamiento, etcétera. [6]

1.6 Sistemas automatizados existentes vinculados a la seguridad de aplicaciones Web

El desarrollo Web se enfrenta a algunos retos adicionales a la programación tradicional. El entorno en el que se ejecutan las aplicaciones e Internet son entornos abiertos al acceso de todo tipo de usuarios y donde se deben extremar las medidas de seguridad.

Con el objetivo de facilitar recursos que capaciten a los desarrolladores de sistemas informáticos a afrontar dichos retos, de encontrar las principales fallas y vulnerabilidades y de solucionar e informar los problemas de este tipo en el mundo de las aplicaciones Web, un sinnúmero de productos y sistemas se han desarrollado a nivel mundial y aunque no cubren en su totalidad la diversidad de obstáculos de este género que van surgiendo, brindan grandes beneficios en cuestiones de protección de datos y sistemas.

Existen grandes compañías destinadas a la seguridad de aplicaciones, como los gigantes Open Web Application Security Project (OWASP) y su consorcio Web Application Security Center (WASC), quienes se dedican a encontrar y luchar las causas de software inseguro facilitando herramientas, documentación y códigos de forma gratuita y asegurando buena calidad y garantías. Otras como la SPI Dinamics y la TB-Security tienen gran fortaleza en el lanzamiento de productos para seguridad de aplicaciones Web con una buena aceptación y calidad en el mercado de la industria informática.

Todas estas fundaciones y entidades han desarrollado muy buenas herramientas a favor de la seguridad de aplicaciones, pero cumplen funcionalidades y servicios propios que no responden la totalidad de las necesidades de un sistema Web específico, los servicios indispensables no son gratuitos y emplean metodologías de certificación y chequeo que se sitúan en escalones inalcanzables en el mundo de la informática capitalista.

WebInspect™

WebInspect es un producto lanzado por la compañía SPI Dinamics dentro de su categoría de aplicaciones de seguridad para Webs. Asegura la protección valorada de toda información crítica y vulnerable dentro de la Web en una organización, detectando principales fallos conocidos y desconocidos en la capa de aplicación propia. También ayuda a la protección segura del Servidor Web incluyendo chequeos que garantizan la correcta validación y configuración de su funcionamiento. WebInspect complementa firewalls y sistemas de detección y descubrimiento de intrusos identificando las principales vulnerabilidades y defectos dentro de servidores de aplicaciones.

Assessment Management Platform (AMP)

AMP, producto de SPI Dinamics, es una plataforma de valoración de seguridad escalable que permite a las organizaciones realizar una valoración ilimitada de seguridad de sus aplicaciones, se desarrolla bajo procesos automatizados de alto nivel en tiempo real consolidando toda la información en la organización de la capa de aplicación.

AMP entrega una plataforma distribuida que brinda al usuario la posibilidad de chequear remotamente el comportamiento de las aplicaciones Web del organismo facilitando el hallazgo de situaciones y fallas de seguridad.[7]

Quality Assurance Inspect (QAInspect)

QAInspect, desarrollado por SPI Dinamics, y puesto en marcha en la compañía Mercury, le permite a los profesionales de control de calidad que incorporen a sus aplicaciones Web una seguridad totalmente automatizada que garantiza un proceso de chequeo en la etapa de dirección de prueba global sin la necesidad de un conocimiento de seguridad especializado y sin el riesgo de retardar los horarios de descarga de productos de protección para la aplicación. Hoy en día, los usuarios de Mercury pueden dirigir y pueden manejar la comprobación funcional de la seguridad de sus sistemas probados desde una sola plataforma.[8]

DevInspect

DevInspect es un producto de seguridad que garantiza el diseño, la creación y entrega de aplicaciones Web seguras. Encuentra y arregla problemas y fallas de seguridad durante del desarrollo y después del despliegue. DevInspect aplica el análisis de vulnerabilidad más innovador y técnicas para apuntar con precisión y las vulnerabilidades de la aplicación. Ofrece la más profunda y más integrada herramienta de seguridad intuitiva con el Visual Studio .Net en todas sus versiones, aplica técnicas de análisis híbridos con funcionamiento dinámico, protege las aplicaciones en desarrollo previniendo la entrada malévola y ataques detectados en el tiempo real, realiza actualizaciones diarias de reportes de chequeos de vulnerabilidad y ofrece información a los investigadores de la seguridad de sus aplicaciones.[9]

1.7 Objeto de estudio

1.7.1 Objetivos estratégicos del INSMET

La misión principal del INSMET es suministrar información meteorológica y climática autorizada, confiable y oportuna sobre el estado y comportamiento futuro de la atmósfera. Esta información está dirigida a velar por la seguridad de la vida humana y a reducir las pérdidas de bienes materiales ante desastres naturales de origen

meteorológico, contribuyendo directamente al bienestar de la comunidad y al desarrollo sostenible.

Para cumplir su misión, el INSMET opera el Servicio Meteorológico como Sistema Nacional y lleva a cabo un amplio plan de investigaciones para perfeccionar el propio servicio y contribuir al desarrollo de los conocimientos científicos de la meteorología.[10]

1.7.2 INSMET y la informatización

El proceso de informatización avanza conjuntamente con el desarrollo de la tecnología en el marco de trabajo del INSMET. Un número amplio de aplicaciones informáticas vinculadas a su objetivo estratégico es el encargado de brindar los principales servicios dentro de la actividad meteorológica. Dependiendo de la naturaleza, sensibilidad y el grado de confidencialidad de los datos, se hace necesario controlar de alguna forma el tratamiento de estos, asegurando que solo el personal capacitado y autorizado pueda acceder a ellos o manipularlos.

Actualmente en la institución no se lleva a cabo ningún proceso de protección a la información que manipulan las aplicaciones existentes, es decir, no se emplean técnicas de autenticación y autorización como tampoco se hace control de usuarios; no se auditan los accesos a la información ni se analizan los reportes de estas actividades.

La ausencia de los mecanismos extensibles a la seguridad de las aplicaciones en la gestión de la información en la entidad encamina todo el proceso de informatización hacia fallas importantes de integridad y protección de una información que tiene por naturaleza ser altamente sensible e importante. Como la protección de estos datos se reduce a la que brinda el sistema operativo, significa que cualquiera que logre violar esta barrera podrá manipular la información a su antojo, provocando consecuencias muy lamentables.

1.7.3 Procesos objeto de automatización

Después de haber hecho el diseño teórico de esta investigación científica hemos identificado como procesos a automatizar los siguientes:

- Proceso de autenticación.

Este proceso debe comprender una forma de autenticación que permita verificar y validar la identidad del usuario para solo así permitirle solicitar un recurso.

- Proceso de autorización.

Debe brindar una forma de chequear la autorización o privilegios de un usuario a un recurso dado para permitirle o denegarle el acceso.

- Proceso de administración de usuarios.

Este proceso se refiere a la actualización (creación, eliminación o modificación) de toda la información de los usuarios y sus permisos a los distintos recursos.

- Proceso de administración de operaciones.

Este proceso se refiere a la actualización (creación, eliminación o modificación) de toda la información de las operaciones que un usuario puede realizar en el sistema, entiéndase: lectura, escritura.

- Proceso de especificación de la estructura del sistema.

A través de este proceso se debe especificar la organización del sistema, sus módulos y formularios (páginas) de la misma forma en que quedó concebido, permitiéndose además la actualización de esta información.

- Proceso de almacenamiento y consulta de reportes de accesos.

Mediante este proceso se debe permitir registrar la actividad de cada usuario en el sistema y que ante cualquier irregularidad con respecto a la seguridad pueda examinarse para reconstruir un escenario en particular.

- Proceso de conservación de información histórica.

Este proceso debe permitir que la información eliminada no sea perdida instantáneamente y pueda ser recuperada en un intervalo de tiempo especificado.

1.8 Tendencias y tecnologías actuales

Control de accesos de usuario

Hoy en día se conocen tres únicos métodos para identificar personas, son:

- Por las características físicas: biométricos.

- Por un secreto compartido: contraseñas (Passwords).
- Por la posesión de un objeto (software o hardware): Tokens o certificados digitales.

Los sistemas más utilizados actualmente son de contraseña, con diferentes variantes se aplican a casi todos los aspectos de la seguridad de la información. Los sistemas biométricos son mucho más novedosos y se están desarrollando a gran velocidad, se espera que en pocos años se incorporen a muchos aspectos de la seguridad, aunque tienen condicionantes que retardan su desarrollo como el precio de los equipos de captación, conceptos éticos, poca costumbre de utilización, etcétera.

Los sistemas de posesión de un objeto son los más antiguos en control de accesos físicos, la llave de las puertas o los sellos de los reyes son tan antiguos como el concepto de acceso o identificación de derechos. Pero en el mundo digital se utilizan muy poco para el acceso a sistemas de información, probablemente por el gasto extra que supone un identificador de objetos. Actualmente se están desarrollando mucho los accesos por sistemas criptográficos llamados certificados digitales.

Igualmente todos los sistemas se pueden combinar para aumentar la seguridad, especialmente el uso de contraseñas normalmente acompaña a los sistemas biométricos y los de objetos. No será extraño en el futuro tener que introducir una contraseña, una tarjeta inteligente y la huella dactilar para acceder a la información.

1.9 El Proceso Unificado de Desarrollo de Software como metodología a seguir

Actualmente existen ciertas tendencias fundamentadas en la idea de construir sistemas más grandes y complejos. Se quiere un software que esté mejor adaptado a las necesidades lo que a su vez hace que el software sea más complejo, pero no solo eso, sino que además sea lo más rápido posible.

Sin embargo la mayoría de los desarrolladores siguen haciendo software con los mismos métodos de hace 20 años, sin percatarse de la importancia que tiene la presencia de un proceso bien definido y bien gestionado, siendo este elemento el que marca una diferencia esencial entre proyectos hiperproductivos y otros que fracasan.

Entonces ¿qué es un proceso de desarrollo de software?

Es el conjunto de actividades necesarias para transformar los requisitos del usuario en un sistema de software.

El Proceso Unificado de Desarrollo (RUP en sus siglas en inglés) es una propuesta de proceso para el desarrollo de software basada en la orientación a objetos, el desarrollo iterativo y la modelación visual usando Lenguaje de Modelación Unificado (UML) para describir un sistema, lo cual permite incorporar al proceso de desarrollo de software un mejor control de los requerimientos y cambios. Posibilita la distribución del trabajo en diversos frentes de forma simultánea.

A pesar de ser una metodología desarrollada directamente para el trabajo con clases y objetos brinda amplias posibilidades con el manejo eficiente del tiempo de diseño e implementación de aplicaciones Web.[11]

Hay que destacar que el RUP capacita a las organizaciones de muchas maneras, la más significativa es que proporciona la forma en la que el equipo de proyecto puede trabajar de forma conjunta con los clientes y demás implicados. Lo que favorece una mayor organización y entendimiento de lo que realmente el cliente necesita y una excelente proyección del trabajo.

1.10 Tecnologías y herramientas a utilizar

Arquitecturas en Capas

La descomposición de un problema en subproblemas de menor complejidad facilita la solución del primero, lo cual no quiere decir que haga que la solución sea eficiente, para ello debe efectuarse una descomposición óptima para el logro de los resultados esperados. Todo esto se resume en la conocida frase ‘divide y vencerás’, con la que se puede describir a las arquitecturas multi-capas. [12]

Una buena arquitectura de software debe facilitar los requerimientos de mantenimiento, reusabilidad, escalabilidad, y robustez del mismo. Al concertar la solución de un problema como una serie de capas, cada capa debe ocuparse de un subconjunto de responsabilidades fuertemente acopladas y tener poca cohesión con las demás. Los cambios internos en cualquier capa deben ocasionar la menor cantidad posible de cambios en las restantes. [11]

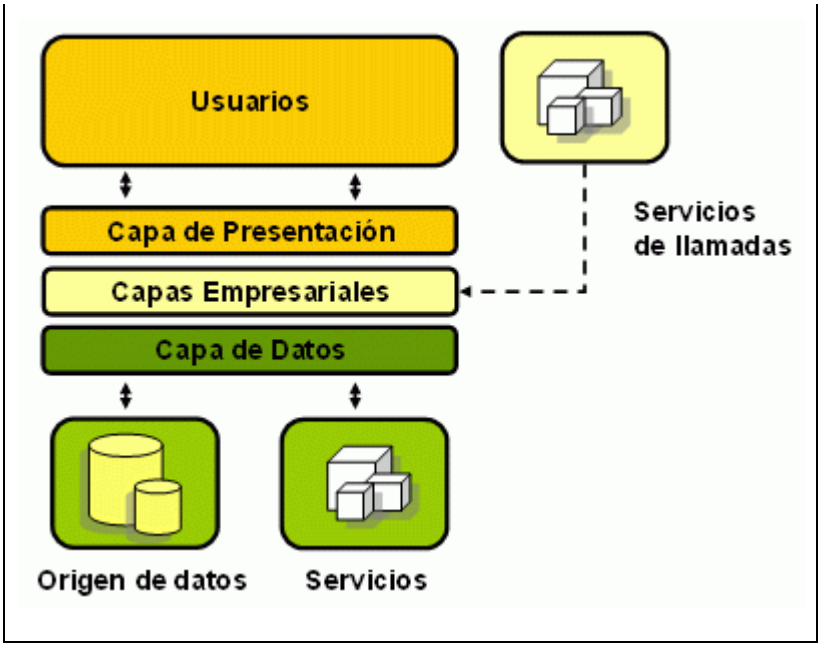


Figura 1.1 Arquitectura en capas.[13]

Una ventaja evidente de este modelo es que la capa de presentación puede desarrollarse de variadas maneras simultáneas: cliente Web, aplicación Windows, aplicación para otro Sistema Operativo, entre otras. Mientras menos responsabilidades recaigan en esta capa tanto mayor será la facilidad de desarrollar múltiples versiones de la misma. Otra ventaja sería la posibilidad de emigrar de servidor de bases de datos con un mínimo de cambios en el sistema, en tal caso los cambios se concentrarían en la capa de datos, quizás hubiera que hacer pequeños ajustes en la capa de negocio, pero nunca en la capa de presentación.

Arquitectura Orientada a Servicios

La Arquitectura Orientada a Servicios (SOA) define los servicios de los cuales estará compuesto el sistema, sus interacciones, y con qué tecnologías serán implementados. Las interfaces que utiliza cada servicio para exponer su funcionalidad son gobernadas por contratos que definen claramente el conjunto de mensajes soportados, su contenido y las políticas aplicables.

Un servicio debe ser una aplicación completamente autónoma e independiente. A pesar de esto, no es una isla, porque expone una interfaz de llamado basada en mensajes, capaz de ser accedida a través de la red. Generalmente, los servicios incluyen tanto lógica de negocio como manejo de estado (datos) relevantes a la solución del problema

para el cual fueron diseñados. La manipulación del estado es gobernada por las reglas de negocio.

La comunicación hacia y desde el servicio, es realizada utilizando mensajes y no llamadas a métodos. Estos mensajes deben contener o referenciar toda la información necesaria para entenderlo. La idea es que haya el mínimo posible de llamadas entre el cliente y el servicio.

Esta arquitectura no se funda en la idea de cualquier servicio en general, comunicado de cualquier manera, sino que más específicamente va de la mano con la expansión de los WebServices.

Otros sistemas interactúan con el WebService de una manera prescripta por su descripción utilizando mensajes SOAP, típicamente transportados usando HTTP con una serialización en XML en conjunción con otros estándares relacionados con la Web.

Vale la pena destacar la forma en la cual este estilo de arquitectura orientada a servicios redefine los modelos de ORPC (Llamadas a Procedimientos de Objetos Remotos) propios de las arquitecturas orientadas a objetos y componentes, y al hacerlo establece un modelo en el que es casi razonable pensar que cualquier entidad computacional podría llegar a conversar o a integrarse con cualquier otra.

Lo que hace diferentes a los WebServices de otros mecanismos de RPC como Java RMI, CORBA o DCOM es que utiliza estándares de la Web para los formatos de datos y los protocolos de aplicación. Esto permite que las aplicaciones ínter operen con mayor libertad, dado que las organizaciones ya seguramente cuentan con una infraestructura activa de HTTP y pueden implementar tratamiento de XML y SOAP en casi cualquier lenguaje y plataforma.[14]

La Arquitectura Orientada a Servicios es un patrón en el que los recursos están interconectados en red y se conciben como servicios accesibles por terceros a través de una interfaz estándar. En este modelo existen tres actores principales: el proveedor del servicio, el registro del servicio y el solicitante del servicio. Un componente en esta arquitectura podría considerarse como un servicio que puede ser publicado, descubierto o invocado de forma dinámica. La característica más importante de este modelo es el bajo grado de acoplamiento entre los componentes junto a una mayor flexibilidad a

cambios futuros ya que una modificación en el diseño interno puede ser factible sin necesidad de modificar el servicio. [11]

Servicios Web y XML

Un ***servicio Web*** es una colección de protocolos y estándares que sirve para intercambiar datos entre aplicaciones. Distintas aplicaciones de software desarrolladas en lenguajes de programación diferentes, y ejecutadas sobre cualquier plataforma pueden utilizar los servicios Web para intercambiar datos en redes de ordenadores como Internet. La interoperabilidad se consigue mediante la adopción de estándares abiertos.

Un servicio Web puede ser usado internamente por una aplicación o ser publicado en Internet. Estos servicios permiten la ejecución de sus funcionalidades sin importar la plataforma, sistema operativo, o lenguaje en el cual estén implementados.

Gracias a los servicios Web podemos hacer que sistemas heterogéneos trabajen conjuntamente como una sola aplicación computacional. Estos servicios constituyen una potente herramienta para el desarrollo de aplicaciones distribuidas y la plataforma .Net brinda mecanismos muy cómodos para el trabajo con ellos.[15]

El Extensible Markup Language (XML), con todas las tecnologías relacionadas, representa una manera distinta de hacer las cosas, más avanzada, cuya principal novedad consiste en permitir compartir los datos con los que se trabaja a todos los niveles, por todas las aplicaciones y soportes. Así pues, el XML juega un papel importantísimo para los servicios Web en la actualidad, es la tecnología que permitirá compartir la información de una manera segura, fiable, fácil. Además, XML permite al programador y los soportes dedicar sus esfuerzos a las tareas importantes cuando trabaja con los datos, porque algunas tareas tediosas como la validación de estos o el recorrido de las estructuras corre a cargo del lenguaje y está especificado por el estándar, de modo que el programador no tiene que preocuparse por ello.[16]

Microsoft Visual Studio.Net

Microsoft ha estado trabajando durante los últimos años en un conjunto de tecnologías, las cuales se conocen como Microsoft.NET. Estas nuevas tecnologías se han hecho con el objetivo de obtener una plataforma sencilla (Plataforma .NET) y potente para

poder distribuir el software en forma de servicios (Servicios Web) que puedan suministrarse remotamente, comunicarse y combinarse unos con otros independientemente de la plataforma, lenguaje y modelo de componentes con que se hayan desarrollado.

La mayoría de los desarrolladores cuando hacen alusión a .NET se refieren a la infraestructura del mismo. La infraestructura .NET es el conjunto de todas las tecnologías que conforman el nuevo entorno para crear (tanto aplicaciones tradicionales, aplicaciones de consola, aplicaciones de ventanas, servicios de Windows NT, Servicios Web), y ejecutar aplicaciones robustas, escalables y distribuidas. La parte de .NET que permite desarrollar estas aplicaciones es el Framework .NET.

El ***Framework .NET*** consiste en el Common Language Runtime (CLR) y en las bibliotecas de clases del Framework .NET, a veces llamadas Librerías de Clases Base. Framework .NET, también conocido como Framework SDK incluye las herramientas necesarias tanto para el desarrollo como para su distribución y ejecución. Visual Studio.NET, permite hacer todo la anterior desde una interfaz visual basada en ventanas con alto desarrollo de facilidades al programador.

El ***Common Language Runtime (CLR)*** es el núcleo de la plataforma .NET, trabaja como una máquina virtual en la que funcionan las aplicaciones, o sea, el código que se ejecuta en el CLR se ejecuta en un entorno encapsulado y gestionado, separadamente de otros procesos de la máquina. El CLR se encarga de la ejecución de las aplicaciones para ella desarrolladas y a las que ofrece numerosos servicios que simplifican su desarrollo y favorecen su fiabilidad y seguridad. Entre las características y servicios que este brinda se pueden mencionar algunos que son de gran importancia y utilidad:

- ***Modelo de programación consistente:*** A todos los servicios y facilidades ofrecidos por el CLR se accede a través de un modelo de programación orientado a objetos.
- ***Modelo de programación sencillo:*** Con el CLR desaparecen muchos elementos complejos incluidos en los sistemas operativos actuales (registro de Windows, GUIDS, HRESULTS, IUnknown, etc.).

- ***Eliminación del "infierno de las DLLs":*** En la plataforma .NET las versiones nuevas de las DLLs pueden coexistir con las viejas, de modo que las aplicaciones diseñadas para ejecutarse usando las viejas podrán seguir usándolas tras instalación de las nuevas, simplificando mucho la instalación y desinstalación de software.
- ***Ejecución multiplataforma:*** Dado que el CLR actúa como una máquina virtual, cualquier plataforma para la que exista una versión del CLR podrá ejecutar cualquier aplicación .NET, ya que el CLR está destinado a la ejecución de las aplicaciones .NET. Al código de estas aplicaciones se le suele llamar código gestionado, y al código no escrito para ser ejecutado directamente en la plataforma .NET se le suele llamar código no gestionado.
- ***Integración de lenguajes:*** Desde cualquier lenguaje para el que exista un compilador que genere código para la plataforma .NET es posible utilizar el generado para la misma usando cualquier otro lenguaje tal.
- ***Gestión de memoria:*** El CLR incluye un recolector de basura que se activa cuando se quiere crear algún objeto nuevo y se detecta que no queda memoria libre para hacerlo, entonces el recolector recorre la memoria dinámica asociada a la aplicación, detecta qué objetos hay en ella que no puedan ser accedidos por el código de la aplicación, y los elimina para limpiar la memoria de "objetos basura" y permitir la creación de otros nuevos. [17]

Las bibliotecas de clases del Framework .NET están presentes en todos los lenguajes .NET e incluyen soporte para todo, desde entrada/salida a archivos y a base de datos hasta XML y SOAP. La tecnología de .NET es un mundo informático que en vez de tener dispositivos individuales y sitios Web simplemente conectados a través de Internet tenga dispositivos, recursos y ordenadores trabajando juntos para proporcionar soluciones más sustanciosas a los usuarios.

C# .Net

El lenguaje C# fue diseñado por Microsoft especialmente para la plataforma .Net, y a pesar de ser un lenguaje joven tiene incorporado las mejores características de otros lenguajes así como nuevas potencialidades, además de que se plantea su compilador

es el más depurado y optimizado de los incluidos en el Framework .Net. Fue diseñado para lograr una combinación idónea de simplicidad, expresividad y desempeño eficiente.

C# tiene una sintaxis similar a la de C++, sin embargo incorpora un modelo de referencia a objetos parecido al de Delphi o Java, eliminando la necesidad del engorroso trabajo con punteros (aunque ofrece las herramientas para usarlos en caso de extrema necesidad). El listado de características del C# abarca mucho más de lo que se enumera a continuación:

- Los tipos básicos son tratados como clases.
- Cuenta con gestión automática de memoria.
- Implementa una fuerte política de seguridad de tipos,
- Brinda mecanismos como los índices y la instrucción foreach, que hacen más fácil e intuitivo el trabajo.
- Elimina la herencia múltiple (ofrece el uso de interfaces).
- Facilita el trabajo con propiedades y eventos.

Todo lo que se lea de C# incita hacia el uso de este lenguaje Orientado a Objetos, todas las facilidades que brinda sin duda hacen optar por su uso y explotación. [18]

Microsoft SQLServer

Para muchos, SQLServer no es considerado como el mejor Sistema Gestor de Bases de Datos Relacionales (SGBDR). Sin embargo, como puntos a su favor, puede decirse que Visual C# cuenta con un proveedor ADO.Net nativo para SQLServer , además de que Microsoft lo ha desarrollado con el objetivo de explotar al máximo las características de los sistemas operativos Windows. Entre sus características figuran:

- Soporte de transacciones.
- Gran estabilidad.
- Gran seguridad.
- Escalabilidad.
- Soporta procedimientos almacenados.

- Incluye también un potente entorno gráfico de administración, que permite el uso de comandos DDL y DML gráficamente.
- Permite trabajar en modo cliente-servidor donde la información y datos se alojan en el servidor y las terminales o clientes de la red sólo accedan a la información.
- Además permite administrar información de otros servidores de datos.

Para el desarrollo de aplicaciones más complejas (tres o más capas), Microsoft SQLServer incluye interfaces de acceso para la mayoría de las plataformas de desarrollo, cuenta con un lenguaje (Transact-SQL) para programar procedimientos almacenados y triggers; permite definir tablas, índices, vistas, etc., es distribuido y escalable, con soporte para 32 procesadores y 64 GB de RAM, es más fácil de usar que el Oracle y más potente que MySQL, brinda facilidades de replicación, seguridad administrada según perfiles configurables, que puede ser sobre cuentas de usuarios propias o integrada con las de Windows, ofrece además mecanismos de salva y restauración, y posibilidad de importar y exportar los datos en múltiples formatos. [19]

Un gestor multiplataformas puede ser muy tentador pero Microsoft SQLServer brinda las potencialidades que se requieren para desarrollar este proyecto, de una manera más sencilla e integrada con el sistema operativo que otros SGBDR.

SQLServer es un gestor de base de datos fácil de utilizar para construir, administrar e implementar aplicaciones de negocios. Esto significa tener que poner a disposición un modelo de programación rápido y sencillo para desarrolladores, eliminando la administración de base de datos para operaciones estándar, y suministrando herramientas sofisticadas para operaciones más complejas.

1.11 Conclusiones

Con la descripción general del objeto de estudio, y el apoyo de las componentes de las diferentes áreas del conocimiento del tema de la seguridad de las aplicaciones Web, se logró una valoración del estado del arte y las tendencias y tecnologías actuales referentes al entorno de investigación.

Además, se constituyó el basamento teórico del proyecto SEGURINET con el fundamento de las herramientas, lenguajes y metodologías a utilizar.

Capítulo 2

Descripción de la solución propuesta

2.1 Introducción

En este capítulo se describen los principales procesos involucrados en el objeto de estudio. Debido a la necesidad de capturar los tipos más importantes de objetos en el sistema se representa el contexto en que se emplaza definiéndose conceptos que se agrupan en un Modelo de Dominio para identificar correctamente los requisitos y poder construir un sistema consistente.

Se realiza una descripción general de los requisitos funcionales y no funcionales del sistema a desarrollar. Además incluye las descripciones de los actores y casos de uso del sistema, así como los diagramas que representan a estos últimos.

2.2 Definición de las entidades y los conceptos principales

- ***Sistema:*** Constituye el elemento primario en la utilización de los servicios de SEGURINET. Es la aplicación Web general a la cual se le presta servicios de seguridad.
- ***Aplicación:*** Es el elemento que conforma a los sistemas.
- ***Módulo:*** Identifica a un subsistema dentro de una aplicación.
- ***Submódulo:*** Es el nivel inferior de un módulo indicando que este tiene otros módulos.

Estos conceptos se muestran a continuación en la figura 2.1 graficando la jerarquía que existe entre cada elemento del sistema.

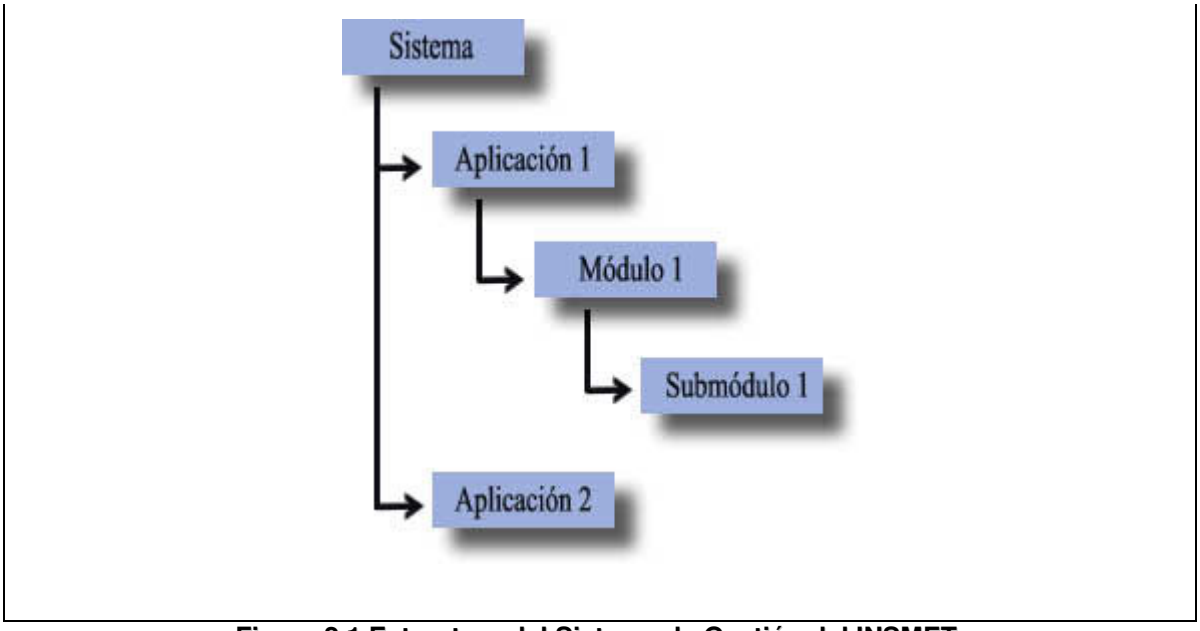


Figura 2.1 Estructura del Sistema de Gestión del INSMET.

- ***Formulario:*** Representa una página Web dentro de un módulo.
- ***Usuario:*** Es aquella persona que interactúa con una aplicación determinada fungiendo un determinado rol.
- ***Rol:*** Determina los permisos de acciones que puede desempeñar un usuario frente a una aplicación específica. Cada rol tiene permisos sobre los formularios.
- ***Acceso:*** Representa un registro sobre la entrada y la actividad de determinado usuario en un formulario.
- ***Histórico:*** Representa la información histórica archivada de los elementos eliminados dentro de SEGURINET.
- ***Operación***: Caracteriza la acción del usuario en el formulario.

2.3 Representación del modelo del dominio

Un modelo del dominio captura los tipos más importantes de objetos en el contexto del sistema. Los objetos del dominio representan las “cosas” que existen o los eventos que suceden en el entorno en que trabaja el sistema.

Las clases del dominio aparecen en tres formas típicas:

- Objetos del negocio que representan cosas que se manipulan en el negocio.
- Objetos del mundo real y conceptos de los que el sistema debe hacer un seguimiento.
- Sucesos que ocurrirán o han ocurrido.

El modelo del dominio se representa mediante diagramas de UML (especialmente mediante diagramas de clases).

Estos diagramas muestran a los clientes, usuarios, revisores y a otros desarrolladores de las clases del dominio y cómo se relacionan unas con otras mediante asociaciones.[11]

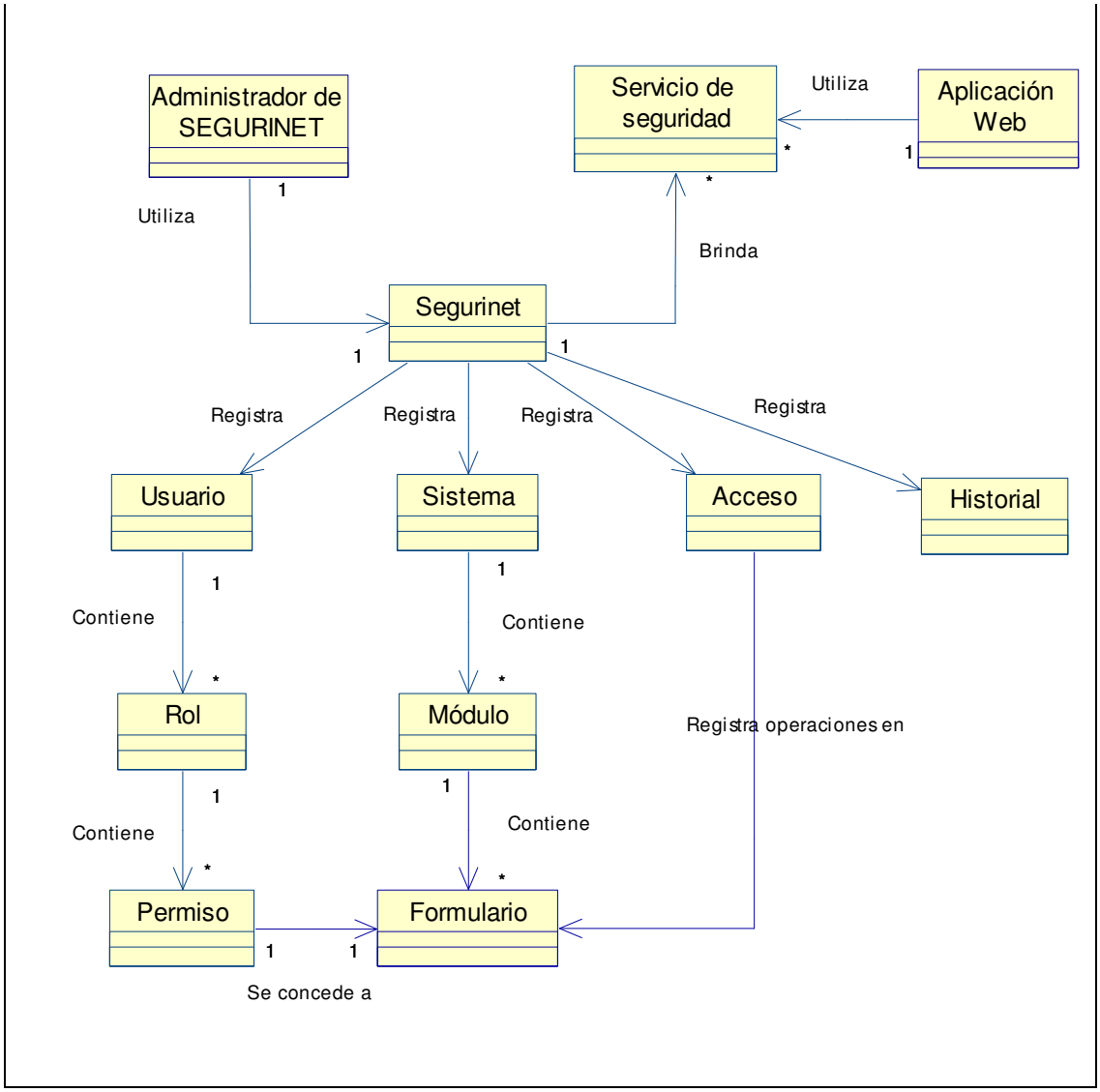


Figura 2.2 Diagrama del modelo del dominio.

2.4 Requerimientos funcionales

De acuerdo con los objetivos planteados y conociendo los conceptos que rodean al objeto de estudio, enumeramos a continuación los requerimientos funcionales. El sistema debe ser capaz de:

R 1. Autentificar usuario administrador de SEGURINET.

1.1. Contraseña del usuario para acceder a SEGURINET.

R 2. Cambiar contraseña del usuario administrador de SEGURINET

2.1. Contraseña.

R 3. Permitir que el usuario cierre la sesión de trabajo desde cualquier lugar del sistema.

R 4. Administrar usuario.

4.1. Insertar usuario siempre que no haya otro usuario con el mismo nombre (Login).

4.1.1. Nombre de usuario (Login).

4.1.2. Nombre completo del usuario.

4.1.3. Contraseña.

4.1.4. Descripción.

4.1.5. Período de cambio de la contraseña.

4.1.6. Fecha de creado.

4.1.7. Fecha de activado.

4.2. Modificar datos de un Usuario.

4.2.1. Nombre del usuario (Login)

4.2.2. Nombre completo del usuario.

4.2.3. Descripción.

4.2.4. Estado (Activo o Inactivo).

4.2.5. Período de cambio de contraseña.

4.2.6. Fecha de activado.

4.3. Eliminar un Usuario

4.3.1. Usuario.

R 5. Visualizar reporte de usuarios.

5.1. Filtrar datos de usuarios

5.1.1. Nombre de usuario (Login).

5.1.2. Nombre completo del usuario.

5.1.3. Descripción.

5.1.4. Estado (Activo o Inactivo).

5.1.5. Fecha de creado.

5.1.6. Fecha de activado.

5.1.7. Período de cambio de contraseña.

5.2. Mostrar reporte de usuarios.

R 6. Asignar rol de los identificados para una aplicación a un usuario.

6.1. Rol.

6.2. Usuario.

6.3. Aplicación

R 7. Administrar una aplicación Web y/o sus módulos.

7.1. Mostrar descripción de un módulo.

7.2. Insertar módulo.

7.2.1. Permitir insertar aplicación. (Tiene el mismo tratamiento que los módulos pero su nivel superior es el Sistema al cual SEGURINET le brinda seguridad).

7.2.1.1. Nombre de la aplicación.

7.2.1.2. Descripción.

7.2.1.3. Fecha de creada.

7.2.1.4. Fecha de activada.

7.2.1.5. Tipo de aplicación.

7.2.2. Permitir insertar módulo a una aplicación del Sistema.

7.2.2.1. Módulo del nivel superior al módulo a insertar.

7.2.2.2. Nombre del módulo.

7.2.2.3. Descripción.

7.2.2.4. Fecha de creado.

7.2.2.5. Fecha de activado.

7.2.3. Permitir insertar submódulo.

7.2.3.1. Modulo del cual el submódulo es hijo

7.2.3.2. Nombre del submódulo.

7.2.3.3. Descripción.

7.2.3.4. Fecha de creado.

7.2.3.5. Fecha de activado.

7.3. Modificar módulo

7.3.1. Modificar datos de una aplicación.

7.3.1.1. Nombre de la aplicación.

7.3.1.2. Descripción.

7.3.1.3. Estado (Activo o Inactivo).

7.3.1.4. Fecha de activado.

7.3.2. Modificar datos de un módulo.

7.3.2.1. Nombre del módulo.

7.3.2.2. Descripción.

7.3.2.3. Estado (Activo o Inactivo).

7.3.2.4. Fecha de activado.

7.3.3. Modificar datos de un submódulo.

7.3.3.1. Nombre del módulo.

7.3.3.2. Descripción.

7.3.3.3. Estado (Activo o Inactivo).

7.3.3.4. Fecha de activado.

7.4. Eliminar módulo.

7.4.1. Eliminar datos de una aplicación

7.4.1.1. Aplicación

7.4.2. Eliminar datos de un módulo.

7.4.2.1. Módulo

7.4.3. Eliminar datos de un submódulo.

7.4.3.1. Submódulo.

R 8. Visualizar reporte de módulos.

8.1. Filtrar datos de módulos:

8.1.1. Nombre de módulo.

8.1.2. Módulo padre.

8.1.3. Descripción.

8.1.4. Estado (Activo o Inactivo).

8.1.5. Fecha de creado.

8.1.6. Fecha de activado.

8.2. Mostrar reporte de módulos.

R 9. Administrar formulario.

9.1. Mostrar descripción de un Formulario.

9.2. Insertar Formulario.

9.2.1. Nombre de formulario.

9.2.2. Módulo al cual pertenece.

9.2.3. Descripción.

9.2.4. Fecha de creado.

9.2.5. Fecha de activado.

9.3. Modificar datos de un Formulario.

9.3.1. Nombre de formulario.

9.3.2. Descripción

9.3.3. Estado (Activo o Inactivo).

9.3.4. Fecha de activado

9.4. Eliminar un Formulario.

9.4.1. Formulario.

R 10. Visualizar reporte de formularios.

10.1. Filtrar datos de formularios.

10.1.1. Nombre de formulario.

10.1.2. Módulo al cual pertenece.

10.1.3. Descripción

10.1.4. Estado (Activo o Inactivo).

10.1.5. Fecha de creado.

10.1.6. Fecha de activado.

10.2. Mostrar reporte de formularios.

R 11. Administrar rol de una aplicación determinada.

11.1. Insertar rol.

11.1.1. Nombre del rol.

11.1.2. Aplicación a la que pertenece.

11.1.3. Descripción.

11.2. Asignar permisos al rol.

11.2.1. Formulario.

11.3. Modificar datos de un rol.

11.3.1. Nombre del rol.

11.3.2. Descripción.

11.3.3. Estado (Activo o Inactivo).

11.3.4. Fecha de activado.

11.3.5. Formulario

11.4. Eliminar un rol.

11.4.1. Rol.

R 12. Visualizar reporte de roles de los usuarios de las aplicaciones.

12.1. Filtrar datos de roles:

12.1.1. Nombre del rol.

12.1.2. Aplicación a la que pertenece.

12.1.3. Descripción del rol.

12.1.4. Estado (Activo o Inactivo).

12.1.5. Fecha de creado.

12.1.6. Fecha de activado.

12.2. Mostrar reporte de roles.

R 13. Visualizar reporte de accesos.

13.1. Filtrar información de accesos los cuales pueden ser:

13.1.1. Aplicación.

13.1.2. Módulo.

13.1.3. Formulario.

13.1.4. Usuario

13.1.5. Operación.

13.1.6. Consulta.

13.1.7. Fecha del acceso.

13.2. Mostrar reporte de accesos.

R 14. Administrar operaciones.

14.1. Insertar operación.

14.1.1. Nombre de la operación.

14.2. Administrar operación.

14.2.1. Operación.

14.2.2. Nombre de la operación.

14.3. Eliminar operación.

14.3.1. Operación.

R 15. Administrar históricos.

15.1. Recuperar elemento desde Históricos.

15.1.1. Recuperar módulo.

15.1.1.1. Módulo.

15.1.2. Recuperar formulario.

15.1.2.1. Formulario.

15.1.3. Recuperar usuario.

15.1.3.1. Usuario.

15.1.4. Recuperar rol.

15.1.4.1. Rol.

15.2. Eliminar elemento de Históricos (Eliminación definitiva del sistema).

15.2.1. Eliminar módulo.

15.2.1.1. Módulo.

15.2.2. Eliminar formulario.

15.2.2.1. Formulario.

15.2.3. Eliminar usuario.

15.2.3.1. Usuario.

15.2.4. Eliminar rol.

15.2.4.1. Rol.

R 16. Visualizar reporte de Históricos.

16.1. Filtrar datos históricos.

16.1.1. Filtrar datos históricos de usuarios.

16.1.1.1. Nombre de usuario (Login).

16.1.1.2. Nombre completo del usuario.

16.1.1.3. Descripción.

16.1.1.4. Estado (Activo o Inactivo).

16.1.1.5. Período de cambio de contraseña.

16.1.1.6. Fecha de creado.

16.1.1.7. Fecha de activado.

16.1.1.8. Fecha de eliminado

16.1.2. Filtrar datos históricos de módulos.

16.1.2.1. Nombre de módulo.

16.1.2.2. Módulo padre.

16.1.2.3. Descripción.

16.1.2.4. Estado (Activo o Inactivo).

16.1.2.5. Fecha de creado.

16.1.2.6. Fecha de activado

16.1.2.7. Fecha eliminado

16.1.3. Filtrar datos históricos de formularios.

16.1.3.1. Nombre de formulario.

16.1.3.2. Módulo padre.

16.1.3.3. Descripción.

16.1.3.4. Estado (Activo o Inactivo).

16.1.3.5. Fecha de creado.

16.1.3.6. Fecha de activado

16.1.3.7. Fecha eliminado

16.1.4. Filtrar datos históricos de roles.

16.1.4.1. Nombre del rol.

16.1.4.2. Aplicación a la que pertenece.

16.1.4.3. Descripción.

16.1.4.4. Estado (Activo o Inactivo).

16.1.4.5. Fecha de creado.

16.1.4.6. Fecha de activado.

16.1.4.7. Fecha de eliminado.

16.1.5. Filtrar datos históricos de accesos.

16.1.5.1. Aplicación.

16.1.5.2. Módulo.

16.1.5.3. Formulario.

16.1.5.4. Usuario

16.1.5.5. Operación.

16.1.5.6. Consulta.

16.1.5.7. Fecha del acceso

16.2. Mostrar reporte de Históricos.

R 17. Configurar Sistema.

17.1. Permitir configurar periodo de cambio de la contraseña del administrador de Segurinet.

17.1.1. Periodo de cambio de contraseña.

17.2. Configurar el período de existencia de los elementos en el historial.

17.2.1. Tiempo de existencia de los datos.

R 18. Dar servicio de controlar la autenticación de usuario a una aplicación Web.

18.1. Aplicación.

18.2. Nombre de usuario (login).

18.3. Contraseña.

18.4. Formulario.

R 19. Dar servicio de controlar vigencia de contraseña a una aplicación Web.

19.1. Nombre de usuario.

19.2. Contraseña.

R 20. Dar servicio cambiar contraseña de usuario a una aplicación Web.

20.1. Nombre de usuario.

20.2. Contraseña.

20.3. Nueva contraseña.

20.4. Confirmación de la nueva contraseña.

R 21. Dar servicio de registrar accesos de usuario.

21.1. Nombre de usuario.

21.2. Contraseña.

21.3. Operación que realiza.

21.4. Formulario al cual accede.

21.5. Consulta que realiza.

21.6. Fecha del acceso al formulario.

2.5 Requerimientos no funcionales

Apariencia o interfaz externa.

- El sistema debe tener una apariencia profesional y un diseño gráfico sencillo, sin muchas imágenes.
- Debe tener una interfaz de fácil navegación, atractiva al usuario, legible e intuitiva, con iconos y metáforas sugerentes.

- Utilización de los colores azul, blanco y gris fundamentalmente porque son los colores representativos de la entidad y de su Sistema de Gestión de la Información.
- La aplicación debe ejecutarse utilizando como resolución 1024x768 pero debe estar preparada para verse en otras resoluciones.

Usabilidad.

- El sistema debe permitir la petición concurrente de un gran número de solicitudes de servicios de otras aplicaciones.
- La información deberá estar disponible en todo momento, limitada solamente por las restricciones que esta tenga de acuerdo a la política de seguridad del sistema.
- El sistema debe permitir paginar los listados resultantes de los reportes si estos son extensos.
- El sistema debe permitir obtener reportes impresos sobre la información que manipula.
- El tiempo y fecha del sistema en el servidor deben corresponderse con la fecha y hora reales.

Rendimiento.

- El sistema debe ser lo más eficiente posible para poder lograr un tiempo de respuesta adecuado.

Soporte.

- Se debe garantizar de implantación y prueba del sistema, además de un breve adiestramiento al personal que tendrá la responsabilidad de administrarlo.

Seguridad

- Debe mantenerse la integridad de la información, es decir, la corrección y completitud de los datos.
- Debe permitirse el almacenamiento de información histórica por un período de tiempo de 60 días que puede ser configurado por el Administrador de SEGURINET.

- Se debe establecer una política de salvas periódicas de la información para permitirle al sistema recuperarse ante un fallo de gran magnitud.
- Permitir que las contraseñas de los usuarios se almacenen y viajen por la red encriptadas.
- Permitir que los Servicios Web sean utilizados solo por las personas autorizadas por el Administrador de la aplicación Web.
- Garantizar que la contraseña sea fuerte, que cumpla con las exigencias de longitud y composición.
- Garantizar que la contraseña sea cambiada cada cierto período prudente de tiempo.
- Advertir al usuario sobre acciones “irreversibles”.
- Permitir la característica de no repudio, o sea, que quien participe en una transacción no pueda negar haberlo hecho.

Confiabilidad.

- Debe mantenerse la consistencia de los datos en correspondencia con la realidad.
- Es importante que el sistema presente un mecanismo de respuesta rápida ante fallos y que en caso de ocurrencia se minimicen las pérdidas de información por lo que deberá existir un plan de salvas y mantenimiento garantizando con esto una rápida protección y recuperación ante un problema dado.
- El sistema debe mostrar mensajes de confirmación luego de ejecutada una acción que ratifiquen si la misma se llevó o no a cabo exitosamente.

Para una correcta ejecución de la aplicación se necesita:

- Sistema operativo cliente: Windows 2000 o superior.
- Sistema operativo servidor: Windows 2000 o superior.
- Navegador: Internet Explorer 4.0 o superior.
- Sistema Gestor de Base de Datos (SGBD): SQL-Server 2000
- Servidor Web: Internet Information Services 5.1 o superior.

- Opción de cliente JavaScript habilitada.

Hardware.

- Velocidad del Procesador: Intel Pentium, 2.4 GHz.
- Mínimo de Memoria RAM: se recomiendan 256 MB
- Mínimo de Disco duro: 15 MB.

2.6 Actores del sistema

Tabla 1. Definición de actores del sistema a automatizar

ACTORES DEL SISTEMA	DESCRIPCIÓN
Administrador de SEGURINET.	Es el encargado de gestionar toda la información que manipula SEGURINET.
Aplicación Web.	Son los sistemas clientes a quienes se les brinda los diferentes servicios de SEGURINET.

2.7 Paquetes y sus relaciones

Para representar los casos de usos del sistema se decidió agruparlos en cuatro paquetes: Servicios, Control, Reportes y Actualización; atendiendo a características similares entre ellos y con el objetivo de hacer comprensible la representación de los diagramas de casos de uso del sistema. Los paquetes se representan y se relacionan según la figura 2.3.

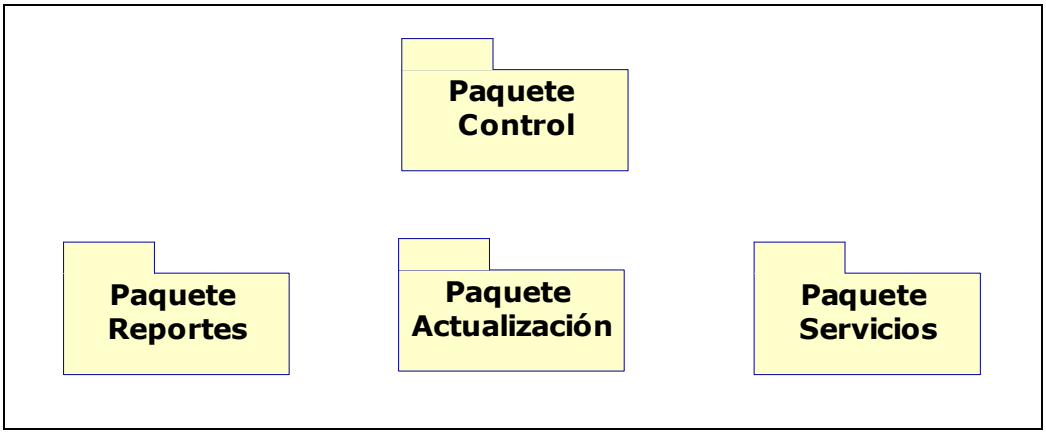


Figura 2.3 Diagrama de paquetes y sus relaciones.

2.8 Casos de uso del sistema a automatizar

Un caso de uso es un documento narrativo que describe la secuencia de un actor (agente externo) que utiliza un sistema para completar un proceso. [20]

Utilizando las facilidades que brinda el UML, se representarán los requisitos funcionales del sistema mediante un diagrama de casos de uso. Para ello hay que definir cuales serían los actores que van a interactuar con el sistema, y los casos de uso que van a representar las funcionalidades.

2.9 Diagramas de casos de uso del sistema

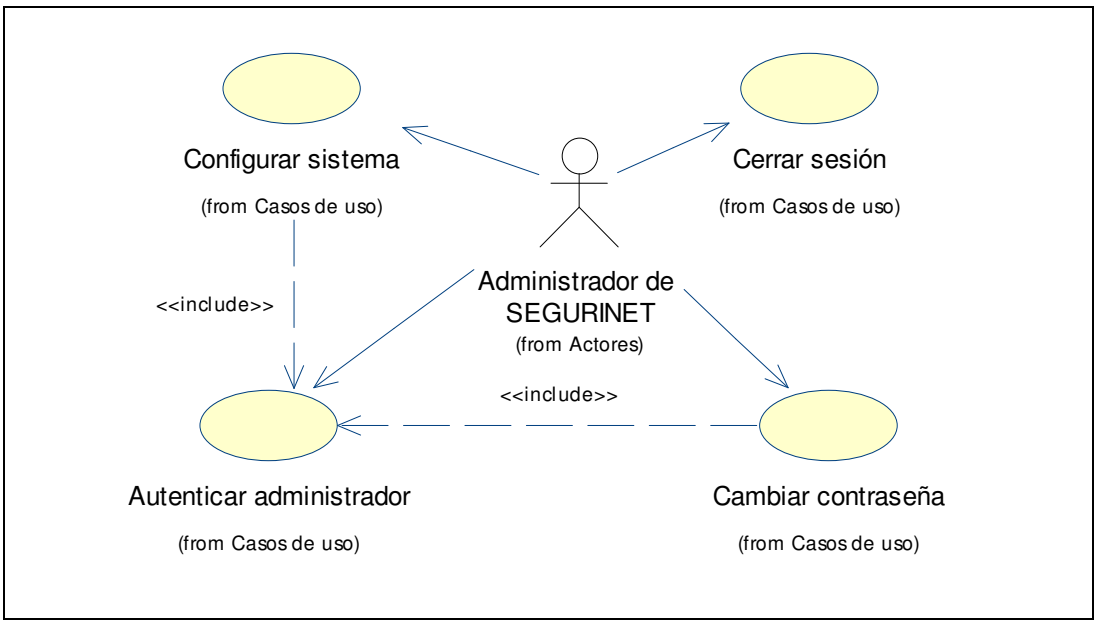


Figura 2.4 Diagrama de casos de uso del paquete Control.

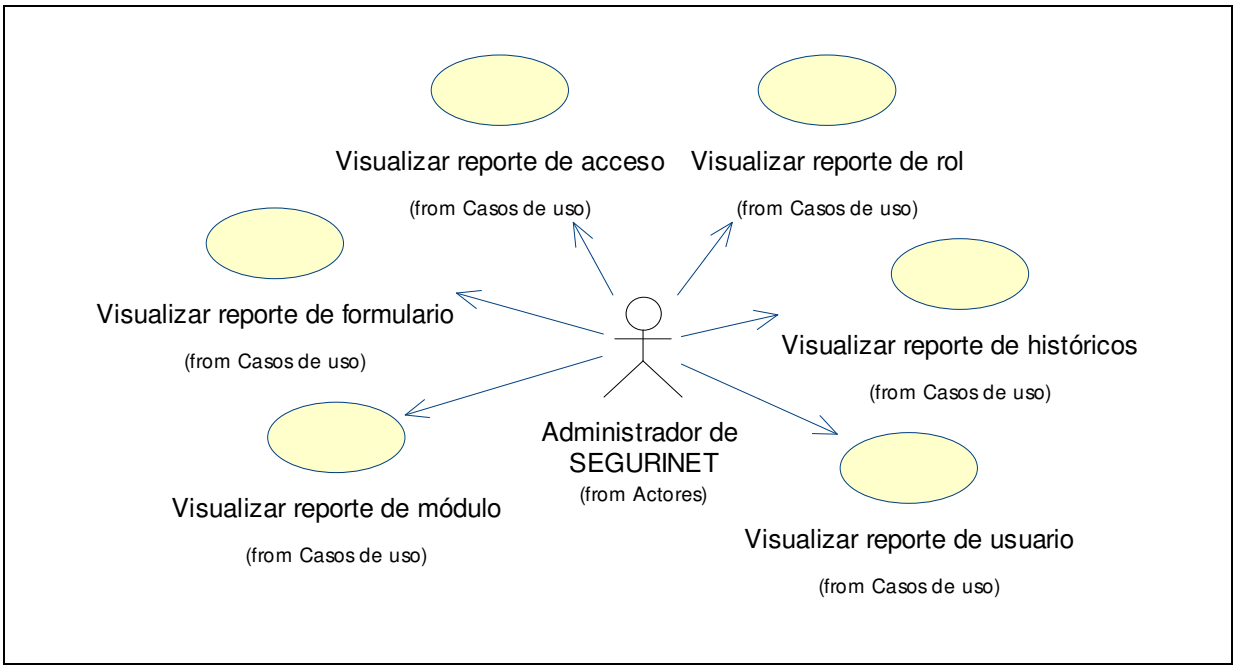


Figura 2.5 Diagrama de casos de uso del paquete Reportes.

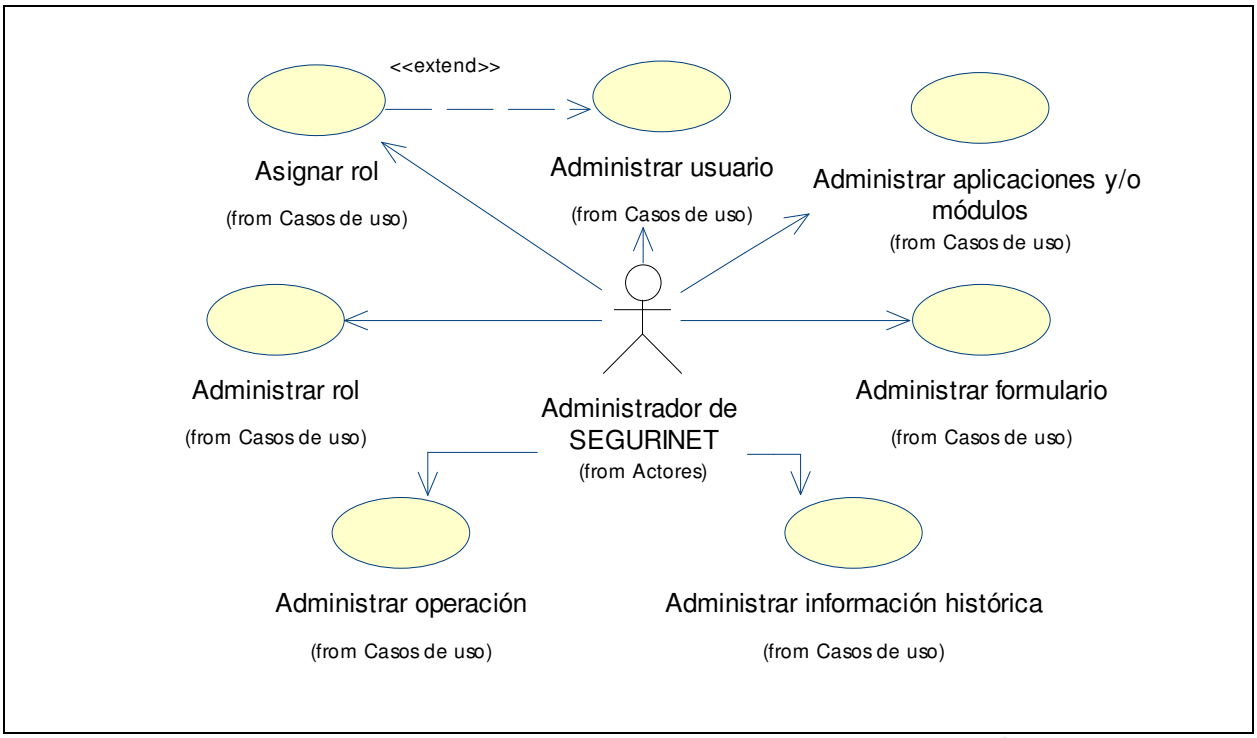


Figura 2.6 Diagrama de casos de uso del paquete Actualización.

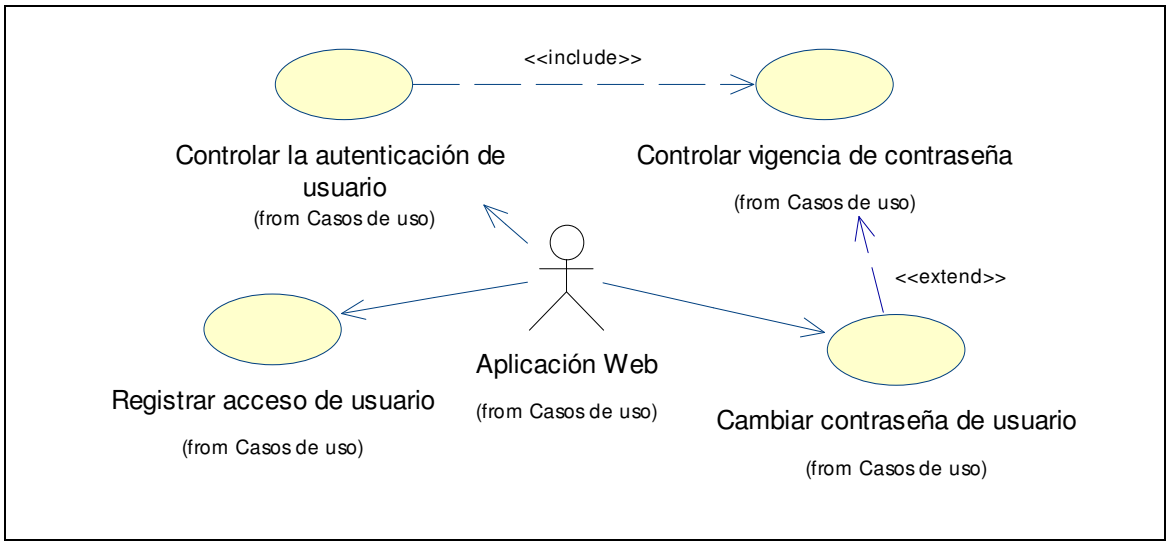


Figura 2.7 Diagrama de casos de uso del paquete Servicios.

2.10 Descripción de los casos de uso

Caso de uso:	**Autenticar administrador**
Actores:	Administrador de SEGURINET.
Propósito:	Reconocer la identidad del administrador de SEGURINET.
Resumen:	El Administrador de SEGURINET se identifica ante el sistema el cual valida sus datos.
Referencias:	R1
Precondiciones:	
Poscondiciones:	El Administrador puede trabajar con el sistema sin volver a identificarse para cada operación que realice.
Caso de uso asociado:	

Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El caso de uso comienza cuando el Administrador de SEGURINET desea acceder el sistema.	2. El sistema solicita la entrada de la contraseña.
3. El Administrador de SEGURINET especifica: contraseña.	4. El sistema valida la contraseña.
	5. El sistema permite la entrada.
Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**

	4. Si al validar la contraseña esta es incorrecta el sistema muestra un mensaje de error.
	5. El sistema no permite la entrada.
Requerimientos especiales:	

Caso de uso:	**Cambiar contraseña**
Actores:	Administrador de SEGURINET.
Propósito:	Establecer una nueva clave de acceso al sistema para el administrador de SEGURINET.
Resumen:	El Administrador de SEGURINET desea cambiar su contraseña facilitando los datos pertinentes y el sistema lo valida y actualiza la contraseña con el nuevo valor especificado si cumple con las restricciones de longitud y composición.
Referencias:	R2
Precondiciones:	El Administrador de SEGURINET se encuentra previamente autenticado en el sistema.
Poscondiciones:	El Administrador posee una nueva clave de acceso al sistema.
Caso de uso asociado:	

Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El caso de uso comienza cuando el Administrador de SEGURINET desea cambiar su	2. El sistema solicita la entrada de: contraseña actual, contraseña nueva y

contraseña.	confirmación de la nueva contraseña.
3. El Administrador de SEGURINET especifica: contraseña actual, contraseña nueva y confirmación de la nueva contraseña.	4. El sistema valida la contraseña actual.
	5. El sistema verifica que la nueva contraseña cumpla con las restricciones de longitud y composición.
	6. El sistema verifica que la nueva contraseña no sea igual a la anterior.
	7. El sistema cambia la actual contraseña por la nueva.
Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
	4. Si al validar la contraseña esta es incorrecta el sistema muestra un mensaje de error.
	5. Si la nueva contraseña no cumple con las restricciones de longitud y composición el sistema muestra un mensaje de error.
	6. Si la nueva contraseña es igual a la anterior el sistema muestra un mensaje de error.
Requerimientos especiales:	

Caso de uso:	**Administrar usuario** (Sección: Principal)
Actores:	Administrador de SEGURINET.
Propósito:	Administrar (insertar, modificar o eliminar) información referente a los usuarios de las aplicaciones a las que SEGURINET brinda seguridad.
Resumen:	El Administrador de SEGURINET actualiza información referente a los usuarios de las aplicaciones a las que SEGURINET brinda seguridad, si desea insertar un usuario nuevo especifica sus datos, si desea cambiar los datos de alguno los modifica y si un usuario deja de existir lo elimina.
Referencias:	R4
Precondiciones:	El Administrador de SEGURINET se encuentra previamente autenticado en el sistema.
Poscondiciones:	
Caso de uso asociado:	

Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El caso de uso comienza cuando el Administrador de SEGURINET selecciona una opción para actualizar información referente a los usuarios: a. Si selecciona *Nuevo*, véase sección *Insertar nuevo usuario*. b. Si selecciona *Modificar*, véase sección *Modificar usuario*. c. Si selecciona *Eliminar*, véase sección *Eliminar*	

usuario.	
Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
Requerimientos especiales:	

Caso de uso:	**Administrar usuario** **(**Sección: Insertar nuevo usuario**)**
Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
	1. El sistema solicita la entrada de: nombre de usuario, contraseña, confirmación de contraseña, período de vigencia de la contraseña, nombre completo del usuario y descripción del usuario.
2. El Administrador de SEGURINET introduce: nombre de usuario, contraseña, confirmación de contraseña, período de vigencia de la contraseña, nombre completo del usuario y descripción del usuario.	3. El sistema verifica que el nombre de usuario entrado no exista en el sistema.
	4. El sistema verifica que la contraseña y la confirmación de la misma coincidan.
	5. El sistema inserta el nuevo usuario y muestra un mensaje que confirma la exitosa ejecución de dicha acción.
Cursos alternativos	

Acción del actor:	**Respuesta del sistema:**
	3. Si se verifica que el usuario especificado existe en el sistema se muestra un mensaje de error.
	4. Si se verifica que la contraseña y la confirmación de la misma no coinciden, el sistema muestra un mensaje de error.
Requerimientos especiales:	

Caso de uso:	**Administrar usuario** **(**Sección: Modificar usuario**)**
Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El Administrador de SEGURINET selecciona el usuario que desea modificar.	2. El sistema muestra los actuales datos del usuario seleccionado que pueden ser modificados: nombre de usuario, período de vigencia de la contraseña, nombre completo del usuario y descripción del usuario.
3. El Administrador de SEGURINET modifica alguno o todos los datos siguientes: nombre de usuario, período de vigencia de la contraseña, nombre completo del usuario y descripción del usuario.	4. El sistema verifica que si se modificó el nombre de usuario no coincida con ninguno existente en el sistema.
	5. El sistema modifica la información del usuario.
Cursos alternativos	

Acción del actor:	**Respuesta del sistema:**
	2. Si el Administrador de SEGURINET no seleccionó el usuario que desea modificar se muestra un mensaje de error.
	4. Si se verifica que se modificó el nombre de usuario y que coincida con uno existente en el sistema se muestra un mensaje de error.
Requerimientos especiales:	

Caso de uso:	**Administrar usuario** **(**Sección: Eliminar usuario**)**
Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El Administrador de SEGURINET selecciona el usuario que desea eliminar.	2. El sistema muestra un mensaje de advertencia solicitando que se confirme si se desea realmente eliminar el usuario seleccionado.
3. El Administrador de SEGURINET confirma que desea eliminar el usuario seleccionado.	4. El sistema elimina la información del usuario seleccionado.
Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
	2. Si el Administrador de SEGURINET no seleccionó el usuario que desea eliminar se muestra un mensaje de error.
Requerimientos especiales:	

Caso de uso:	**Visualizar reporte de usuarios.**
Actores:	Administrador de SEGURINET.
Propósito:	Mostrar un listado de usuarios y sus respectivos datos que cumplan con las condiciones especificadas por el administrador de SEGURINET.
Resumen:	El Administrador de SEGURINET desea obtener un reporte de los usuarios que cumplan con las restricciones que él mismo establece y el sistema usando este criterio los selecciona y muestra.
Referencias:	R5
Precondiciones:	El Administrador de SEGURINET se encuentra previamente autenticado en el sistema.
Poscondiciones:	
Caso de uso asociado:	

Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El caso de uso comienza cuando el Administrador de SEGURINET desea obtener un reporte determinado de los usuarios de de las aplicaciones a las que SEGURINET brinda seguridad especificando un criterio de selección dado por un: nombre de usuario, nombre completo de usuario, descripción, estado, fecha de creado, fecha de activado y/o período de cambio de contraseña.	2. El sistema muestra el listado de usuarios correspondiente.
3. El Administrador de SEGURINET solicita la elaboración del reporte a partir del listado	4. El sistema elabora el reporte como un documento, listo para imprimirse, con un

obtenido.	encabezado que lo describe.
Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
	2. Si el Administrador de SEGURINET no especificó un criterio de selección el sistema muestra un listado completo con todos los usuarios que administra SEGURINET.
Requerimientos especiales:	

Caso de uso:	**Asignar rol.**
Actores:	Administrador de SEGURINET.
Propósito:	Asignar un rol específico perteneciente a una aplicación de las que SEGURINET brinda seguridad a un usuario del sistema que cumplirá esta función.
Resumen:	El Administrador de SEGURINET quiere asignarle un nuevo rol a un usuario determinado, selecciona dicho usuario y la aplicación en la que ejercerá dicho papel, el sistema facilita entonces el listado de todos los roles que se han especificado para la aplicación en cuestión, de donde el Administrador seleccionará el indicado para asignarle finalmente a ese usuario.
Referencias:	R6
Precondiciones:	El Administrador de SEGURINET se encuentra previamente autenticado en el sistema.
Poscondiciones:	
Caso de uso asociado:	

Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El caso de uso comienza cuando el Administrador de SEGURINET desea asignarle un rol a un usuario del sistema al seleccionar un usuario y la opción asignar rol.	2. El sistema solicita la selección de la aplicación a la que pertenece el rol que se desea asignar.
3. El Administrador de SEGURINET selecciona la aplicación indicada.	4. El sistema muestra el listado de todos los roles de la aplicación que pueden ser asignados a dicho usuario.
5. El Administrador de SEGURINET selecciona uno o más roles que desea asignar al usuario.	6. El sistema asigna el rol seleccionado al usuario y muestra un mensaje que confirma el desarrollo exitoso de la operación especificada.
Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
	2. Si el Administrador de SEGURINET no seleccionó un usuario al que desea asignarle un nuevo rol el sistema muestra un mensaje de error.
Requerimientos especiales:	

Caso de uso:	**Servicio cambiar contraseña de usuario**
Actores:	Aplicación Web.
Propósito:	Establecer una nueva clave de acceso para un usuario a una aplicación de las que SEGURINET les brinda seguridad.

Resumen:	Una aplicación Web de las que SEGURINET les brinda seguridad solicita cambiar la contraseña de uno de sus usuarios facilitando los datos pertinentes, el sistema los valida y actualiza la contraseña con el nuevo valor especificado si cumple con las restricciones de longitud y composición.
Referencias:	R20
Precondiciones:	
Poscondiciones:	El usuario de la aplicación Web que solicita el cambio de contraseña posee una nueva clave de acceso al sistema.
Caso de uso asociado:	

Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El caso de uso comienza cuando una aplicación Web solicita a SEGURINET cambiar la contraseña de uno de sus usuarios facilitando: nombre de usuario, contraseña, nueva contraseña y su confirmación.	2. El sistema valida el nombre de usuario.
	3. El sistema valida la contraseña actual.
	4. El sistema verifica que la nueva contraseña cumpla con las restricciones de longitud y composición.
	5. El sistema verifica que la nueva contraseña no sea igual a la anterior.
	6. El sistema cambia la actual contraseña por la nueva informándole el cambio exitoso a la aplicación Web en cuestión.

Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
	2. Si al validar el nombre de usuario este no existe el sistema informa a la aplicación Web un error.
	3. Si al validar la contraseña esta es incorrecta el sistema informa a la aplicación Web un error.
	4. Si la nueva contraseña no cumple con las restricciones de longitud y composición el sistema informa a la aplicación Web un error.
	5. Si la nueva contraseña es igual a la anterior el sistema informa a la aplicación Web un error.
Requerimientos especiales:	

Caso de uso:	**Servicio registrar acceso de usuario**
Actores:	Aplicación Web.
Propósito:	Registrar el acceso de un usuario a una aplicación de las que SEGURINET les brinda seguridad.
Resumen:	Una aplicación Web de las que SEGURINET les brinda seguridad solicita registrar el acceso de uno de sus usuarios a un formulario perteneciente a ella facilitando los datos pertinentes, el sistema los valida y ejecuta el registro pertinente.
Referencias:	R21
Precondiciones:	

Poscondiciones:	Ha quedado registrado el acceso del usuario de la aplicación Web al formulario dado en SEGURINET.
Caso de uso asociado:	

Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El caso de uso comienza cuando una aplicación Web solicita a SEGURINET registrar el acceso de uno de sus usuarios facilitando: nombre de usuario, contraseña, aplicación, formulario, operación, consulta y fecha de acceso.	2. El sistema valida el nombre de usuario.
	3. El sistema valida la contraseña.
	4. El sistema verifica que la aplicación suministrada este registrada en SEGURINET.
	5. El sistema verifica que el formulario suministrado exista.
	6. El sistema verifica que el usuario tenga acceso al formulario especificado.
	7. El sistema verifica que la operación suministrada sea valida, exista en SEGURINET.
	8. El sistema registra el acceso.
Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
	2. Si al validar el nombre de usuario este no existe el sistema informa a la aplicación Web un

	error.
	3. Si al validar la contraseña esta es incorrecta el sistema informa a la aplicación Web un error.
	4. Si aplicación no esta registrada en SEGURINET el sistema informa a la aplicación Web un error.
	5. Si el formulario no existe el sistema informa a la aplicación Web un error.
	6. Si el usuario no tiene acceso al formulario dado el sistema informa a la aplicación Web un error.
	7. Si la operación no es válida, no existe en SEGURINET, el sistema informa a la aplicación Web un error.
Requerimientos especiales:	

2.11 Conclusiones

En este capítulo se modeló el proceso de funcionamiento del sistema a través del Modelo del Dominio, se detallaron los principales conceptos y se describieron los elementos básicos del funcionamiento de SEGURINET. Se logró una descripción general de los requerimientos funcionales y no funcionales del sistema, así como de los actores que intervienen. Fueron explicados, a través de los diagramas correspondientes, los casos de uso a automatizar; distribuidos en forma de paquetes para un mejor entendimiento y una mejor representación.

Capítulo 3

Diseño y construcción de la solución propuesta

3.1 Introducción

En este capítulo se plantea todas las líneas de descripción del diseño y construcción de la solución propuesta, basadas en el futuro funcionamiento del sistema a través de la representación de diagramas de clases del diseño, de clases persistentes, del modelo de datos, de componentes y despliegue. Además se fundamenta los principios de diseño que sustentan el entorno gráfico de SEGURINET.

3.2 Modelo de diseño

3.2.1 Diagrama de clases del diseño

Para una mejor representación gráfica de las clases del diseño, se agruparon las mismas en subsistemas de acuerdo a su distribución y funcionalidad.

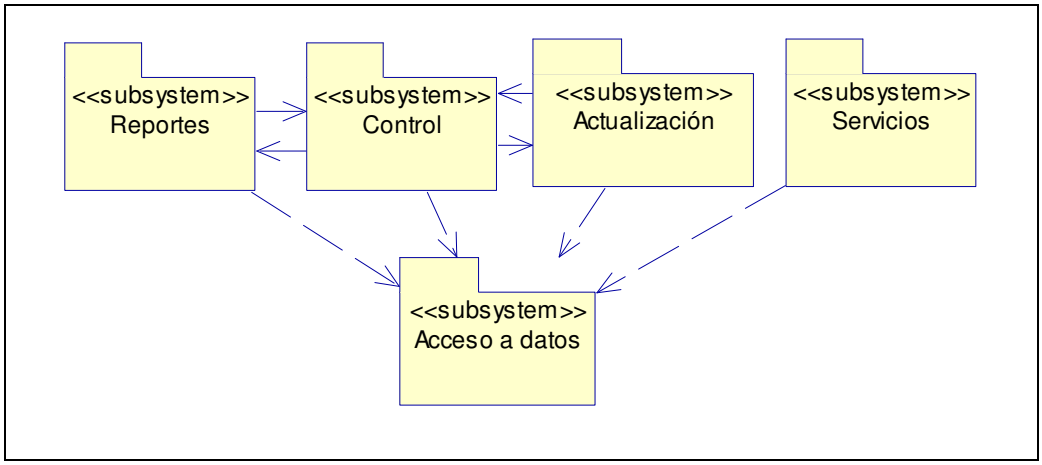


Figura 3.1 Subsistemas del Modelo del Diseño.

3.2.1.1 Subsistema de Acceso a datos

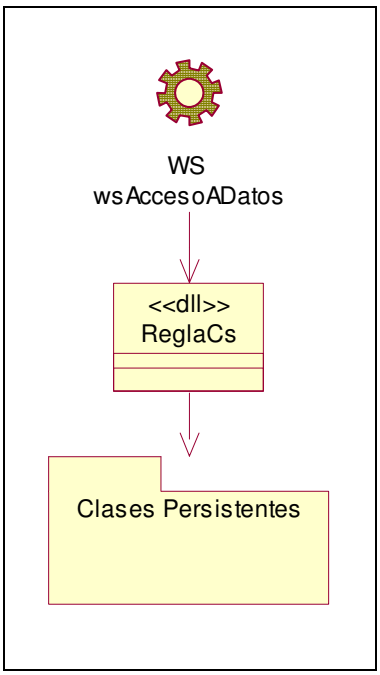


Figura 3.2 Diagrama de clases del paquete Acceso a datos.

3.2.1.2 Subsistema de Control

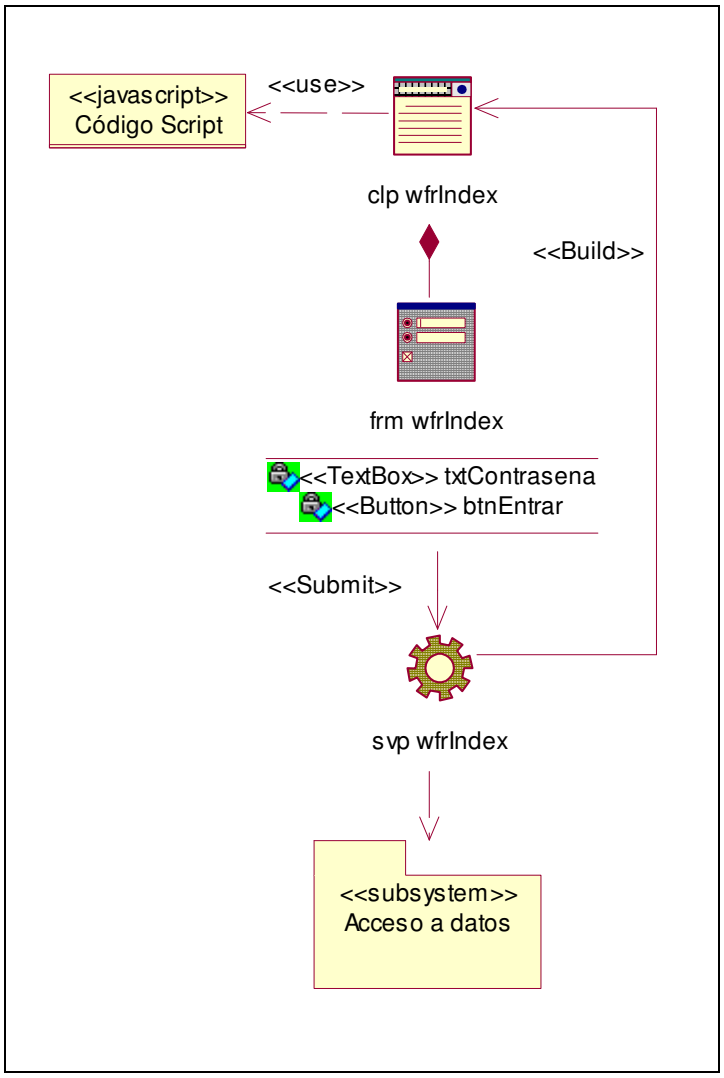


Figura 3.3 Diagrama de clases del paquete CU Autenticar Usuario.

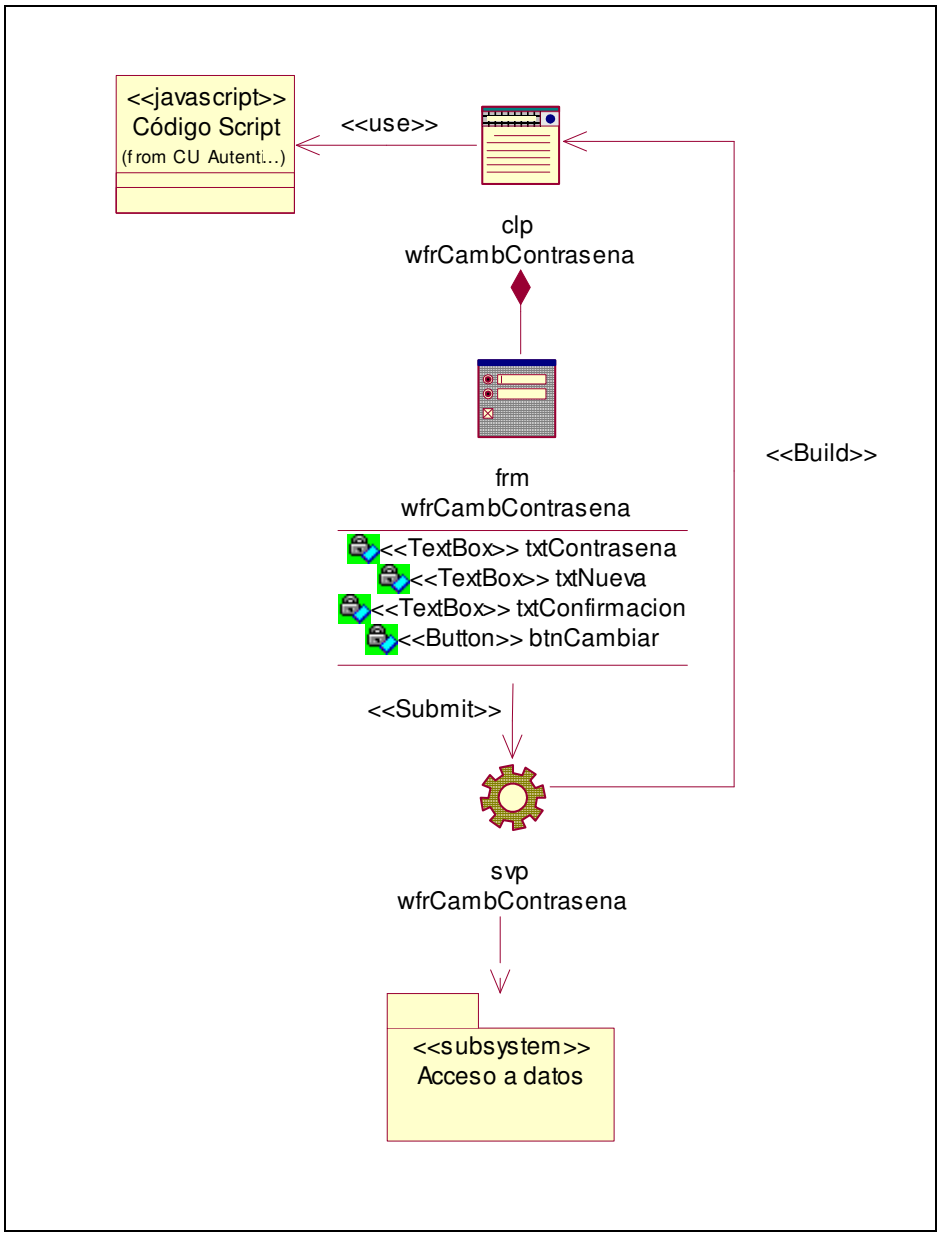


Figura 3.4 Diagrama de clases del paquete CU Cambiar Contraseña.

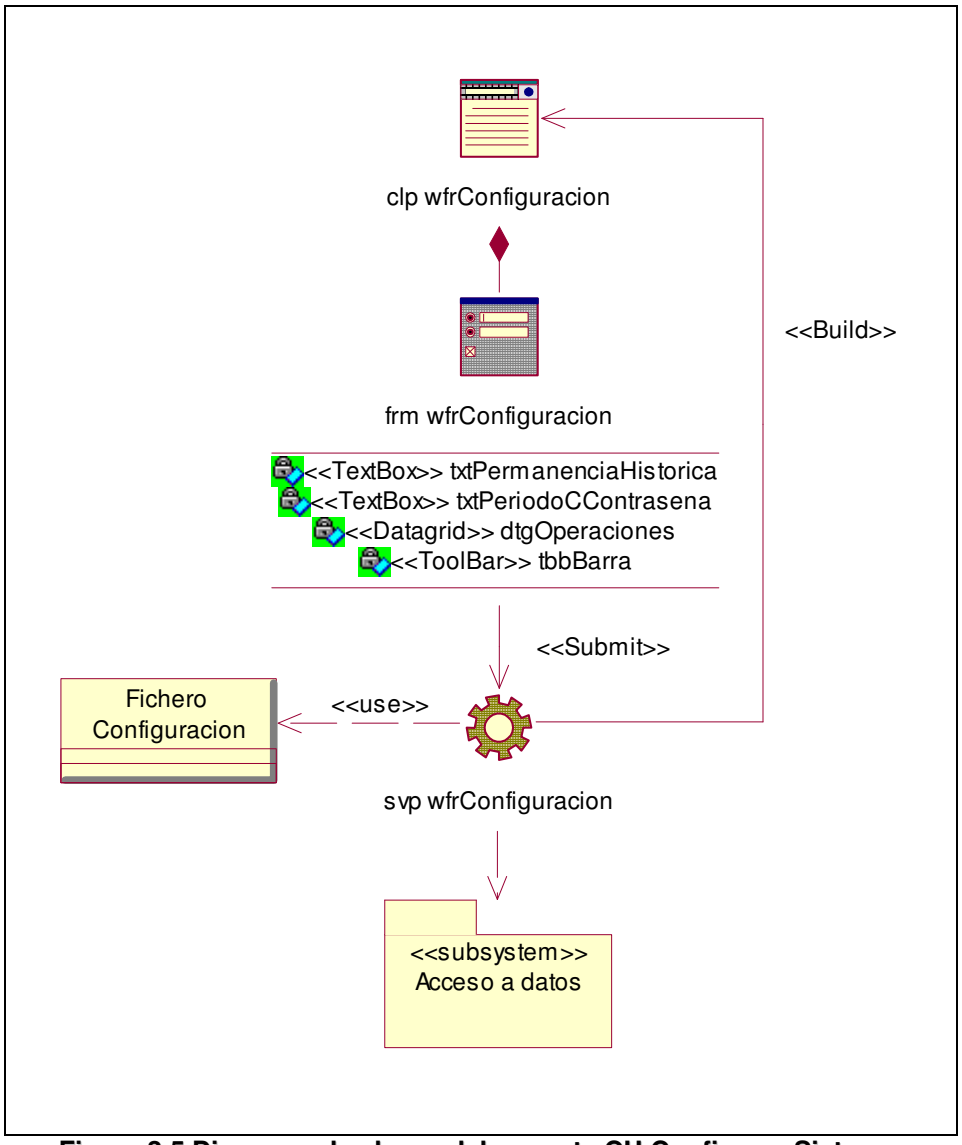


Figura 3.5 Diagrama de clases del paquete CU Configurar Sistema.

3.2.1.3 Subsistema de Actualización

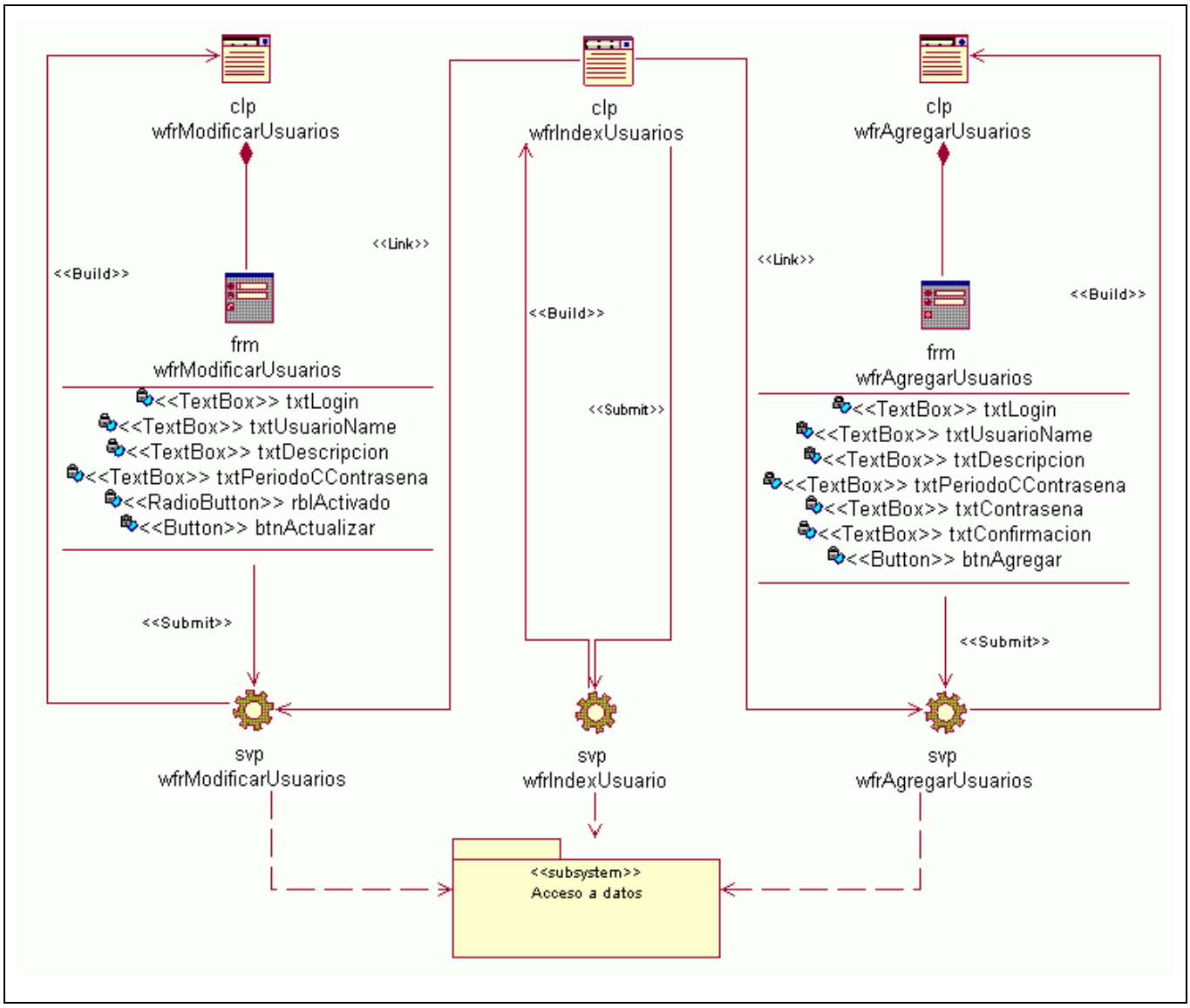


Figura 3.6 Diagrama de clases del paquete CU Administrar Usuarios.

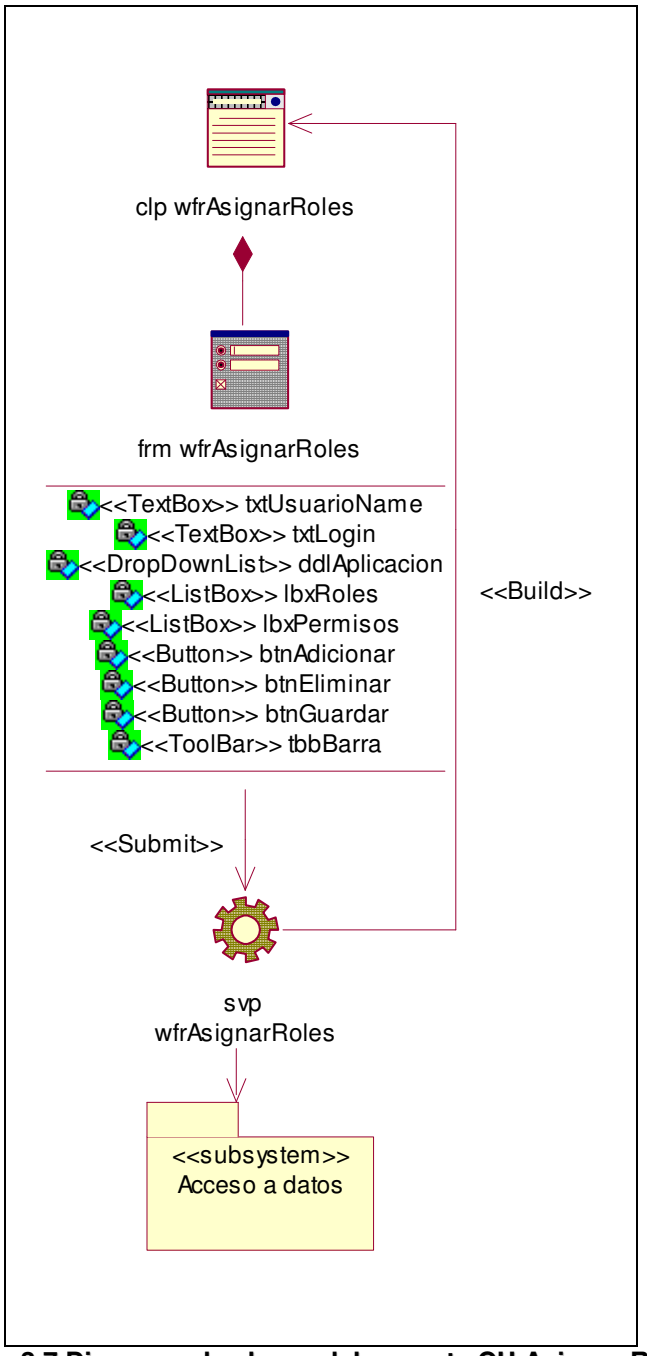


Figura 3.7 Diagrama de clases del paquete CU Asignar Roles.

3.2.1.4 Subsistema de Reportes

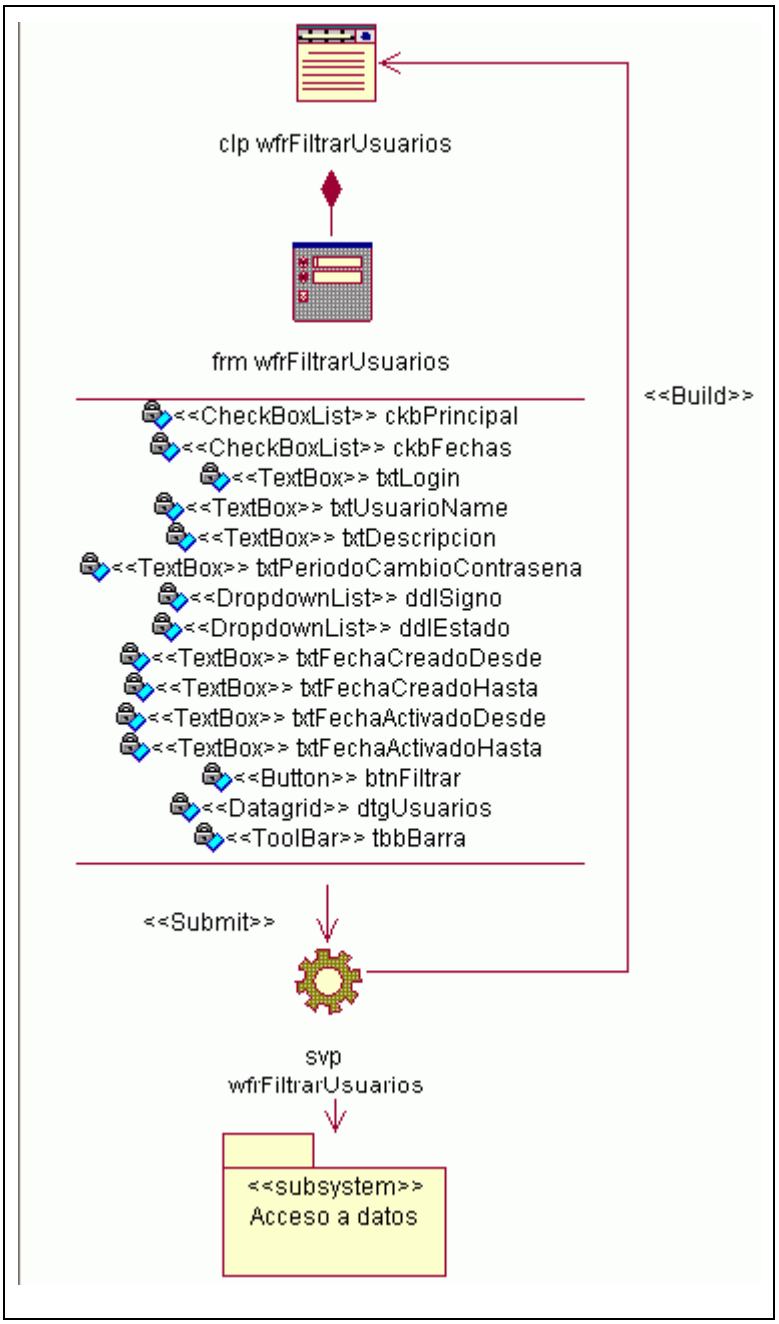


Figura 3.8 Diagrama de clases del paquete CU Visualizar Reporte de Usuarios.

3.2.1.5 Subsistema de Servicios

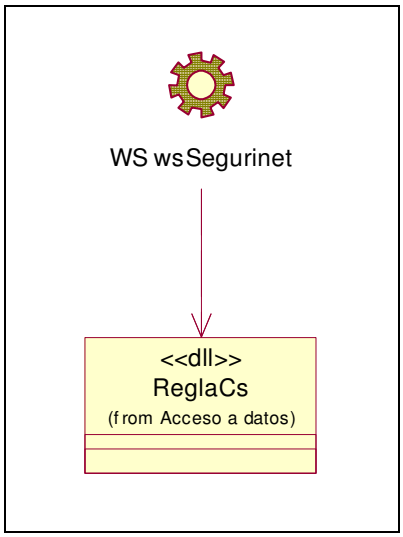


Figura 3.9 Diagrama de clases Subsistema de Servicios.

3.2.2 Diagrama de clases persistentes

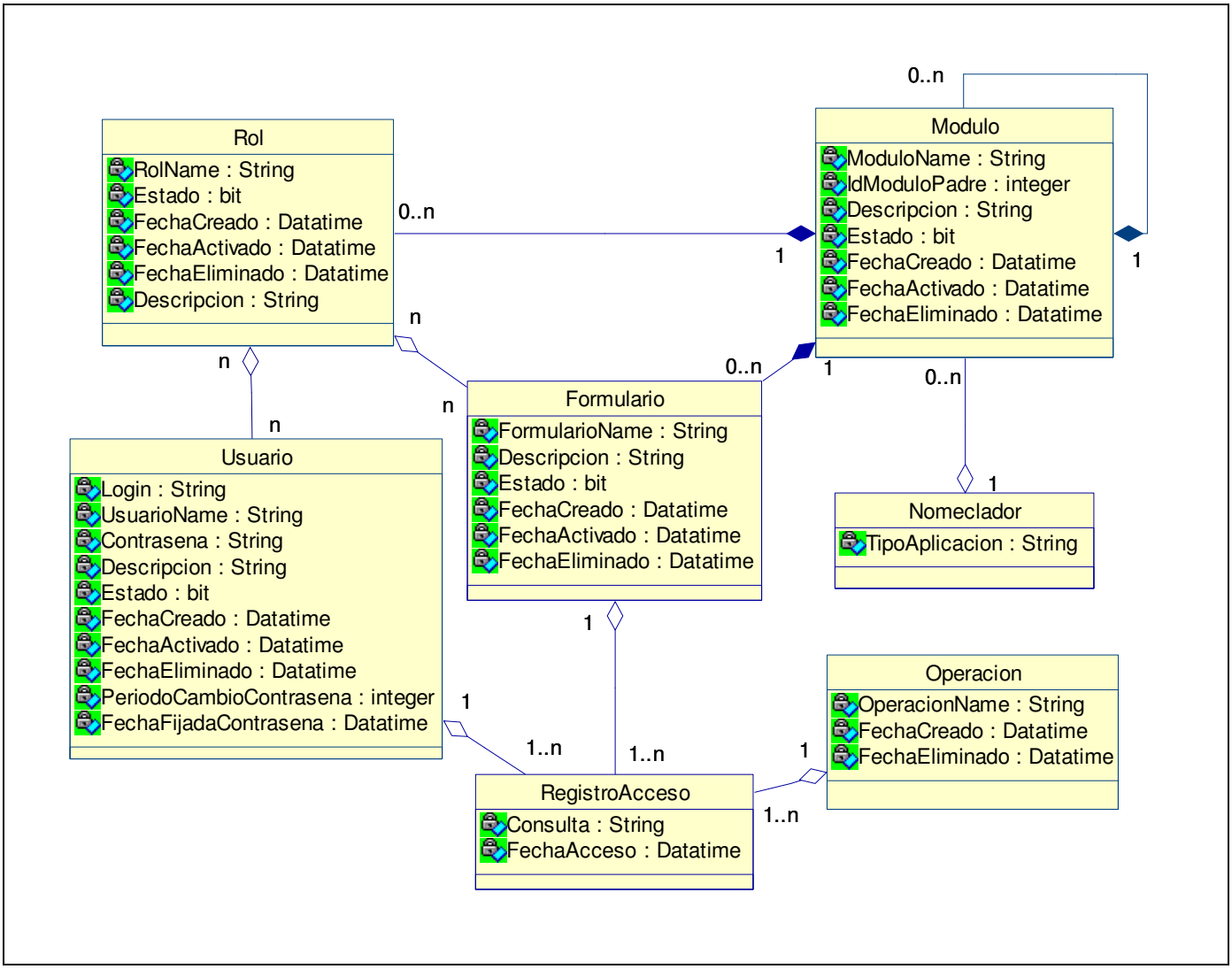


Figura 3.10 Diagrama de clases persistentes.

3.2.3 Diagrama del modelo de datos

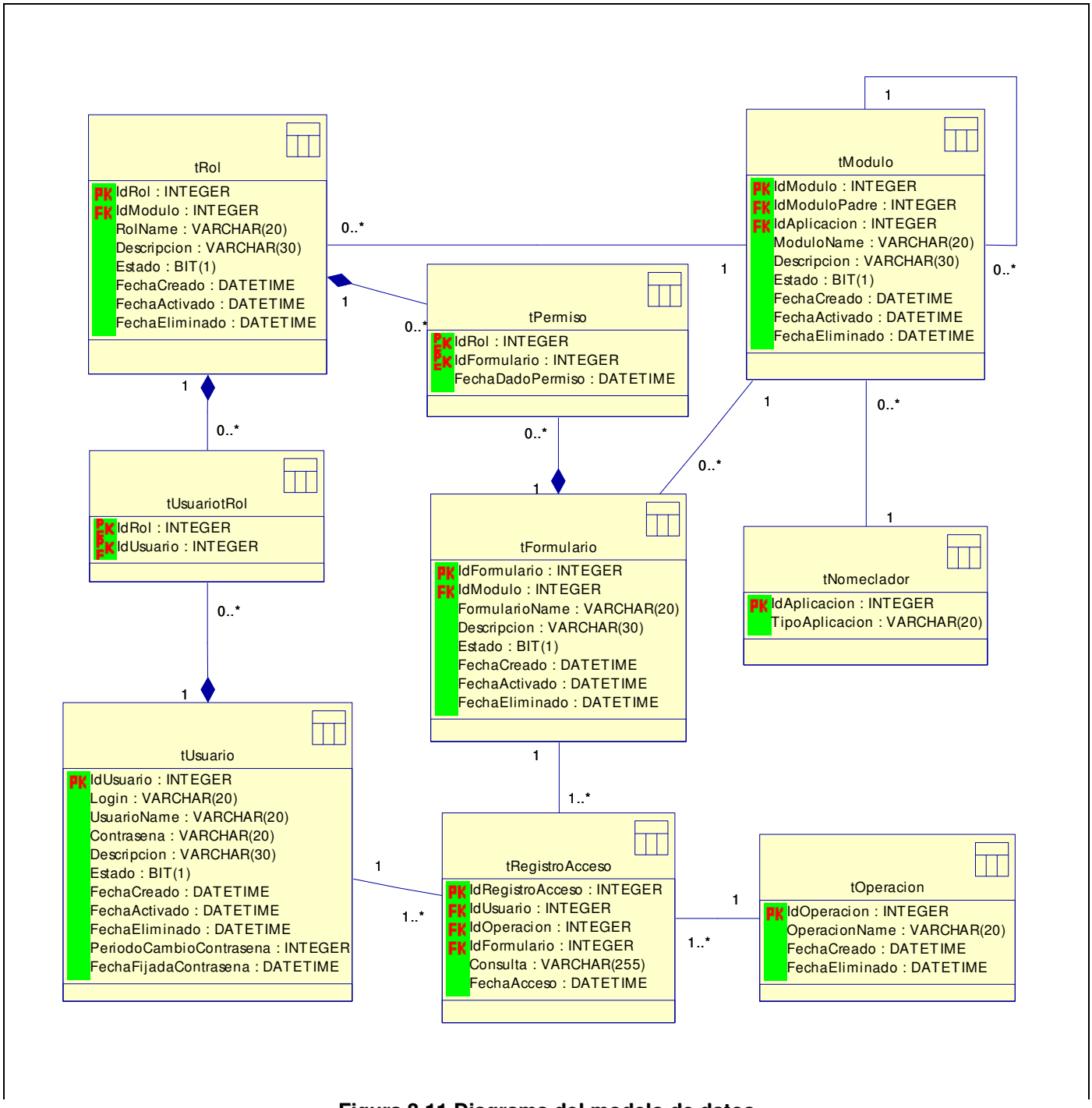


Figura 3.11 Diagrama del modelo de datos

3.2.4 Principios de diseño

3.2.4.1 Interfaz de usuario

La interfaz de del sistema SEGURINET, en correspondencia de las normas básicas de diseño para aplicaciones Web, se desarrolla empleando tonalidades de azul, blanco y gris en su diseño gráfico, marcando las mismas pautas y estilos que representan las demás aplicaciones del INSMET. La distribución del contenido se representa en tres partes fundamentales:

- El ***banner*** ilustrativo que identifica a SEGURINET incluyendo la fecha actual y las acciones básicas generales del sistema como Cambiar contraseña, Ayuda y Salir.

- El ***menú lateral*** que muestra todas las opciones a las que puede acceder el administrador en la utilización del sistema. Para su diseño se utilizaron metáforas, acompañadas de información escrita, y su ubicación en la aplicación, formando parte del marco izquierdo, garantiza una fácil navegación hacia cualquier parte del sistema.

Figura 3.12 Iconos del menú lateral.

- El **marco de trabajo** que representa el soporte principal de operaciones generales del sistema. Ocupa el mayor espacio en la aplicación ya que incluye todas las actividades del funcionamiento de SEGURINET.

Se utiliza una hoja de estilos para guardar la configuración del diseño para todas las páginas y componentes del sistema, eliminando así el número de imágenes que ralenticen la presentación de la página. Como tipo de letra se combina Verdana y Tahoma de tamaño 11, permitiendo una separación óptima entre las letras para presentar información.

3.2.4.2 Formato de salida de los reportes

Los reportes tienen como componentes principales a los datagrids (grillas), llevan paginado de 5 y 10, dependiendo de la cantidad de información que se muestre; esta organizado en números consecutivos, y tienen el enlace a los siguientes resultados, excepto el que está activo en el momento. Se utilizan pequeñas imágenes para presentar el contenido y como funcionalidad se le agrega el ordenamiento por campos.

	Rol	Descripción	Estado	Fecha de creado	Fecha de activado
	Administrador general	El jefe de todo esto	Activo	17-Apr-06	17-Apr-06
	Administrador de Modulos	secretario de modulos	Activo	17-Apr-06	17-Apr-06
	PowerUser	Puede hacer alguito	Activo	17-Apr-06	17-Apr-06
	Invitado	Un fula de la vida	Activo	17-Apr-06	17-Apr-06
	YoenisPantoja	sdfd	Activo	25-Apr-06	03-May-06
	prueba	aa	Activo	03-May-06	03-May-06
	Rol Nuevo	aa	Activo	03-May-06	03-May-06

1

Figura 3.13 Formas de representar los reportes.

3.2.4.3 Ayuda

La ayuda está accesible como parte del menú de acciones generales que se representa en la región superior en todas las páginas de la aplicación, y con el fin de que el usuario vea solo la información que necesita en ese momento, cada página muestra como realizar solo aquellas operaciones que se estén realizando en la página específica, además se aportan los conceptos que se manejan en la aplicación para que el usuario se familiarice con algunas entradas.

3.2.4.4 Tratamiento de errores.

SEGURINET evita por todas las vías los errores y las excepciones síncronas que puedan surgir en la parte de la interacción del usuario con los distintos formularios de la aplicación. Para ello se basa en una fuerte validación de los controles, tanto a nivel de cliente como en el servidor, garantizando la entrada correcta de la información para su tratamiento.

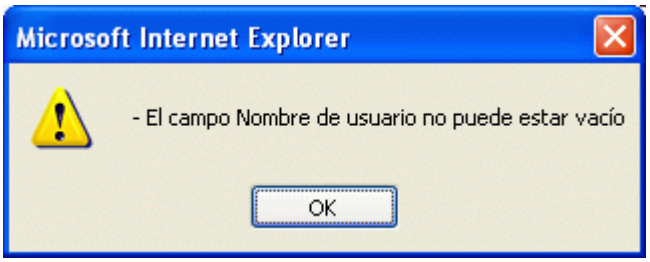


Figura 3.14 Mensaje de advertencia para evitar posibles errores síncronos.

Los errores y situaciones excepcionales que se presenten de carácter asíncrono se tratan en una página especial que incluye un fichero con la información detallada que recoge una descripción completa de las causas que provocaron el fallo específico.

3.2.4.5 Estándar de codificación

Toda codificación de la aplicación, incluyendo propiedades de los componentes, diseño de tablas, procedimientos almacenados y hoja de estilos, está sustentada y organizada con la aplicación de las normas de codificación para proyectos .NET de la compañía IVAD-SOFT.

3.3 Modelo de implementación

Entre los artefactos que se construyen durante la fase de implementación se encuentra el Modelo de Implementación, el cual describe los componentes y la organización de acuerdo a los nodos, así como las dependencias de funcionamiento entre ellos.

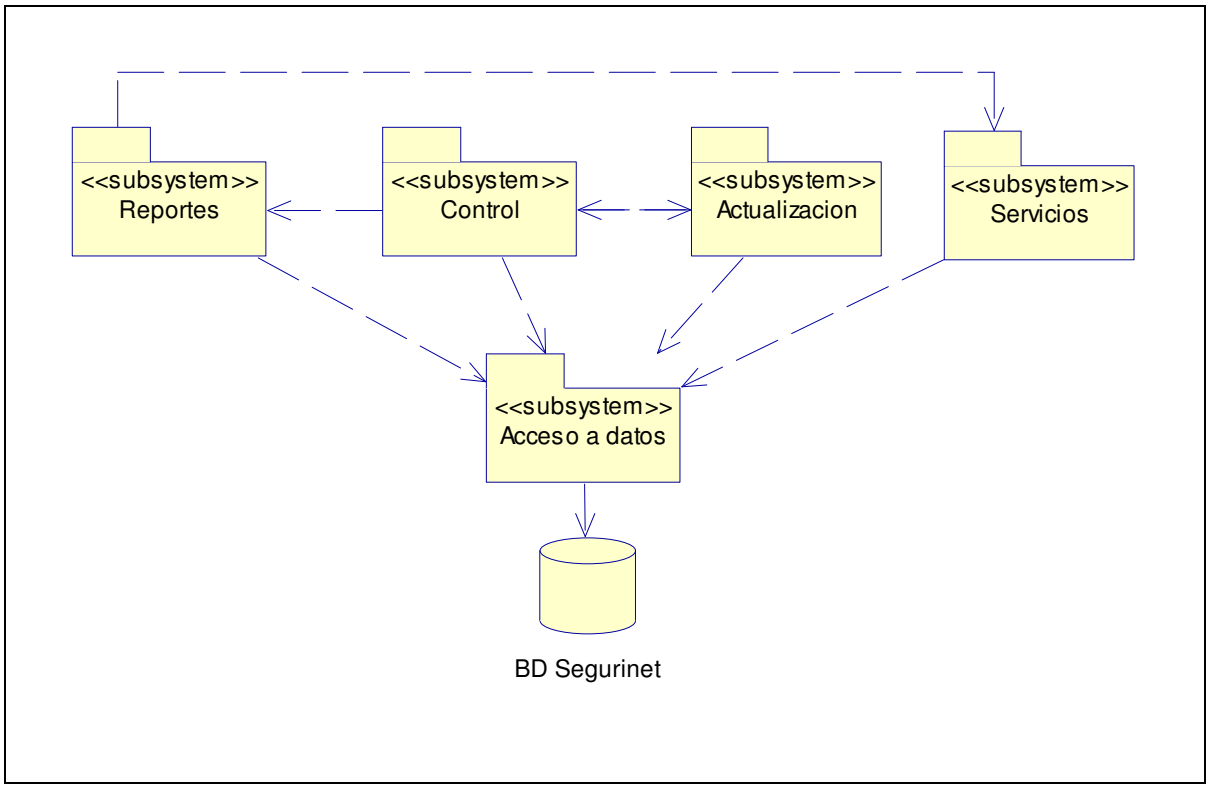


Figura 3.15 Subsistemas del Modelo de Implementación

3.3.1 Diagrama de componentes por subsistemas

3.3.1.1 Subsistema de Control

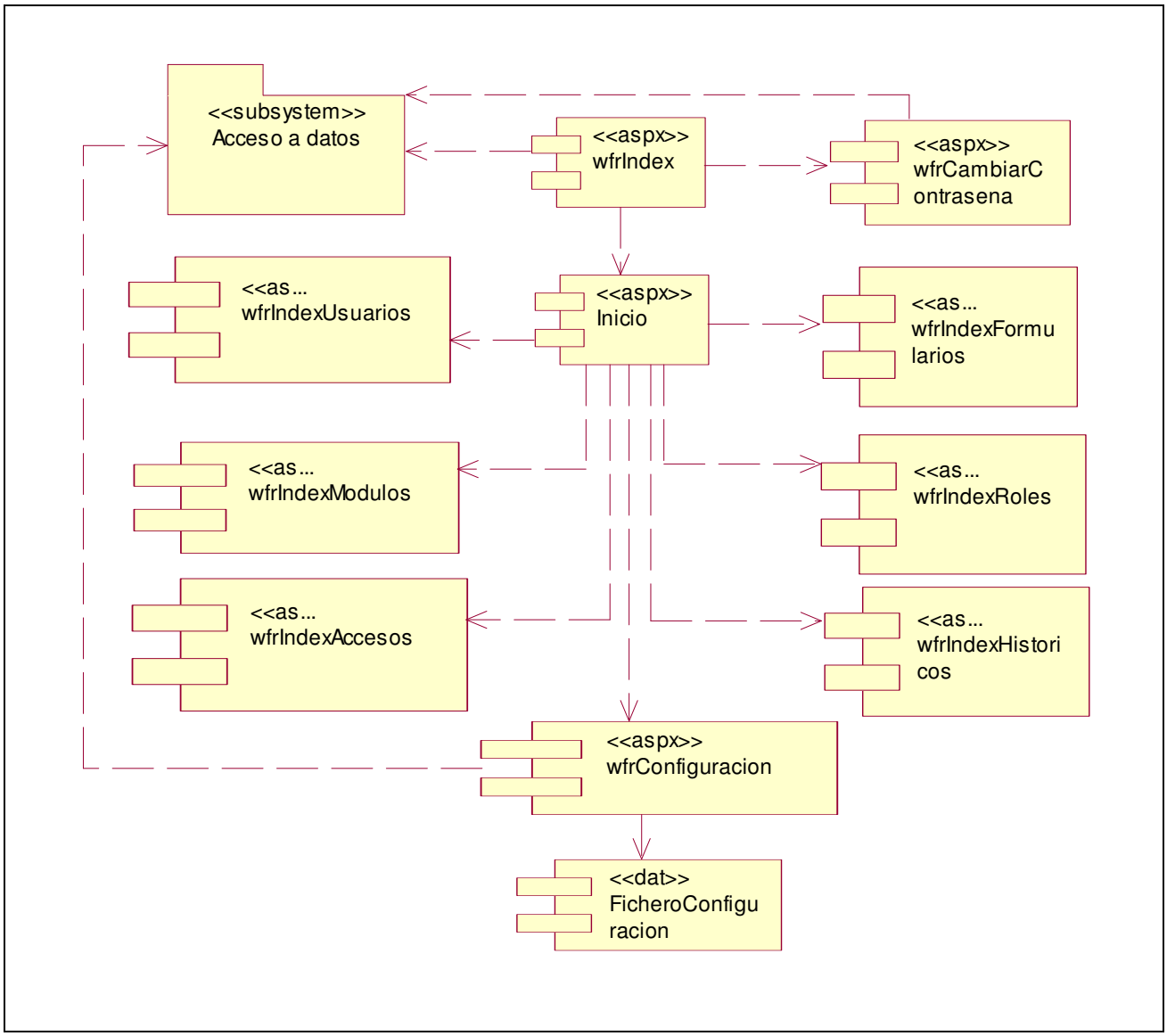


Figura 3.16 Diagrama de componentes del subsistema de Control

3.3.1.2 Subsistema de Actualización

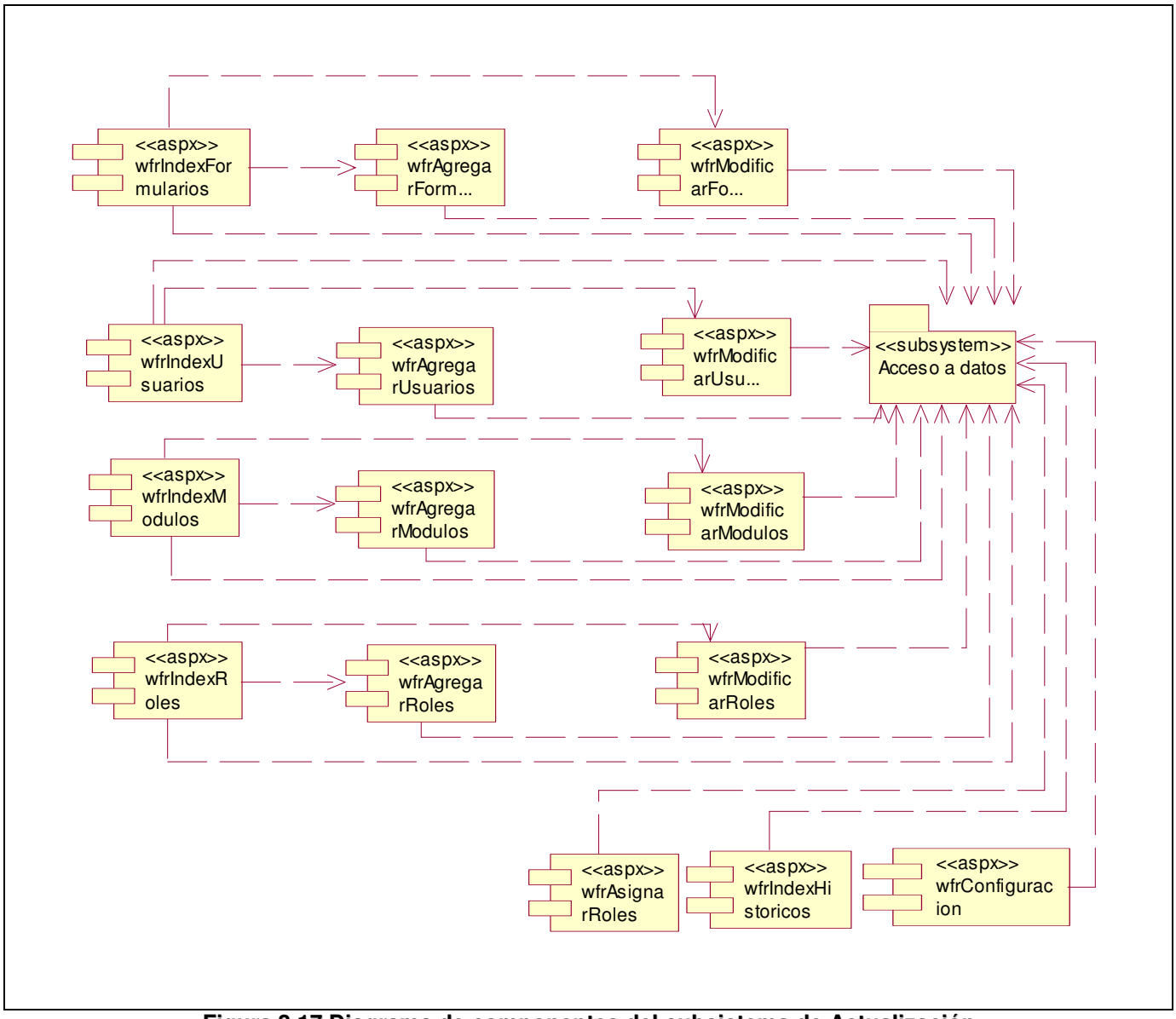


Figura 3.17 Diagrama de componentes del subsistema de Actualización

3.3.1.3 Subsistema de Reportes

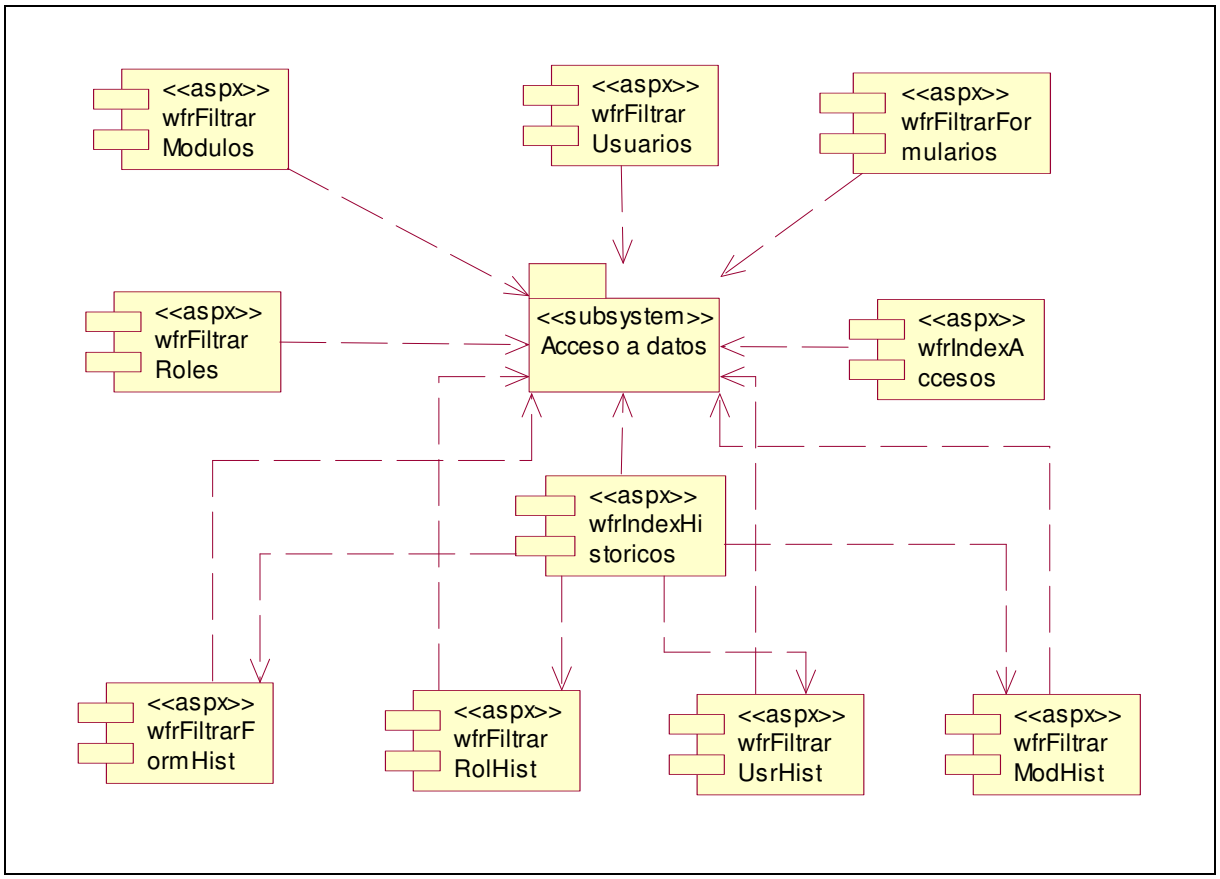


Figura 3.18 Diagrama de componentes del subsistema de Reportes

3.3.1.4 Subsistema de Servicios

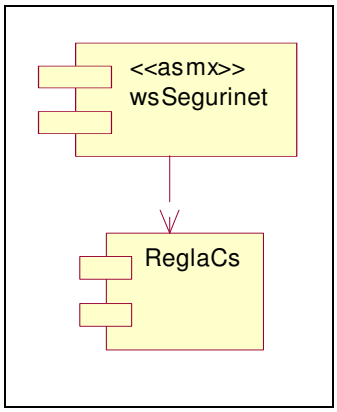


Figura 3.19 Diagrama de componentes del subsistema de Reportes

3.4 Modelo de despliegue

SEGURINET está desarrollado sobre la bases de la lógica de funcionamiento del patrón arquitectónico Capas (*Layer Pattern*) para definir su arquitectura, en tres capas, con un estilo arquitectónico orientado a servicios.

La figura 4.3 muestra el diagrama de despliegue. El nodo *Servidor de BD* representa un servidor SQL Server, en el cual se ubica toda la información persistente del sistema, en el *Servidor Web*, Internet Information Services, se ubican íntegramente, las capas de presentación, lógica del negocio y de acceso a datos del sistema, así como los servicios que se brindan. *Terminales clientes* representan el conjunto de computadoras desde las cuales se puede administrar y usar los servicios disponibles, es un nodo con capacidad de procesamiento, al igual que los anteriores, porque es donde se interpretan las respuestas a las solicitudes hechas al servidor en forma de HTML y donde se ejecutan códigos JavaScript para validaciones y encriptaciones de información. *Impresora* representa un dispositivo de esta naturaleza que dispone de conexión física con las Terminales Clientes para obtener copias duras de reportes suministrados por SEGURINET.

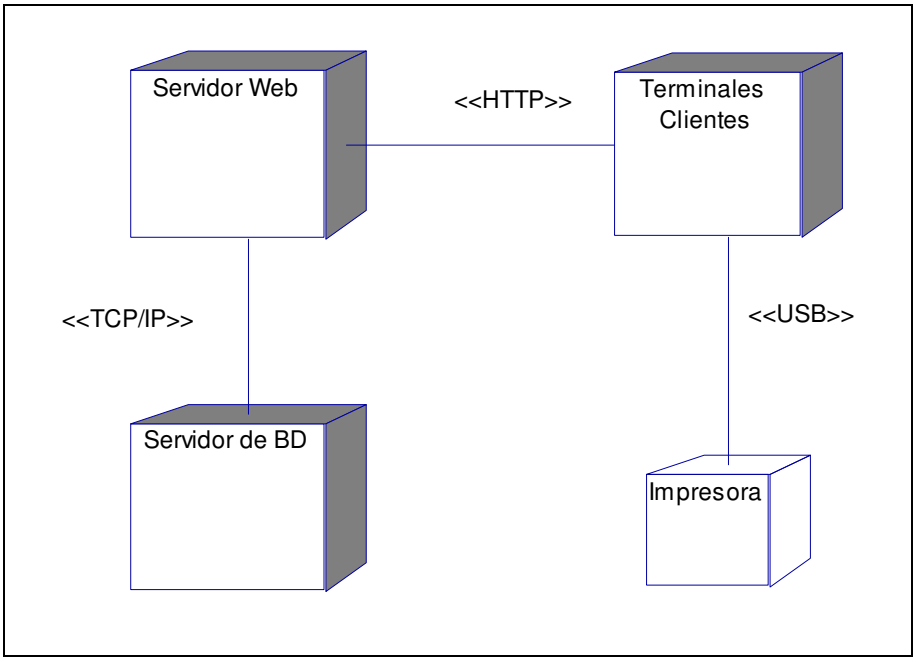


Figura 3.20 Diagrama de despliegue

3.5 Conclusiones

En este capítulo se modeló el sistema y se encontró su forma, incluida la arquitectura, que determina el soporte de todos los requerimientos, incluyendo los requisitos no funcionales y otras restricciones. Se expresó el sistema en términos de componentes agrupados en subsistemas encontrados en el diseño. Con la fundamentación de todos estos elementos se crean las condiciones para la etapa de implementación de la aplicación.

Capítulo 4

Estudio de factibilidad

4.1 Introducción

El estudio de factibilidad es un paso importante que no se debe obviar en la realización de un proyecto, pues como resultado de este análisis se obtienen las estimaciones de: esfuerzo, tiempo de desarrollo en meses, costo del producto, la cantidad de personas que se necesitan para desarrollar el proyecto, entre otras; sirviendo de elemento esencial de planificación para el equipo de trabajo y posibilitando fijar con los clientes una fecha de terminación del producto.

En este capítulo se describe la estimación de costos del sistema propuesto y sus beneficios, basado en las técnicas de Análisis de Puntos de Casos de Uso.

4.2 Planificación basada en casos de uso. Análisis de Puntos de Casos de Uso.

Paso 1. Cálculo de los Puntos de casos de uso Desajustados.

UUCP=UAW+UUCW

Donde:

- UUCP: Puntos de casos de uso sin ajustar.
- UAW: Factor de peso de los actores sin ajustar.
- UUCW: Factor de peso de los casos de uso sin ajustar.

Tipo de actor	Descripción	Factor de peso	Actores	Total
Simple	Sistema con sistema a través de interfaz de programación.	1	0	0

Medio	Sistema con sistema mediante protocolo de interfaz basada en texto.	2	1	2
Complejo	Persona que interactúa con el sistema mediante interfaz gráfica.	3	1	3

$$UAW = \sum cant\ \ actores * peso$$

UAW=5

Tipo de CU	**Descripción**	**Peso**	**Cantidad de CU**	**Total**
Simple	El caso de uso tiene de 1 a 3 transacciones.	5	15	75
Medio	El caso de uso tiene de 4 a 7 transacciones.	10	4	40
Complejo	El caso de uso tiene más de 8 transacciones.	15	2	30

$$UUCW = \sum cant\,CU * Peso$$

UUCW=145

UUCP=5+145

UUCP=150

Paso 2. Cálculo de los Puntos de casos de uso ajustados.

UCP=UUCP*TCF*EF

Donde:

- UCP: Puntos de casos de uso ajustados.
- UUCP: Puntos de casos de uso sin ajustar.
- TCP: Factor de complejidad técnica.
- EF: Factor de ambiente.

El factor de complejidad técnica (TCF) se calcula mediante la cuantificación de un conjunto de factores que determinan la complejidad técnica del sistema. Cada factor se cuantifica en un valor desde 0 (aporte irrelevante) hasta 5 (aporte muy relevante).

Factor	Descripción	Peso	Valor asignado	Total
T1	Sistema distribuido	2	0	0
T2	Tiempo de respuesta	1	4	4
T3	Eficiencia del usuario final	1	3	3
T4	Funcionamiento Interno complejo	1	3	3
T5	El código debe ser reutilizable	1	4	4
T6	Facilidad de instalación	0,5	4	2
T7	Facilidad de uso	0,5	5	2,5
T8	Portabilidad	2	0	0
T9	Facilidad de cambio	1	4	4
T10	Concurrencia	1	5	5
T11	Incluye objetivos especiales de seguridad	1	5	5
T12	Provee acceso directo a terceras partes	1	0	0

T13	Se requieren facilidades especiales de entrenamiento de usuarios	1	2	2

$$TCF = 0.6 + 0.01 * \sum (peso * valor\ asignado)$$

TCF = 0.6 + 0.01 * 34.5

TCF = 0.6 + 0.345

TCF=0.945

El factor de ambiente (EF) está relacionado con las habilidades y entrenamiento del grupo de desarrollo que realiza el sistema. Cada factor se cuantifica con un valor desde 0 (aporte irrelevante) hasta 5 (aporte muy relevante).

Factor	**Descripción**	**Peso**	**Valor asignado**	**Total**
E1	Familiaridad con el modelo de proyecto utilizado	1,5	3	4.5
E2	Experiencia en la aplicación	0,5	3	1.5
E3	Experiencia en la orientación a objetivos.	1	4	4
E4	Capacidad del analista líder.	0,5	2	1
E5	Motivación.	1	4	4
E6	Estabilidad de requerimientos	2	4	8
E7	Personal Part–Time	-1	5	-5
E8	Dificultad del lenguaje de programación	-1	1	-1

$$EF = 1.4 - 0.03 * \sum (peso * valor\ asignado)$$

EF = 1.4 – 0.03 * 17

EF = 1.4 – 0.51

EF=0.89

UCP = UUCP * TCF * EF

UCP = 150 *0.945 * 0.89

UCP=126.1575

Paso 3. Estimación de esfuerzo a través de los puntos de casos de uso.

E=UCP*CF

Donde:

- E: Esfuerzo estimado en horas hombres.
- UCP: Punto de casos de usos ajustados.
- CF: Factor de conversión.

Para obtener el factor de conversión (CF) se cuentan cuantos valores de los que afectan el factor ambiente (E1...E6) están por debajo de la media (3), y los que están por arriba de la media para los restantes (E7, E8). Si el total es 2 o menos se utiliza el factor de conversión 20 Horas-Hombre / Punto de Casos de uso. Si el total es 3 o 4 se utiliza el factor de conversión 28 Horas-Hombre / Punto de Casos de uso. Si el total es mayor o igual que 5 se recomienda efectuar cambios en el proyecto ya que se considera que el riesgo de fracaso del mismo es demasiado alto.

En este caso se puede decir que:

CF = 20 Horas-Hombre / Punto de Casos de uso.

E=126.1575*20

E=2523.15 Horas-Hombre

Paso 4. Calcular esfuerzo de todo el proyecto.

Actividad	**Porcentaje %**	**Horas-Hombres**

Análisis	10	630.7875
Diseño	20	1261.575
Implementación	40	2523.15
Pruebas	15	946.18125
Sobrecarga (otras actividades)	15	946.18125
Total	100	6307.875

Si E_T = 6307.875 horas-hombre y cada mes los desarrolladores trabajan como promedio 144 horas, eso daría un:

E_T = 43.8046875 mes-hombre.

Esto quiere decir que 1 persona puede realizar el problema analizado en 44 meses aproximadamente.

Costo del Proyecto.

Se asume como salario promedio mensual $50.00

CHM = 2 * Salario Promedio

CHM = 100.00 $/mes

Costo = Salario Promedio* E_T

Costo = 50.00 * 43.8046875

Costo= $ 2190.23438.

4.3 Beneficios tangibles e intangibles.

La implantación del software propuesto trae consigo una serie de beneficios fundamentalmente intangibles al INSMET, pues permitirá mantener el control detallado y organizado sobre todas sus aplicaciones y de los usuarios que hacen uso de ellas, proporcionará un punto de partida para brindar seguridad a su información y un control

avanzado de la administración de usuarios y aplicaciones, disminuirá los costos de producción de otras aplicaciones que sean para el consumo del centro o producidas por la UCI bajo estos principios, los usuarios finales se sentirán satisfechos al tener un identificador y contraseña únicos para acceder a sus respectivos sistemas. En el caso de los beneficios tangibles queda en manos de la dirección de la Universidad de Ciencias Informáticas la decisión de comercializar o no el software una vez terminado, pues es de gran utilidad a toda organización que cuente con un número medio de trabajadores que tengan acceso a múltiples sistemas que necesiten manipular el acceso a través de usuarios.

4.4 Análisis de costos y beneficios.

Desarrollar un producto informático cuesta. Justificar entonces su desarrollo depende de los beneficios que reportarían su implantación y uso.

La utilización de este sistema para controlar la administración de los usuarios del Sistema de Gestión de la Información del INSMET brinda una útil herramienta que en primer orden protege información sensible y valiosa que hasta el momento carecía de seguridad, en caso del surgimiento de nuevas aplicaciones optimizaría el proceso de administración de usuarios explotando su característica de sistema centralizado, evitando el almacenamiento redundante de datos de usuarios que pertenecen a un mismo ámbito y posibilitando que se pueda acceder a todos los sistemas disponibles con un mismo nombre de usuario y contraseña; gestionando la seguridad de una forma natural, según los roles que desempeña cada usuario en el negocio y con solo un costo de **$ 2190.23438.**

Este sistema permitirá detectar posibles violaciones de seguridad y reconstruir escenarios ante cualquier novedad, mantendrá una buena política de contraseñas que incluyen restricciones de longitud, composición y vigencia así como la encriptación de las mismas para evitar ataques en este sentido.

Por tanto se decide que es factible desarrollar una herramienta que proporcione estas ventajas.

4.5 Conclusiones.

En este capítulo se describió el estudio de factibilidad realizado correspondiente al sistema propuesto, teniendo en cuenta el costo estimado y los beneficios que reportará al ser implantado.

La herramienta propuesta reportará beneficios significativos e importantes para la gestión segura de la información del INSMET, lo que indica que es factible implementar la herramienta propuesta.

Conclusiones

Con el desarrollo del SEGURINET, como sistema genérico de seguridad para aplicaciones Web y adaptación para el sistema de gestión del INSTMET se da cumplimiento a los objetivos de este trabajo, pues da camino a la obtención de una aplicación en la que se aplican los resultados de todo el proceso de investigación realizado a lo largo de las etapas del proyecto, lográndose:

- Un mecanismo de autenticación de usuarios para identificarlos.
- Un mecanismo de control de acceso a la información que garantiza que los usuarios de la entidad manipulen la información que les es accesible por regla.
- Un proceso automatizado de auditoría de las violaciones en el acceso a la información, basado en reportes, de los usuarios en la manipulación de esta.
- La posibilidad de realizar la administración centralizada de los perfiles de usuarios para el posible conjunto de aplicaciones con las que contará la entidad en un futuro.
- La creación de un nuevo subsistema o módulo del sistema general de gestión que realice estas funciones de control general de los usuarios y sus claves, facilitándoles el acceso a los distintos subsistemas con los que debe contar el sistema general.
- Se evita que en el futuro cada sistema tenga que implementar sus propios elementos de seguridad, disminuyendo el costo y tiempo de desarrollo.
- Se permite concentrar todos los esfuerzos de seguridad en un solo punto, logrando genericidad, consistencia de la información y garantía contra las violaciones a nivel de sistemas de jerarquía menor.
- Se garantiza que los usuarios de las aplicaciones se beneficien de las comodidades que brinda el uso de un usuario único para el acceso a cualquier sistema que se integre con SEGURINET.
- La posibilidad a la entidad que emplee este sistema de un mejor control de las aplicaciones con que cuenta, permitiendo conocer en todo momento la totalidad de

módulos, submódulos y formularios registrados en el sistema y de los usuarios que acceden a los mismos.

- Un sistema al cual se pueden incorporar todas las aplicaciones que requieran los servicios de seguridad siempre que puedan usar los Servicios Web elaborados bajo la filosofía de SOAP.

Recomendaciones

De acuerdo a los resultados de todo el proceso de investigación realizado y basados en la experiencia acumulada se proponen las siguientes recomendaciones:

- Ampliar a nuevas funcionalidades el sistema SEGURINET como pueden ser las restricciones de horarios o lugar de uso para las aplicaciones integradas.
- Migrar la arquitectura del sistema hacia plataformas libres, para lograr soluciones de mejor costo económico desde el punto de vista funcional para los desarrolladores de software en el país y facilitar su portabilidad.
- Desarrollar procesos de análisis estadístico de actividad de los usuarios.

Referencias bibliográficas

- [[1] Angelfire.com, "Seguridad de aplicaciones.," 2006. [http://www.angelfire.com/ri2/aspectos/Tesis/Final.pdf]
- [2] IMPERVA, "Seguridad de aplicaciones Web," 2006. [http://www.imperva.com/application_defense_center/papers/]
- [3] R. M. Marcus Hennecke, Herb Swan, "Ataques vía web," 2002. [http://es.tldp.org/Manuales-LuCAS/doc-unixsec/unixsec-html/node280.html]
- [4] I. E. G. González, "Los 2 niveles de ataque a servidores web," 2000. [http://www.monografias.com/trabajos30/2-niveles-ataque-servidores-web/2-niveles-ataque-servidores-web.shtml#_Toc124825054]
- [5] J. Pestberg, "Seguridad del hardware," 2001. [http://www.redhat.com/docs/manuals/enterprise/RHEL-4-Manual/es/security-guide/ch-netprot.html]
- [6] TbSecurity.com, "Principales Riesgos.".[http://www.tb-security.com/riesgos.htm]
- [7] SPIDinamics.com, "AMP," 2006. [http://www.spidynamics.com/products/amp/index.html]
- [8] SPIDinamics.com, "QAInspect," 2006. [http://www.spidynamics.com/products/qainspect/index.html]
- [9] SPIDinamics.com, "DevInspect," 2006. [http://www.spidynamics.com/products/devinspect/index.html]
- [10] INSMET, "Mision del ISMET," 2006. [http://www.met.inf.cu/]
- [11] G. B. Ivar Jacobson, James Rumbaugh, *El Proceso Unificado de Desarrollo de Software*, 2000.
- [12] MSDN.com, "Arquitectura de Software." [http://www.microsoft.com/spanish/msdn/arquitectu.a/roadmap_arq/arquitectura_soft.asp]

- [13] Departamento_Ingeniería_de_Software, "Conferencia Arquitectura 1.ppt," 2006. [http://docencia.uci.cu/is/cgi-bin/admins/coments/files/10131.rar]
- [14] Departamento_Ingeniería_de_Software, "Patrones de arquitectura," 2006. [http://ucimedia.uci.cu/teleclases/1er_Semestre/3er/Ingenieria_de_Software_I/conf7
- [15] Wikipedia, "Servicios Web," 2006. [http://es.wikipedia.org/wiki/Servicio_Web]
- [16] G. A. Marañón, "Seguridad en servicios web ", 2006. [http://www.instisec.com/publico/verarticulo.asp?id=70]
- [17] D. P. Francisco Recio, "Common Language Runtime (CLR)," 2000. [http://www.desarrolloweb.com/articulos/1328.php?manual=48]
- [18] J. A. G. Seco, "El lenguaje de programación C#. ." [http://www.josanguapo.com/librocsharp2.zip]
- [19] R. Martin, "Designing SQL Server 2000. Databases for .net Enterprises Servers," Syngress ed, 2001.
- [20] G. Booch, Rumbaugh, J., Jacobson, I. , *El Lenguaje Unificado de Modelado".* , 1999.

Glosario de términos

- **WebService**: Sistema de software diseñado para soportar interacción máquina-a-máquina sobre una red. Posee una interfaz descrita en un formato procesable por máquina.
- **Trigger:** Un trigger o un disparador es un evento que se ejecuta en una base de datos cuando se cumple una condición establecida al realizar una operación de inserción, actualización o borrado.
- **Cross Site Scripting:** Tipo de vulnerabilidad surgida como consecuencia de errores de filtrado de las entradas del usuario en aplicaciones Web. También es conocido como XSS.
- **Telnet:** Es el nombre de un protocolo (y del programa informático que implementa el cliente) que sirve para acceder mediante una red a otra máquina, para manejarla como un terminal de conexión remoto.
- **SSH:** (Secure SHell) es el nombre de un protocolo y del programa que lo implementa, y sirve para acceder a máquinas remotas a través de una red. Permite manejar por completo el ordenador mediante un intérprete de comandos, y también puede redirigir el tráfico para poder ejecutar programas gráficos si tenemos un Servidor arrancado. Permite copiar datos de forma segura, gestionar claves RSA y pasar los datos de cualquier otra aplicación por un canal seguro.
- **Transact-SQL:** (T-SQL) es el lenguaje de programación del SQL Server.
- **DDL:** (Data Definition Lenguaje) Lenguaje de definición de datos. Las sentencias DDL son aquellas utilizadas para la creación de una base de datos y todos sus componentes: tablas, índices, relaciones, disparadores, procedimientos almacenados, etcétera.
- **DML:** (Data Manipulation Lenguaje) Lenguaje de manipulación de datos. Las sentencias DML son aquellas utilizadas para insertar, borrar, modificar y consultar los datos de una base de datos.

- **CORBA:** (Common Object Request Broker Architecture) Es un estándar que establece una plataforma de desarrollo de sistemas distribuidos facilitando la invocación de métodos remotos bajo un paradigma orientado a objetos
- **DCOM:** (Distributed Component Object Model) Modelo de Objeto Componente Distribuido. Es un juego de conceptos e interfaces de programa de Microsoft en el cual los objetos de programa del cliente pueden solicitar servicios de objetos de programa servidores en otros ordenadores dentro de una red.
- **Datagrid:** Es un control o componente Web que muestra los datos en formato de tabla y, de manera opcional, admite la selección, ordenación, paginación y edición de los mismos.

Anexo 1 Descripción de los casos de uso

Caso de uso:	**Cerrar sesión.**
Actores:	Administrador de SEGURINET.
Propósito:	Cerrar la aplicación SEGURINET.
Resumen:	El Administrador de SEGURINET sale definitivamente del sistema.
Referencias:	R3
Precondiciones:	El Administrador de SEGURINET se encuentra previamente autenticado en el sistema.
Poscondiciones:	El sistema es cerrado.
Caso de uso asociado:	

Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El caso de uso comienza cuando el Administrador de SEGURINET desea salir del sistema.	2. El sistema muestra un mensaje de advertencia, solicitando que se confirme si realmente se desea salir del sistema.
3. El Administrador de SEGURINET confirma que desea abandonar el sistema.	4. El sistema se cierra.

Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**

Requerimientos especiales:

Caso de uso:	**Administrar Aplicación Web y/o módulo** **(**Sección: Principal**)**
Actores:	Administrador de SEGURINET.
Propósito:	Administrar (insertar, modificar o eliminar) información referente a las aplicaciones, módulos y/o submódulos permitiendo describir la estructura y composición de los elementos a los que SEGURINET brindará seguridad.
Resumen:	El Administrador de SEGURINET desea actualizar información referente a las aplicaciones, módulos y/o submódulos a los que SEGURINET brindará seguridad. Si desea insertar una nueva aplicación, módulo o submódulo, especifica sus datos; si desea modificar su información, elige el elemento correspondiente y cambia sus características y si desea borrarlo, lo selecciona y elimina.
Referencias:	R7
Precondiciones:	El Administrador de SEGURINET se encuentra previamente autenticado en el sistema.
Poscondiciones:	
Caso de uso asociado:	

Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El caso de uso comienza cuando el Administrador de SEGURINET habiendo seleccionado o no una aplicación, módulo o submódulo escoge luego una opción para actualizar información referente a ellos:	

a. Si selecciona la opción *Nuevo*, sin haber seleccionado ninguna aplicación, módulo o submódulo, véase sección *Insertar nueva aplicación.* b. Si habiendo seleccionado una aplicación selecciona luego la opción *Modificar*, véase sección *Modificar aplicación.* *c.* a. Si habiendo seleccionado una aplicación selecciona luego la opción *Eliminar*, véase sección *Eliminar aplicación.* d. Si habiendo seleccionado una aplicación selecciona luego la opción *Nuevo*, véase sección *Insertar nuevo módulo.* e. Si habiendo seleccionado un módulo selecciona luego la opción *Modificar*, véase sección *Modificar módulo.* f. Si habiendo seleccionado un módulo selecciona luego la opción *Eliminar*, véase sección *Eliminar módulo.* g. Si habiendo seleccionado un módulo o submódulo selecciona luego la opción *Nuevo*, véase sección *Insertar nuevo submódulo.* h. Si habiendo seleccionado un submódulo selecciona luego la opción *Modificar*, véase sección *Modificar sub módulo.* i. Si habiendo seleccionado un submódulo selecciona luego la opción *Eliminar*, véase sección *Eliminar submódulo.*	
Cursos alternativos	

Acción del actor:	**Respuesta del sistema:**
Requerimientos especiales:	

Caso de uso:	**Administrar Aplicación Web y/o módulo** **(**Sección: Insertar nueva aplicación**)**
Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
	1. El sistema solicita la entrada de: nombre de la aplicación, descripción y tipo de aplicación.
2. El Administrador de SEGURINET introduce: nombre de la aplicación, descripción y tipo de aplicación.	3. El sistema verifica que la aplicación especificada no exista dentro de sus módulos aplicaciones.
	4. El sistema inserta la nueva aplicación y muestra un mensaje que confirma el desarrollo exitoso de la operación especificada.
Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
	3. Si se verifica que la aplicación especificada existe en el sistema, este muestra un mensaje de error.
Requerimientos especiales:	

Caso de uso:	**Administrar Aplicación Web y/o módulo** **(**Sección: Modificar aplicación**)**
Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El Administrador de SEGURINET selecciona la aplicación que desea modificar.	2. El sistema muestra los actuales datos de la aplicación seleccionada: nombre de la aplicación, descripción, fecha de activada, estado.
3. El Administrador de SEGURINET modifica: nombre de la aplicación, descripción, fecha de activada y/o estado.	4. El sistema verifica, que si se modificó el nombre de la aplicación no coincida con ninguna de las existentes en el sistema.
	5. El sistema modifica la información de la aplicación.
Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
	4. Si se verifica que se modificó el nombre de la aplicación y ahora coincide con una existente en el sistema este muestra un mensaje de error.
Requerimientos especiales:	

Caso de uso:	**Administrar Aplicación Web y/o módulo** **(**Sección: Eliminar aplicación**)**
Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El Administrador de SEGURINET selecciona	2. El sistema muestra un mensaje de

una aplicación que desea eliminar.	advertencia solicitando que se confirme si se desea realmente eliminar la aplicación seleccionada.
3. El Administrador de SEGURINET confirma que desea eliminar la aplicación seleccionada.	4. El sistema elimina la información de la aplicación seleccionada.
Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
	2. Si el Administrador de SEGURINET no seleccionó la aplicación que desea eliminar, el sistema muestra un mensaje de error.
Requerimientos especiales:	

Caso de uso:	**Administrar Aplicación Web y/o módulo** (Sección: Insertar nuevo módulo)
Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
	1. El sistema solicita la entrada de: nombre del módulo, nombre de la aplicación padre y descripción.
2. El Administrador de SEGURINET introduce: nombre del módulo, nombre de la aplicación padre y descripción.	3. El sistema verifica que en la aplicación especificada no exista ya un módulo con el nombre del nuevo módulo a insertar.
	4. El sistema inserta el nuevo módulo y muestra un mensaje que confirma el desarrollo exitoso de la operación especificada.

Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
	3. Si se verifica que en la aplicación especificada ya existe un módulo con el nombre del nuevo módulo a insertar, el sistema muestra un mensaje de error.
Requerimientos especiales:	

Caso de uso:	**Administrar Aplicación Web y/o módulo** **(**Sección: Modificar módulo**)**
Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El Administrador de SEGURINET selecciona el módulo que desea modificar.	2. El sistema muestra los actuales datos del módulo seleccionado: nombre del módulo, descripción, estado.
3. El Administrador de SEGURINET modifica: nombre del módulo, descripción, estado.	4. El sistema verifica, que si se modificó el nombre del módulo no coincida con ningún otro existente en la aplicación en cuestión.
	5. El sistema modifica la información del módulo.
Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
	4. Si se verifica que se modificó el nombre del módulo y coincide con uno existente en la

	aplicación en cuestión, el sistema muestra un mensaje de error.
Requerimientos especiales:	

Caso de uso:	**Administrar Aplicación Web y/o módulo** **(**Sección: Eliminar módulo**)**
Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El Administrador de SEGURINET selecciona el módulo que desea eliminar.	2. El sistema muestra un mensaje de advertencia solicitando que se confirme si se desea realmente eliminar el módulo seleccionado.
3. El Administrador de SEGURINET confirma que desea realmente eliminar el módulo seleccionado.	4. El sistema elimina la información del módulo seleccionado.
Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
	2. Si el Administrador de SEGURINET no seleccionó el módulo que desea eliminar, el sistema muestra un mensaje de error.
Requerimientos especiales:	

Caso de uso:	**Administrar Aplicación Web y/o módulo** **(**Sección: Insertar nuevo submódulo**)**

Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
	1. El sistema solicita la entrada de: nombre del submódulo, nombre del módulo o submódulo padre y descripción.
2. El Administrador de SEGURINET introduce: nombre del submódulo, nombre del módulo o submódulo padre y descripción.	3. El sistema verifica que en el módulo o submódulo especificado no exista ya un submódulo con el nombre del nuevo submódulo a insertar.
	4. El sistema inserta el nuevo submódulo y muestra un mensaje que confirma el desarrollo exitoso de la operación especificada.
Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
	3. Si se verifica que en módulo o submódulo especificado ya existe un submódulo con el nombre del nuevo submódulo a insertar, el sistema muestra un mensaje de error.
Requerimientos especiales:	

Caso de uso:	**Administrar Aplicación Web y/o módulo** **(**Sección: Modificar submódulo**)**
Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**

1. El Administrador de SEGURINET selecciona el submódulo que desea modificar.	2. El sistema muestra los actuales datos del submódulo seleccionado: nombre del submódulo, descripción, estado.
3. El Administrador de SEGURINET modifica: nombre del submódulo, descripción, estado.	4. El sistema verifica, que si se modificó el nombre del submódulo no coincida con ningún otro existente en el módulo o submódulo en cuestión.
	5. El sistema modifica la información del submódulo.
Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
	4. Si se verifica que se modificó el nombre del submódulo y coincide con uno existente en el módulo o submódulo en cuestión, el sistema muestra un mensaje de error.
Requerimientos especiales:	

Caso de uso:	**Administrar Aplicación Web y/o módulo** **(**Sección: Eliminar submódulo**)**
Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El Administrador de SEGURINET selecciona el submódulo que desea eliminar.	2. El sistema muestra un mensaje de advertencia solicitando que se confirme si se desea realmente eliminar el submódulo seleccionado.
3. El Administrador de SEGURINET confirma	4. El sistema elimina la información del

que desea realmente eliminar el submódulo seleccionado.	submódulo seleccionado.
Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
	2. Si el Administrador de SEGURINET no seleccionó el módulo o submódulo que desea eliminar, el sistema muestra un mensaje de error.
Requerimientos especiales:	

Caso de uso:	**Visualizar reporte de módulos.**
Actores:	Administrador de SEGURINET.
Propósito:	Mostrar un listado de las aplicaciones, módulos y/o submódulos y sus respectivos datos que cumplan con las condiciones especificadas por el administrador de SEGURINET.
Resumen:	El Administrador de SEGURINET desea obtener un reporte de las aplicaciones, módulos y/o submódulos que cumplan con las restricciones que él mismo establece y el sistema usando este criterio los selecciona y muestra.
Referencias:	R8
Precondiciones:	El Administrador de SEGURINET se encuentra previamente autenticado en el sistema.
Poscondiciones:	
Caso de uso asociado:	
Curso normal de eventos	

Acción del actor:	**Respuesta del sistema:**
1. El caso de uso comienza cuando el Administrador de SEGURINET desea obtener un reporte determinado de las aplicaciones, módulos y/o submódulos de de las aplicaciones a las que SEGURINET brinda seguridad especificando un criterio de selección dado por un: nombre de aplicación, módulo o submódulo, descripción; aplicación, módulo o submódulo padre; estado, fecha de creado y/o fecha de activado.	2. El sistema muestra el listado de aplicaciones, módulos y/o submódulos correspondiente.
3. El Administrador de SEGURINET solicita la elaboración del reporte a partir del listado obtenido.	4. El sistema elabora el reporte como un documento, listo para imprimirse, con un encabezado que lo describe.
Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
	2. Si el Administrador de SEGURINET no especificó un criterio de selección el sistema muestra un listado completo con todas las aplicaciones, módulos y submódulos que administra SEGURINET.
Requerimientos especiales:	

Caso de uso:	**Administrar formulario** (Sección: Principal)
Actores:	Administrador de SEGURINET.
Propósito:	Administrar (insertar, modificar o eliminar) información referente a los

	formulario de las aplicaciones a las que SEGURINET brinda seguridad.
Resumen:	El Administrador de SEGURINET desea actualizar información referente a los formulario de las aplicaciones a las que SEGURINET brinda seguridad, si desea insertar un formulario nuevo especifica sus datos, si desea cambiar los datos de alguno los modifica y si un formulario deja de existir lo elimina.
Referencias:	R9
Precondiciones:	El Administrador de SEGURINET se encuentra previamente autenticado en el sistema.
Poscondiciones:	
Caso de uso asociado:	

Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El caso de uso comienza cuando el Administrador de SEGURINET selecciona una opción para actualizar información referente a los formulario: a. Si selecciona *Nuevo*, véase sección *Insertar nuevo formulario*. b. Si selecciona *Modificar*, véase sección *Modificar formulario.* c. Si selecciona *Eliminar*, véase sección *Eliminar formulario*.	

Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**

Requerimientos especiales:

Caso de uso:	**Administrar formulario** **(**Sección: Insertar nuevo formulario**)**
Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El Administrador de SEGURINET selecciona la aplicación, el módulo o submódulo al que desea insertar un formulario.	2. El sistema solicita la entrada de: nombre de formulario, módulo o submódulo al que pertenecerá y descripción.
3. El Administrador de SEGURINET introduce: nombre de formulario, módulo o submódulo al que pertenecerá y descripción.	4. El sistema verifica que el formulario especificado no exista ya en la aplicación, el módulo o submódulo seleccionado.
	5. El sistema inserta el nuevo formulario y muestra un mensaje que confirma el desarrollo exitoso de la operación especificada.
Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
	4. Si se verifica que el formulario especificado existe en la aplicación, el módulo o submódulo seleccionado, el sistema muestra un mensaje de error.
Requerimientos especiales:	

Caso de uso:	**Administrar formulario** **(**Sección: Modificar formulario**)**
Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El Administrador de SEGURINET selecciona el formulario que desea modificar.	2. El sistema muestra los actuales datos del formulario seleccionado: nombre de formulario, descripción y estado.
3. El Administrador de SEGURINET modifica: nombre de formulario, descripción y/o estado.	4. El sistema verifica, que si se modificó el nombre del formulario no coincida con ninguno existente en la aplicación, el módulo o submódulo seleccionado.
	5. El sistema modifica la información del formulario.
Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
	2. Si el Administrador de SEGURINET no seleccionó el formulario que desea modificar el sistema muestra un mensaje de error.
	4. Si se verifica que se modificó el nombre de formulario y ahora coincide con uno existente en la aplicación, el módulo o submódulo seleccionado, el sistema muestra un mensaje de error.
Requerimientos especiales:	

Caso de uso:	**Administrar formulario** **(**Sección: Eliminar formulario**)**
Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El Administrador de SEGURINET selecciona el formulario que desea eliminar.	2. El sistema muestra un mensaje de advertencia solicitando que se confirme si se desea realmente eliminar el formulario seleccionado.
3. El Administrador de SEGURINET confirma que desea realmente eliminar el formulario seleccionado.	4. El sistema elimina la información del formulario seleccionado.
Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
	2. Si el Administrador de SEGURINET no seleccionó el formulario que desea eliminar el sistema muestra un mensaje de error.
Requerimientos especiales:	

Caso de uso:	**Visualizar reporte de formularios.**
Actores:	Administrador de SEGURINET.
Propósito:	Mostrar un listado de los formularios y sus respectivos datos que cumplan con las condiciones especificadas por el administrador de SEGURINET.
Resumen:	El Administrador de SEGURINET desea obtener un reporte de los formularios que cumplan con las restricciones que él mismo establece

	y el sistema usando este criterio los selecciona y muestra.
Referencias:	R10
Precondiciones:	El Administrador de SEGURINET se encuentra previamente autenticado en el sistema.
Poscondiciones:	
Caso de uso asociado:	

Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El caso de uso comienza cuando el Administrador de SEGURINET desea obtener un reporte determinado de los formularios de las aplicaciones a las que SEGURINET brinda seguridad especificando un criterio de selección dado por un: nombre de formulario; aplicación, módulo o submódulo al que pertenece, descripción, estado, fecha de creado y/o fecha de activado.	2. El sistema muestra el listado de formularios correspondiente.
3. El Administrador de SEGURINET solicita la elaboración del reporte a partir del listado obtenido.	4. El sistema elabora el reporte como un documento, listo para imprimirse, con un encabezado que lo describe.

Cursos alternativos:	
Acción del actor:	**Respuesta del sistema:**
	2. Si el Administrador de SEGURINET no especificó un criterio de selección el sistema muestra un listado completo con todos los formularios que administra SEGURINET.

Requerimientos especiales:

Caso de uso:	**Administrar rol** (Sección: Principal)
Actores:	Administrador de SEGURINET.
Propósito:	Administrar (insertar, modificar o eliminar) información referente a los roles de las aplicaciones a las que SEGURINET brinda seguridad.
Resumen:	El Administrador de SEGURINET desea actualizar información referente a los roles de las aplicaciones a las que SEGURINET brinda seguridad, si desea insertar un rol nuevo especifica sus datos, si desea cambiar los datos de alguno los modifica y si un rol deja de existir lo elimina.
Referencias:	R11
Precondiciones:	El Administrador de SEGURINET se encuentra previamente autenticado en el sistema.
Poscondiciones:	
Caso de uso asociado:	

Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El caso de uso comienza cuando el Administrador de SEGURINET selecciona una opción para actualizar información referente a los roles: a. Si selecciona *Nuevo*, véase sección *Insertar*	

nuevo rol. b. Si selecciona *Modificar*, véase sección *Modificar rol.* c. Si selecciona *Eliminar*, véase sección *Eliminar rol*.	
Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
Requerimientos especiales:	

Caso de uso:	**Administrar rol** **(**Sección: Insertar nuevo rol**)**
Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
	1. El sistema solicita la entrada de: nombre del rol y descripción.
2. El Administrador de SEGURINET introduce: nombre del rol y descripción.	
3. El Administrador de SEGURINET selecciona la aplicación de la que formará parte el nuevo rol.	
4. El Administrador de SEGURINET especifica a cuales formularios de la aplicación seleccionada tendrá acceso el nuevo rol.	

	5. El sistema verifica que el rol especificado no exista ya en la aplicación seleccionada.
	6. El sistema inserta el nuevo rol y muestra un mensaje que confirma el desarrollo exitoso de la operación especificada.
Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
	5. Si el administrador de SEGURINET no seleccionó una aplicación o se verifica que el rol especificado existe en la aplicación seleccionada, el sistema muestra un mensaje de error.
Requerimientos especiales:	

Caso de uso:	**Administrar rol** **(**Sección: Modificar rol**)**
Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El Administrador de SEGURINET selecciona el rol que desea modificar.	2. El sistema muestra los actuales datos del rol seleccionado: nombre del rol, descripción, estado, aplicación a la que pertenece y los formularios de la misma a los que tiene acceso.
3. El Administrador de SEGURINET modifica: nombre del rol, descripción, estado, aplicación a la que pertenece y/o los formularios de la misma	4. El sistema verifica, que si se modificó el nombre del rol no coincida con ninguno existente

a los que tiene acceso.	en la aplicación seleccionada.
	5. El sistema modifica la información del rol.
Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
	2. Si el Administrador de SEGURINET no seleccionó el rol que desea modificar el sistema muestra un mensaje de error.
	4. Si se verifica que se modificó el nombre del rol y ahora coincide con uno existente en la aplicación seleccionada, el sistema muestra un mensaje de error.
Requerimientos especiales:	

Caso de uso:	**Administrar rol** **(**Sección: Eliminar rol**)**
Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El Administrador de SEGURINET selecciona el rol que desea eliminar.	2. El sistema muestra un mensaje de advertencia solicitando que se confirme si se desea realmente eliminar el rol seleccionado.
3. El Administrador de SEGURINET confirma que desea realmente eliminar el rol seleccionado.	4. El sistema elimina la información del rol seleccionado.
Cursos alternativos	

Acción del actor:	**Respuesta del sistema:**
	2. Si el Administrador de SEGURINET no seleccionó el rol que desea eliminar el sistema muestra un mensaje de error.
Requerimientos especiales:	

Caso de uso:	**Visualizar reporte de roles.**
Actores:	Administrador de SEGURINET.
Propósito:	Mostrar un listado de los roles y sus respectivos datos que cumplan con las condiciones especificadas por el administrador de SEGURINET.
Resumen:	El Administrador de SEGURINET desea obtener un reporte de los roles que cumplan con las restricciones que él mismo establece y el sistema usando este criterio los selecciona y muestra.
Referencias:	R12
Precondiciones:	El Administrador de SEGURINET se encuentra previamente autenticado en el sistema.
Poscondiciones:	
Caso de uso asociado:	
Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El caso de uso comienza cuando el Administrador de SEGURINET desea obtener	2. El sistema muestra el listado de roles correspondiente.

un reporte determinado de los roles de las aplicaciones a las que SEGURINET brinda seguridad especificando un criterio de selección dado por un: nombre de rol; aplicación a la que pertenece, descripción, estado, fecha de creado y/o fecha de activado, módulo, submódulo o formulario al que tiene acceso.	
3. El Administrador de SEGURINET solicita la elaboración del reporte a partir del listado obtenido.	4. El sistema elabora el reporte como un documento, listo para imprimirse, con un encabezado que lo describe.
Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
	2. Si el Administrador de SEGURINET no especificó un criterio de selección el sistema muestra un listado completo con todos los roles que administra SEGURINET.
Requerimientos especiales:	

Caso de uso:	**Visualizar reporte de accesos.**
Actores:	Administrador de SEGURINET.
Propósito:	Mostrar un listado de los accesos y sus respectivos datos que cumplan con las condiciones especificadas por el administrador de SEGURINET.
Resumen:	El Administrador de SEGURINET desea obtener un reporte de los accesos que cumplan con las restricciones que él mismo establece y el sistema usando este criterio los selecciona y muestra.
Referencias:	R13

Precondiciones:	El Administrador de SEGURINET se encuentra previamente autenticado en el sistema.
Poscondiciones:	
Caso de uso asociado:	

Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El caso de uso comienza cuando el Administrador de SEGURINET desea obtener un reporte determinado de los accesos a las diferentes aplicaciones a las que SEGURINET brinda seguridad especificando un criterio de selección dado por un: nombre de usuario; nombre de aplicación, nombre de módulo, nombre de formulario, operación, descripción, estado, y/o fecha del acceso.	2. El sistema muestra el listado de accesos correspondiente.
3. El Administrador de SEGURINET solicita la elaboración del reporte a partir del listado obtenido.	4. El sistema elabora el reporte como un documento, listo para imprimirse, con un encabezado que lo describe.

Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
	2. Si el Administrador de SEGURINET no especificó un criterio de selección el sistema muestra un listado completo con todos los accesos a las aplicaciones que administra SEGURINET.

Requerimientos especiales:

Caso de uso:	**Administrar operación** (Sección: Principal)
Actores:	Administrador de SEGURINET.
Propósito:	Administrar (insertar, modificar o eliminar) información referente a las operaciones que se pueden hacer en las aplicaciones a las que SEGURINET brinda seguridad.
Resumen:	El Administrador de SEGURINET desea actualizar información referente a las operaciones que se pueden hacer en las aplicaciones a las que SEGURINET brinda seguridad, si desea insertar una operación nueva, especifica sus datos, si desea cambiar los datos de alguna los modifica y si operación deja de existir la elimina.
Referencias:	R14
Precondiciones:	El Administrador de SEGURINET se encuentra previamente autenticado en el sistema.
Poscondiciones:	
Caso de uso asociado:	

Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El caso de uso comienza cuando el Administrador de SEGURINET selecciona una opción para actualizar información referente a las operaciones: a. Si selecciona *Nuevo*, véase sección *Insertar nueva operación*. b. Si selecciona *Modificar*, véase sección *Modificar operación.*	

c. Si selecciona *Eliminar*, véase sección *Eliminar operación*.	
Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
Requerimientos especiales:	

Caso de uso:	**Administrar operación** **(**Sección: Insertar nueva operación**)**
Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
	1. El sistema solicita la entrada del nombre de la operación.
2. El Administrador de SEGURINET introduce: nombre de la operación.	3. El sistema verifica que la operación especificada no exista ya en el sistema.
	4. El sistema inserta la nueva operación y muestra un mensaje que confirma el desarrollo exitoso de la operación especificada.
Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
	3. Si se verifica que la operación especificada existe en el sistema, este muestra un mensaje de error.
Requerimientos especiales:	

Caso de uso:	**Administrar operación** **(**Sección: Modificar operación**)**
Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El Administrador de SEGURINET selecciona la operación que desea modificar.	2. El sistema muestra los actuales datos la operación seleccionada: nombre de la operación.
3. El Administrador de SEGURINET modifica: nombre de operación.	4. El sistema verifica que al modificar el nombre de la operación no coincida con ninguna existente ya en él.
	5. El sistema modifica la información de la operación.
Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
	2. Si el Administrador de SEGURINET no seleccionó el rol que desea modificar el sistema muestra un mensaje de error.
	4. Si el sistema verifica que al modificar el nombre de la operación coincida con alguna existente ya en él, el sistema muestra un mensaje de error.
Requerimientos especiales:	

Caso de uso:	**Administrar operación** **(**Sección: Eliminar operación**)**

Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El Administrador de SEGURINET selecciona la operación que desea eliminar.	2. El sistema muestra un mensaje de advertencia solicitando que se confirme si se desea realmente eliminar la operación seleccionada.
3. El Administrador de SEGURINET confirma que desea realmente eliminar la operación seleccionada.	4. El sistema elimina la información la operación seleccionada.
Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
	2. Si el Administrador de SEGURINET no seleccionó la operación que desea eliminar, el sistema muestra un mensaje de error.
Requerimientos especiales:	

Caso de uso:	**Administrar históricos** (Sección: Principal)
Actores:	Administrador de SEGURINET.
Propósito:	Recuperar o eliminar datos que han sido borrados pero se mantienen aún en el sistema como una información histórica.
Resumen:	El Administrador de SEGURINET desea recuperar información que el sistema manipulaba en algún momento anterior y fue eliminada por él o quiere eliminar definitivamente del sistema esta información que se encuentra en estado de borrada.

Referencias:	R15
Precondiciones:	El Administrador de SEGURINET se encuentra previamente autenticado en el sistema.
Poscondiciones:	
Caso de uso asociado:	

Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El caso de uso comienza cuando el Administrador de SEGURINET selecciona que tipo de elemento de la información histórica desea manipular: módulo, formulario, usuario, rol.	
2. El Administrador de SEGURINET selecciona una opción para manipular información histórica: a. Si selecciona *Recuperar*, véase sección *Recuperar elemento del histórico*. b. Si selecciona *Eliminar*, véase sección *Eliminar elemento del histórico.*	

Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**

Requerimientos especiales:

Caso de uso:	**Administrar históricos** **(**Sección: Recuperar elemento del histórico**)**
Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
	1. El sistema recupera el elemento indicado de los datos históricos.
Cursos alternativos:	
Acción del actor:	**Respuesta del sistema:**
Requerimientos especiales:	

Caso de uso:	**Administrar históricos** **(**Sección: Eliminar elemento del histórico**)**
Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
	1. El sistema muestra un mensaje de advertencia solicitando que se confirme si se desea realmente eliminar el elemento histórico seleccionado.
2. El Administrador de SEGURINET confirma que desea realmente eliminar el elemento histórico seleccionado.	3. El sistema elimina el elemento indicado de los datos históricos y con esto se borra definitivamente del sistema.
Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**

Requerimientos especiales:

Caso de uso:	**Visualizar reporte de históricos** (Sección: Principal)
Actores:	Administrador de SEGURINET.
Propósito:	Mostrar un listado de los datos históricos (usuarios, módulos, formularios, roles) y sus respectivos datos que cumplan con las condiciones especificadas por el administrador de SEGURINET.
Resumen:	El Administrador de SEGURINET especifica el elemento (usuarios, módulos, formularios, roles) sobre el que desea obtener un reporte de los datos históricos que cumplan con las restricciones que él mismo establece y el sistema usando este criterio los selecciona y muestra.
Referencias:	R16
Precondiciones:	El Administrador de SEGURINET se encuentra previamente autenticado en el sistema.
Poscondiciones:	
Caso de uso asociado:	

Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El caso de uso comienza cuando el Administrador de SEGURINET selecciona el elemento (usuarios, módulos, formularios, roles) sobre el que desea obtener un reporte de los históricos correspondientes: a. Si selecciona *Usuario*, véase sección	

Históricos de usuario. b. Si selecciona *Módulo*, véase sección *Históricos de módulo*. c. Si selecciona *Formulario*, véase sección *Históricos de formulario*. d. Si selecciona *Rol*, véase sección *Históricos de rol*.	
Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
Requerimientos especiales:	

Caso de uso: **Visualizar reporte de históricos** (Sección: Históricos de usuario)		
3. El Administrador de SEGURINET solicita la elaboración del reporte a partir del listado obtenido.		4. El sistema elabora el reporte como un documento, listo para imprimirse, con un encabezado que lo describe.
Curso normal de eventos		
Cursos alternativos		
Acción del actor:		**Respuesta del sistema:**
Acción del actor:		**Respuesta del sistema:**
1. El Administrador de SEGURINET desea obtener un reporte determinado de los datos históricos de los usuarios de las diferentes aplicaciones a las que SEGURINET brinda seguridad especificando un criterio de selección dado por un: nombre de usuario; nombre completo del usuario, descripción, estado, período de cambio de contraseña, fecha de creado, fecha de activado, fecha de eliminado.		2. El sistema muestra el listado de los datos históricos de los usuarios que corresponde. 2. Si el Administrador de SEGURINET no especificó un criterio de selección el sistema muestra un listado completo con todos los datos históricos de los usuarios de las aplicaciones que administra SEGURINET.

Requerimientos especiales:

Caso de uso:	**Visualizar reporte de históricos** (Sección: Históricos de módulo)
Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El Administrador de SEGURINET desea obtener un reporte determinado de los datos históricos de los módulos a los que SEGURINET brinda seguridad especificando un criterio de selección dado por un: nombre de módulo; módulo padre, descripción, estado, fecha de creado, fecha de activado, fecha de eliminado.	2. El sistema muestra el listado de los datos históricos de los módulos que corresponde.
3. El Administrador de SEGURINET solicita la elaboración del reporte a partir del listado obtenido.	4. El sistema elabora el reporte como un documento, listo para imprimirse, con un encabezado que lo describe.
Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
	2. Si el Administrador de SEGURINET no especificó un criterio de selección el sistema muestra un listado completo con todos los datos históricos de los módulos de las aplicaciones que administra SEGURINET.

Requerimientos especiales:

Caso de uso:	**Visualizar reporte de históricos** (Sección: Históricos de formulario)
Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El Administrador de SEGURINET desea obtener un reporte determinado de los datos históricos de los formularios a los que SEGURINET brinda seguridad especificando un criterio de selección dado por un: nombre de formulario; descripción, estado, fecha de creado, fecha de activado, fecha de eliminado.	2. El sistema muestra el listado de los datos históricos de los formularios que corresponde.
3. El Administrador de SEGURINET solicita la elaboración del reporte a partir del listado obtenido.	4. El sistema elabora el reporte como un documento, listo para imprimirse, con un encabezado que lo describe.
Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
	2. Si el Administrador de SEGURINET no especificó un criterio de selección el sistema muestra un listado completo con todos los datos históricos de los formularios de las aplicaciones que administra SEGURINET.
Requerimientos especiales:	

Caso de uso:	**Visualizar reporte de históricos** (Sección: Históricos de rol)
Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El Administrador de SEGURINET desea obtener un reporte determinado de los datos históricos de los roles correspondientes a una de las aplicaciones a las que SEGURINET brinda seguridad especificando un criterio de selección dado por un: nombre del rol; Aplicación a la que pertenece, descripción, estado, fecha de creado, fecha de activado, fecha de eliminado.	2. El sistema muestra el listado de los datos históricos de los roles que corresponde.
3. El Administrador de SEGURINET solicita la elaboración del reporte a partir del listado obtenido.	4. El sistema elabora el reporte como un documento, listo para imprimirse, con un encabezado que lo describe.
Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
	2. Si el Administrador de SEGURINET no especificó un criterio de selección el sistema muestra un listado completo con todos los roles de las aplicaciones que administra SEGURINET.
Requerimientos especiales:	

Caso de uso:	**Configurar sistema.**

Actores:	Administrador de SEGURINET.
Propósito:	Establecer algunos parámetros de configuración del sistema SEGURINET como el tiempo que permanecerán los datos eliminados en los históricos hasta su posterior borrado definitivo y el período de vigencia de la contraseña del administrador.
Resumen:	El Administrador de SEGURINET configura el sistema a su conveniencia especificando el tiempo que permanecerán los datos eliminados en los históricos hasta su posterior borrado definitivo y el período de vigencia de su contraseña.
Referencias:	R17
Precondiciones:	El Administrador de SEGURINET se encuentra previamente autenticado en el sistema.
Poscondiciones:	
Caso de uso asociado:	

Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El caso de uso comienza cuando el Administrador de SEGURINET desea configurar el sistema.	2. El sistema solicita: el tiempo que permanecerán los datos eliminados en los históricos hasta su posterior borrado definitivo y el período de vigencia de la contraseña del Administrador de SEGURINET.
3. El Administrador de SEGURINET especifica: el tiempo que permanecerán los datos eliminados en los históricos y el período de vigencia de su contraseña.	4. El sistema establece la configuración establecida.

Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
Requerimientos especiales:	

Caso de uso:	**Servicio controlar la autenticación de usuario.**
Actores:	Aplicación Web.
Propósito:	Reconocer la identidad de un usuario de alguna aplicación de las que SEGURINET les brinda seguridad.
Resumen:	Una aplicación de las que SEGURINET les brinda seguridad identifica ante el sistema un usuario, él cual valida sus datos.
Referencias:	R18
Precondiciones:	
Poscondiciones:	
Caso de uso asociado:	
Curso normal de eventos	
Acción del actor:	**Respuesta del sistema:**
1. El caso de uso comienza cuando una aplicación Web solicita a SEGURINET la autenticación de un usuario facilitando: nombre de usuario, contraseña, aplicación y formulario a la que desea acceder.	2. El sistema valida los datos.
	3. El sistema le informa a la aplicación Web la

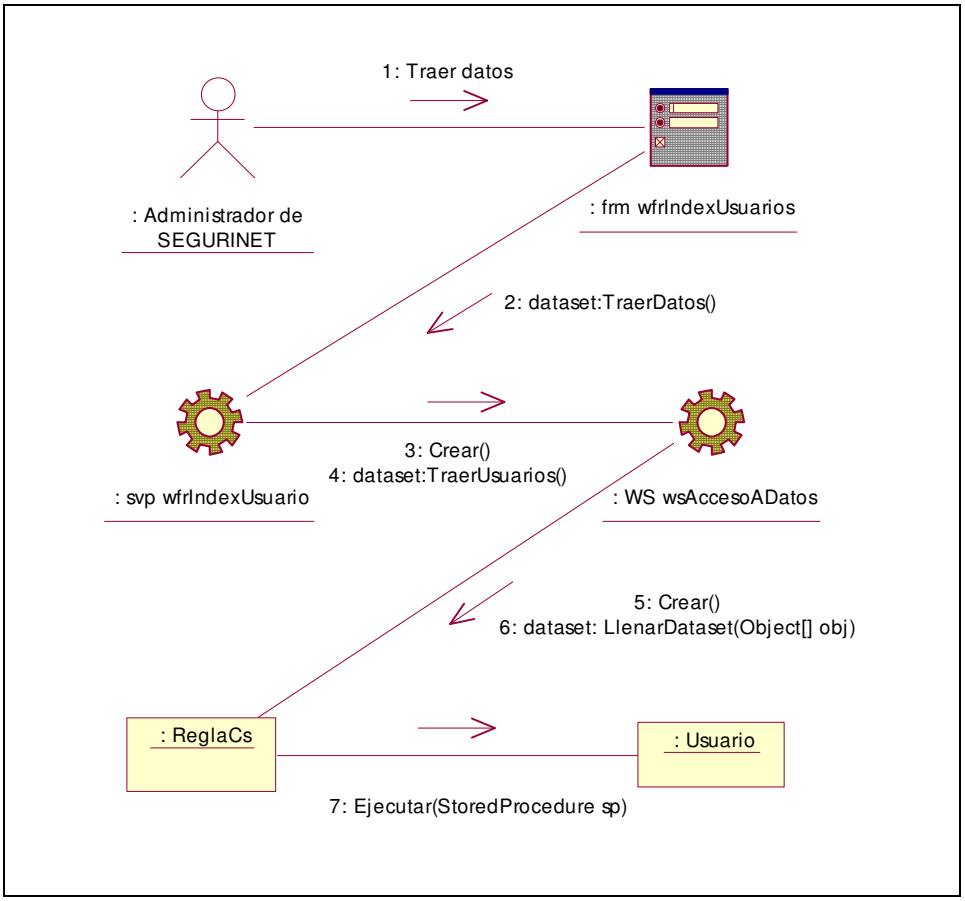
	autorización de acceso para el usuario dado.
Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
	2. Si al validar los datos el sistema detecta que la aplicación, el usuario o el formulario no existen, si la contraseña está mal o el usuario no tiene permiso de acceso al formulario especificado entonces el sistema informa a la aplicación Web la denegación del acceso para el usuario dado.
Requerimientos especiales:	

Caso de uso:	**Servicio controlar vigencia de contraseña**
Actores:	Aplicación Web.
Propósito:	Verificar si el período de vigencia de la contraseña de un usuario de alguna aplicación de las que SEGURINET les brinda seguridad eximió.
Resumen:	Una aplicación de las que SEGURINET les brinda seguridad se solicita verificar si se ha eximido o no la contraseña de uno de sus usuarios, el sistema chequea los datos suministrados y responde a la petición.
Referencias:	R19
Precondiciones:	
Poscondiciones:	
Caso de uso asociado:	
Curso normal de eventos	

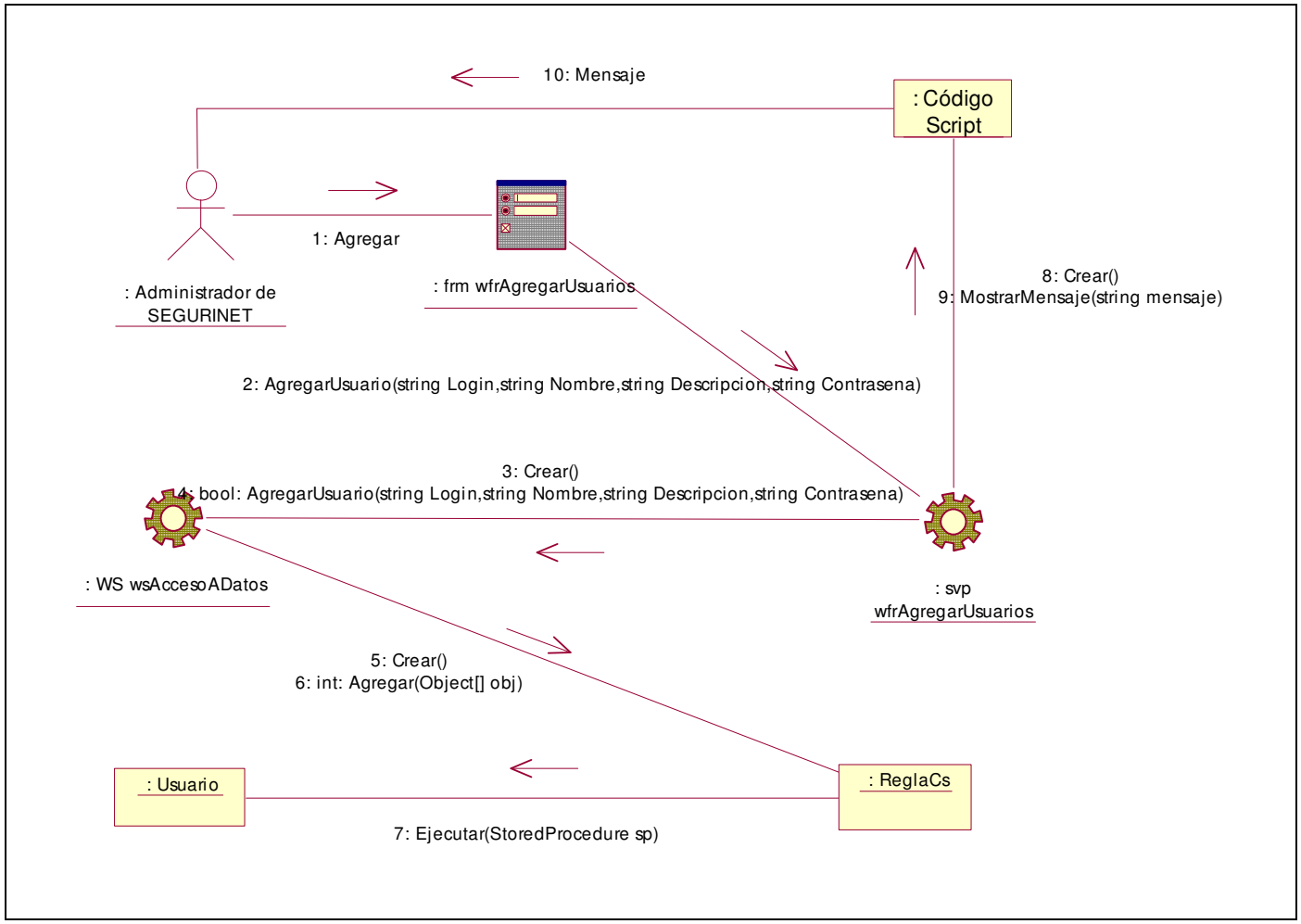
Acción del actor:	**Respuesta del sistema:**
1. El caso de uso comienza cuando una aplicación Web solicita a SEGURINET la verificación de la vigencia de la contraseña de un de sus usuarios facilitando: nombre de usuario y contraseña.	2. El sistema valida los datos.
	3. El sistema le informa a la aplicación Web si se ha consumido o no el periodo de vigencia de la contraseña del usuario dado.
Cursos alternativos	
Acción del actor:	**Respuesta del sistema:**
	2. Si al validar los datos el sistema detecta que el usuario no existen o la contraseña está mal, el sistema informa a la aplicación un error.
Requerimientos especiales:	

Anexo 2 Diagramas de colaboración

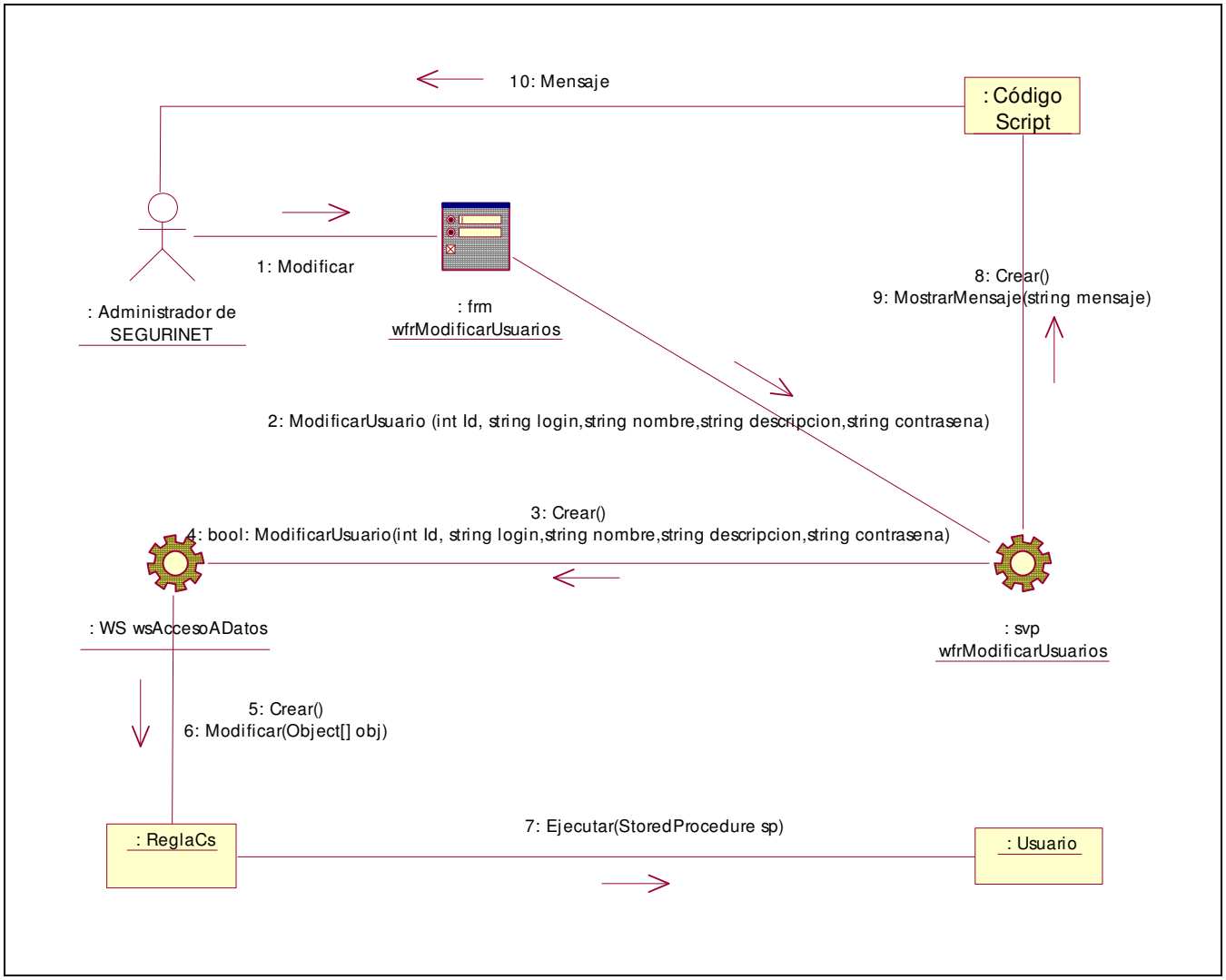
CU Administrar Usuarios. Escenario Principal



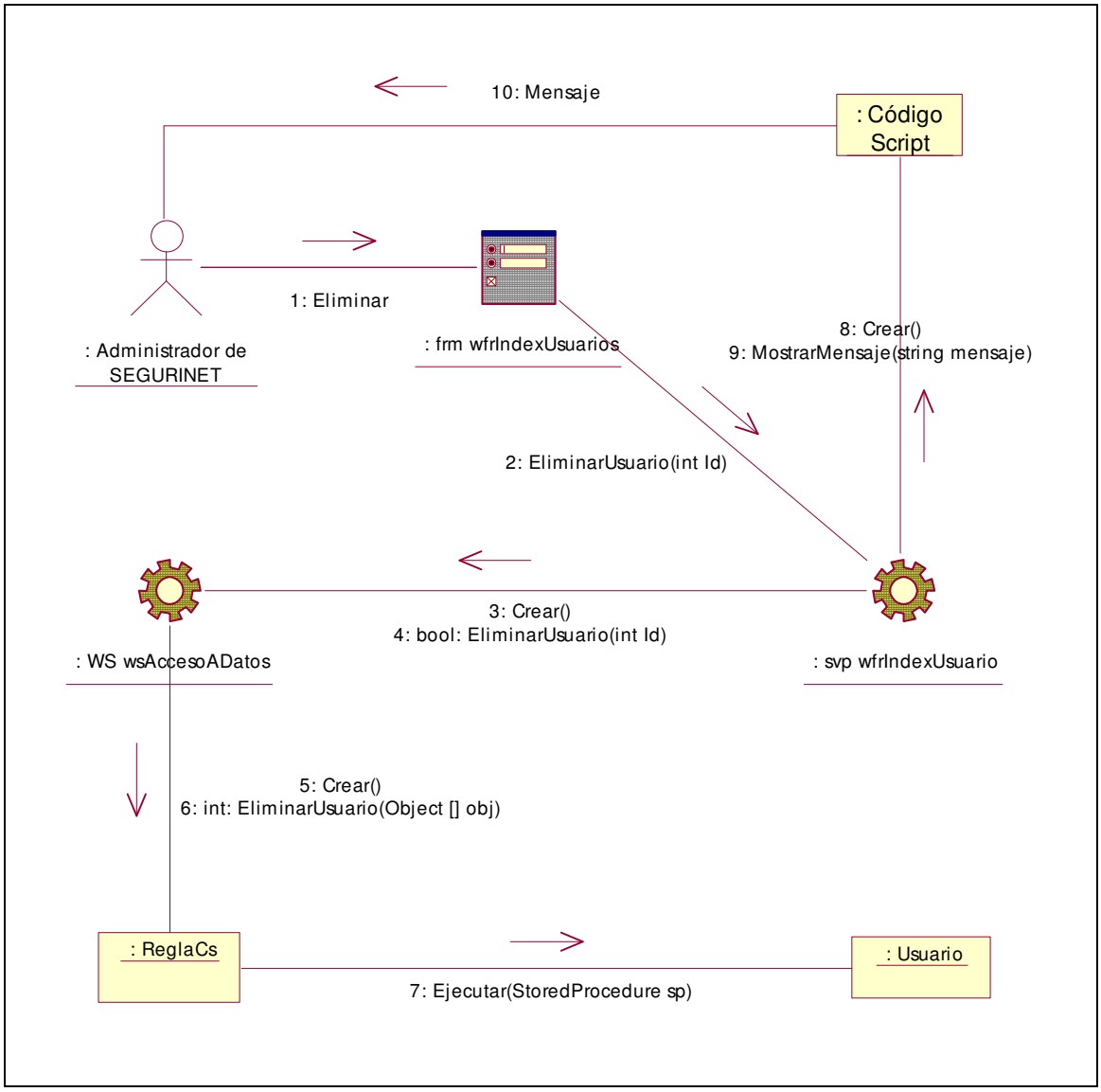
CU Administrar Usuarios. Escenario Insertar Usuario



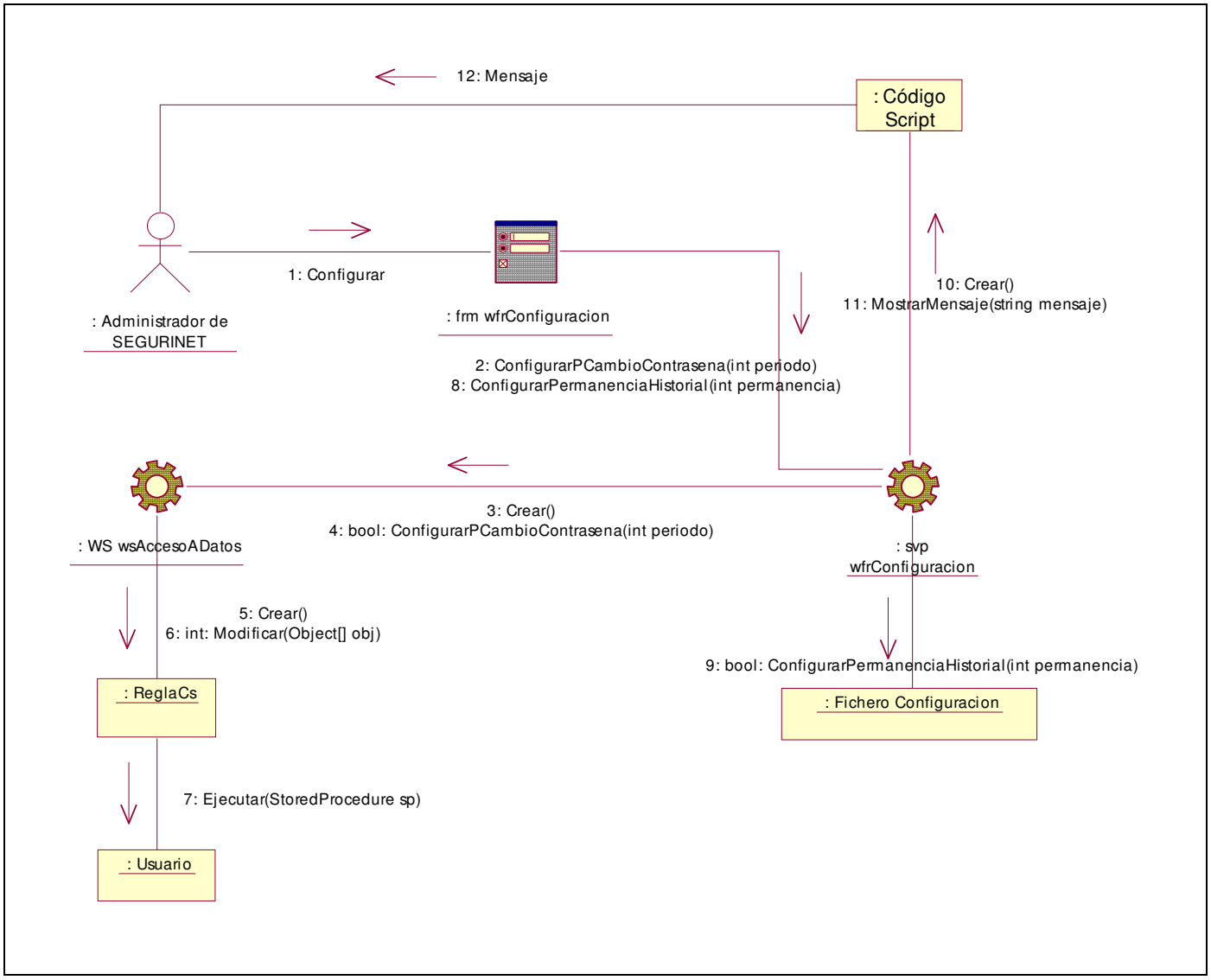
CU Administrar Usuarios. Escenario Modificar Usuario



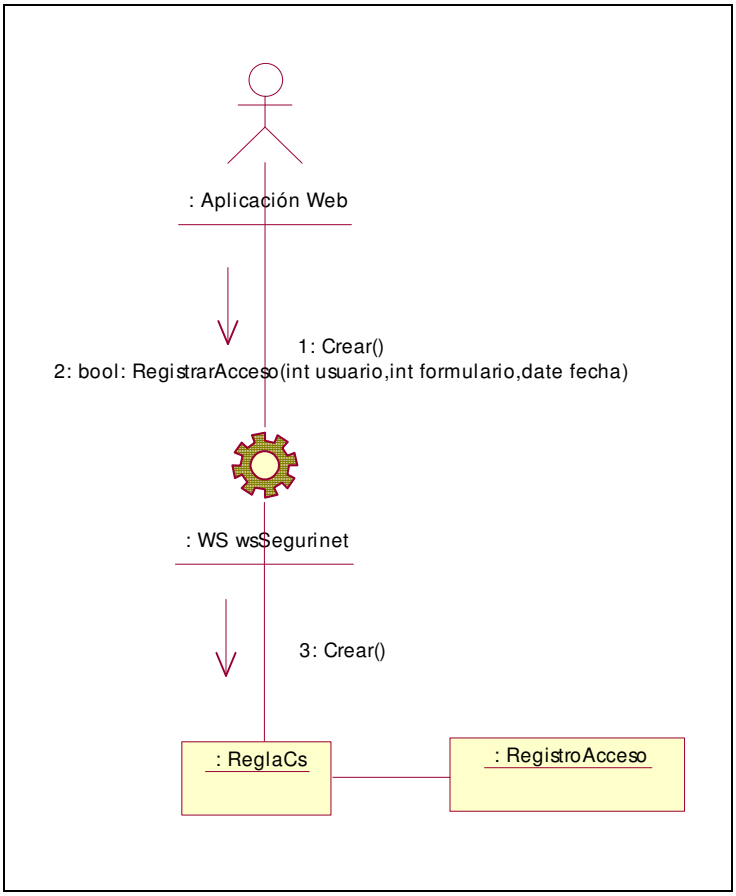
CU Administrar Usuarios. Escenario Eliminar Usuario



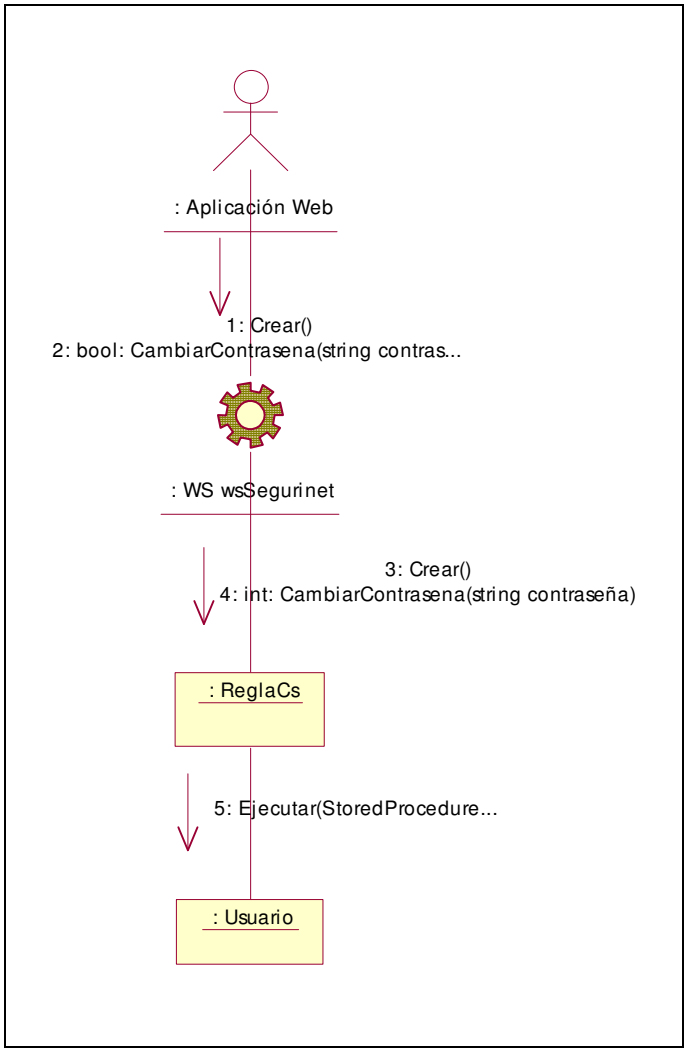
CU Configurar Sistema



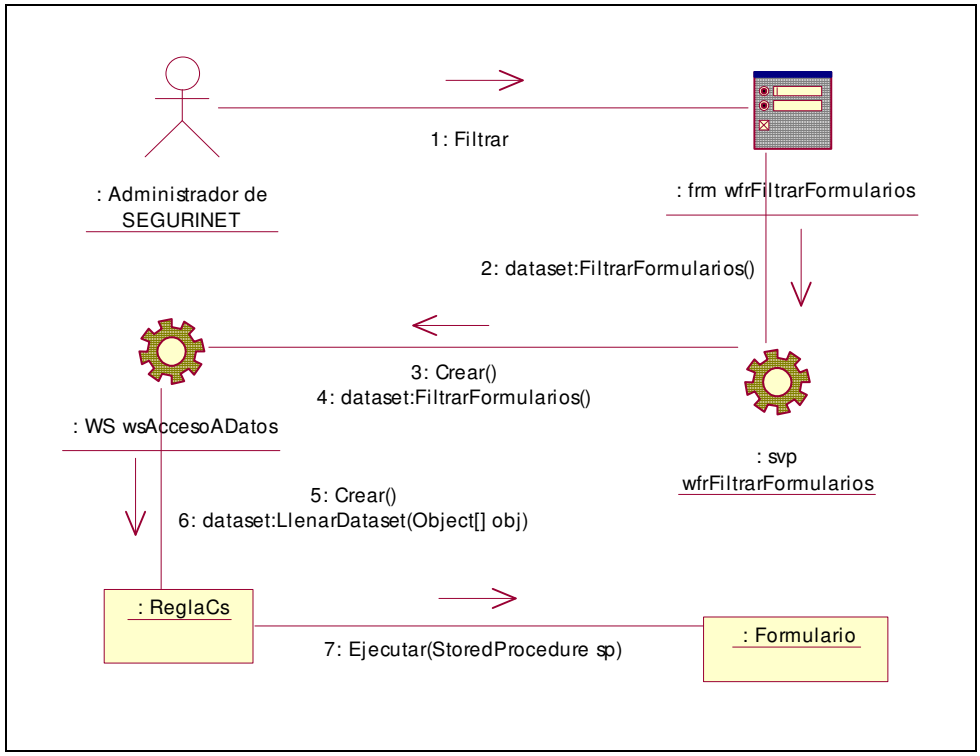
CU Servicio Registrar Acceso



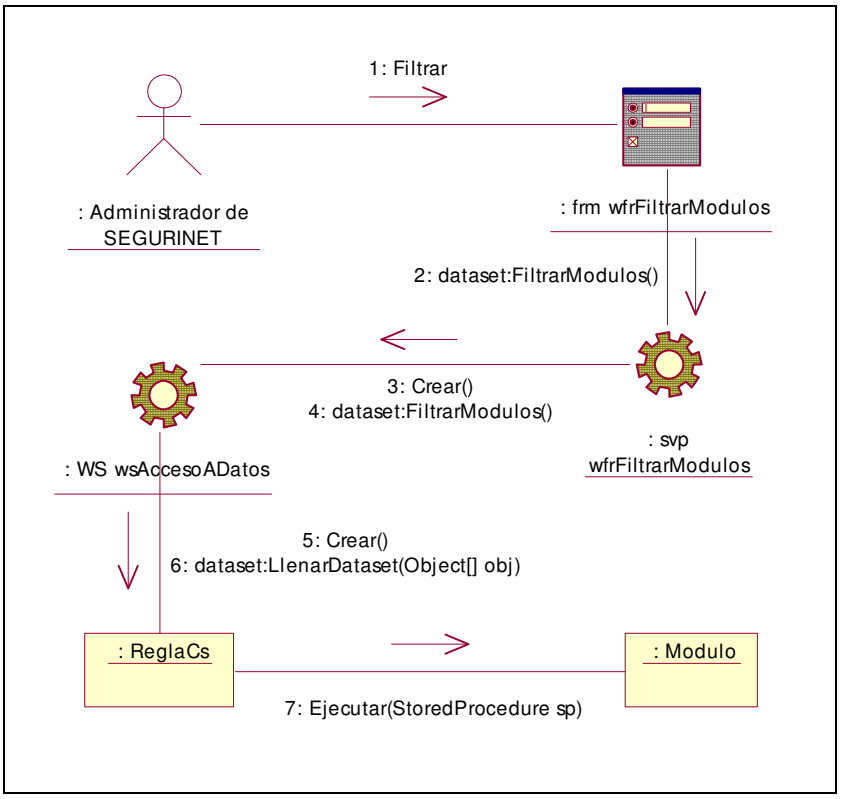
CU Servicio Cambiar contraseña



CU Visualizar Reporte de Formularios



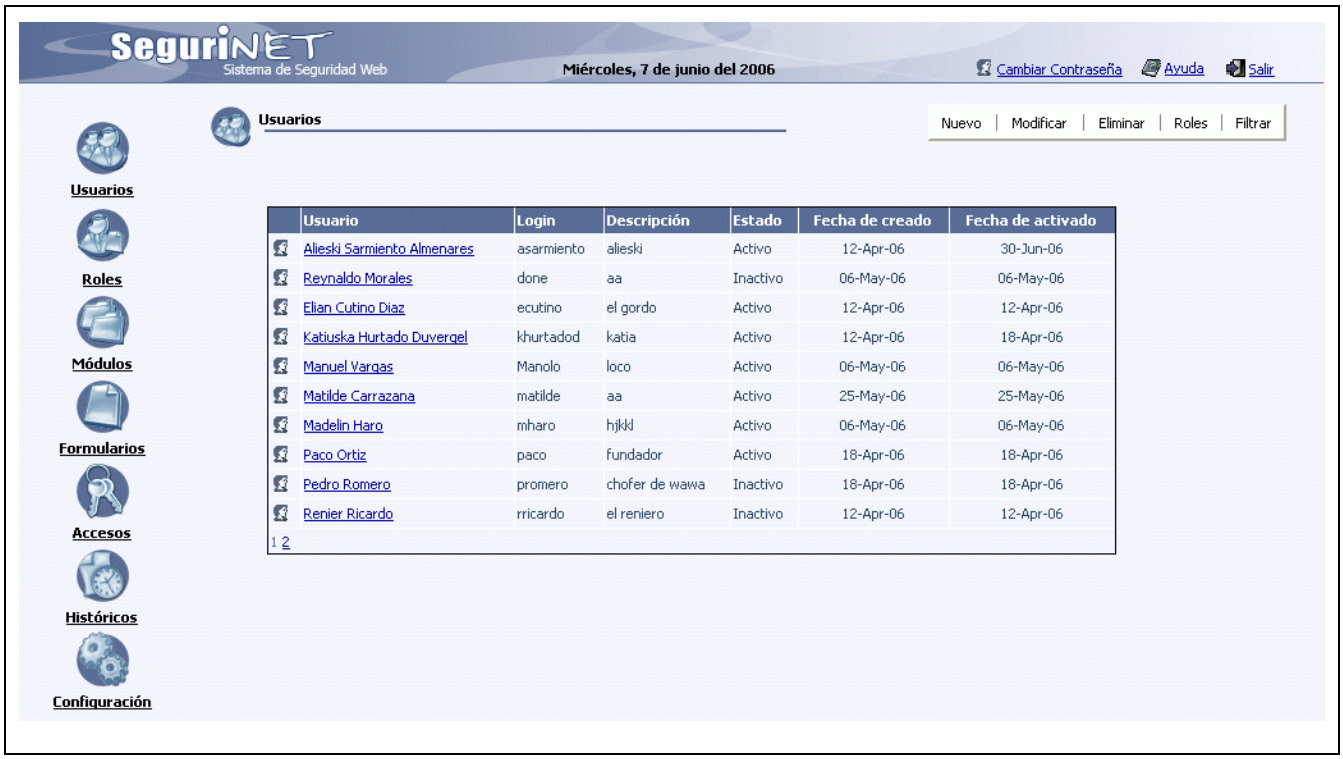
CU Visualizar Reporte de Módulos



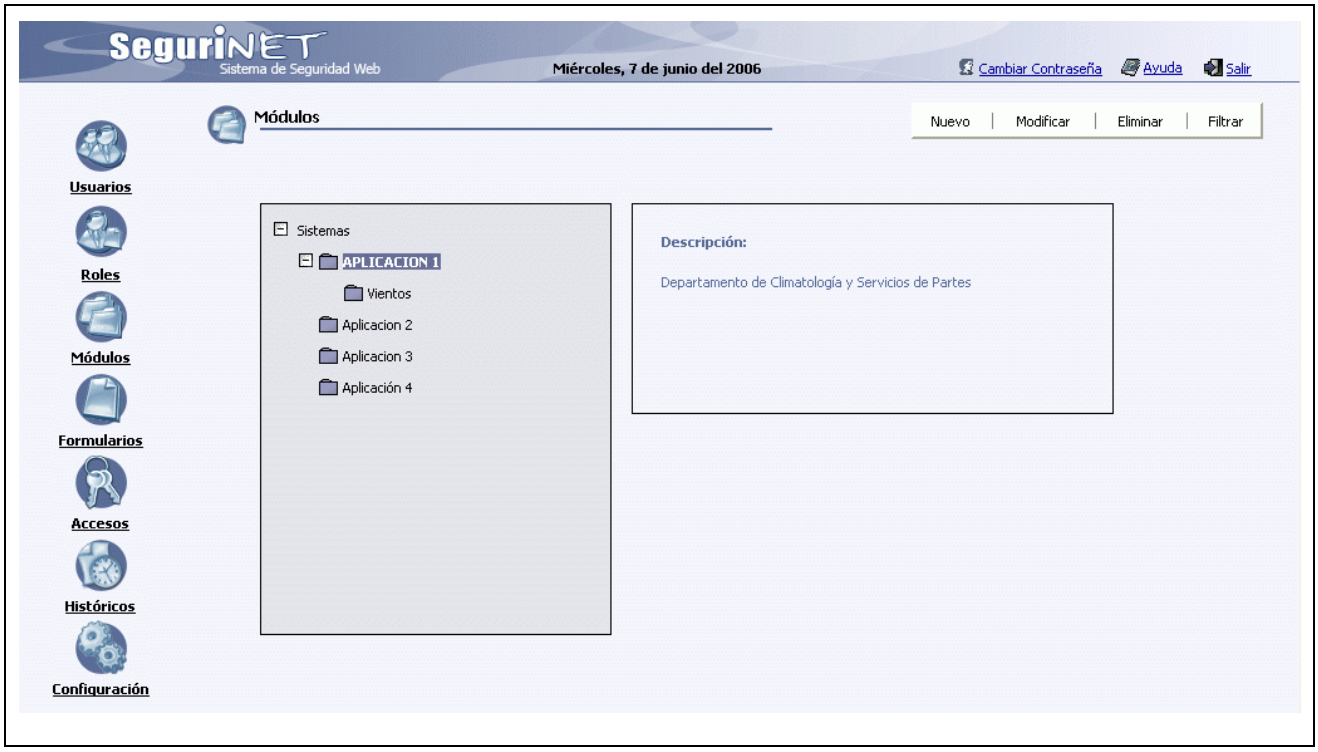
Anexo 2 Imágenes del sistema SEGURINET

Página de entrada al sistema

Página de administración de usuarios



Página de administración de módulos



Página de cambio de contraseña