Universidad de las Ciencias Informáticas

Facultad 2

Título: Gestión de la Seguridad Informática. Sistema para la Gestión de Riesgos.

Trabajo de Diploma para optar por el título de
Ingeniero Informático

Autor: Gerlin Vidal Rodriguez

Tutor: Ing. Dariena Ramirez Lujan

Co-tutor: Ing. Raydel Montesino Perurena

Junio 2008

DECLARACIÓN DE AUTORÍA

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

Para que así conste firmo la presente a los ____ días del mes de _____ del año _____.

<Nombre autor>

_____

Firma del Autor

<Nombre tutor>

_____

Firma del Tutor

DEDICATORIA

A mis padres por estar siempre a mi lado, educarme sin descanso, por todo su apoyo, compresión, por sus oportunos consejos y su entrega sin límites. Especialmente a mi mamita, quien ha sido mi guia incondicional, por su amor, dedicación, por su preocupación constante y darme fe.

A mi hermanita, por comportarse como una madre, darme mucho aliento y no perder jamás la confianza en mi, te adoro.

A mi abuela por seguir mis pasos y pensar siempre en mí.

A mi tía Josefa por quererme como a un hijo y estar atenta en todo momento de mí.

A mi cuñado Jiuber por toda su gradiosa ayuda durante todos estos años.

A toda mi familia en general, que siempre ha aportado un granito de arena en mi educación, gracias de todo corazón.

A mis amigos que sin ellos todo este sueño no se hubiera hecho realidad.

AGRADECIMIENTOS

A mi tutora, por todo su apoyo e instrucción, por guiarme y transmitirme un poquito de su sabiduría y constribuir con ello, en mi formación profesional.

Agradezco infinitamente a todos mis amigos, por apoyarme incondicionalmente y de manera especial a aquellos que me han acompañado de cerca durante todos estos años de la carrera.

A la Revolución, por darme la oportunidad de cumplir muchos de mis sueños.

Gracias a todos.

DATOS DE CONTACTO

Tutor:
Ing. Dariena Ramírez Lujan

Síntesis del Tutor:
Graduada de Ingeniero en Ciencias Informáticas en la Universidad de las Ciencias Informáticas (UCI) en el año 2007. Actualmente imparte clases de la asignatura de Ingeniería de Software en la facultad 2 y es la Asesora del Grupo de Calidad en dicha facultad.

Co-Tutor:
Ing. Raydel Montesino Perurena

Síntesis del Tutor:
Graduado de Ingeniero en Telecomunicaciones y Electrónica en el Instituto Superior Politécnico José Antonio Echeverría (CUJAE) en el año 2003. Profesor de la Universidad de las Ciencias Informáticas desde el 2003 y Director de Redes y Seguridad Informática de la UCI desde el año 2005.

RESUMEN

El presente trabajo procura lograr una mejor eficiencia en la gestión de la Seguridad Informática y particularmente en la gestión de riesgos en una Organización como la Universidad de las Ciencias Informáticas que hace uso de las TICs. Se abordan los principales aspectos desarrollados con el propósito de realizar el análisis y el diseño de una propuesta de Sistema para la Gestión de Riesgos basada en una metodología para estos fines que, con su posterior implementación, se podrá emplear como herramienta de apoyo en el análisis y gestión de riesgos en el sistema de información de dicha Organización, lo cual permitirá; mitigar y dar tratamiento a los riegos y llevarlo los hasta los nivel de aceptación establecidos por la entidad, asegurar los activos y bienes informáticos y la adecuada elaboración del Plan de Seguridad de la Información. El principal resultado de esta investigación es la elaboración de una propuesta de aplicación, que cumpla con las exigencias necesarias para la implementación de un sistema que permita analizar y gestionar los riesgos de un sistema de información.

PALABRAS CLAVE
Gestión, riesgo, impacto, activos, amenazas, salvaguardas.

INDICE

INTRODUCCIÓN

A fines del siglo XX las sociedades avanzadas han sido denominadas frecuentemente sociedades de la información, puesto que el volumen de datos procesado, almacenado y transmitido se hizo cada vez mayor. Actualmente en el siglo XXI la información es poder, por ello las organizaciones la valoran mucho. Gran parte de los datos que las entidades de nuestra sociedad manejan, han sido tratados, ya sea durante su proceso, almacenamiento, o transmisión, mediante las llamadas tecnologías de la información, entre las que ocupa un lugar fundamental la informática. Como consecuencia de lo anterior la seguridad informática se convierte en un tema de crucial importancia para el continuo progreso de la sociedad, e incluso para su propia supervivencia.

El vertiginoso avance de las comunicaciones, la conexión entre las computadoras, las posibilidades de transmisión de datos entre ellas, así como el uso de Internet, ha causado no pocos problemas y ha sido el factor esencial que ha hecho que la Seguridad Informática y sus estándares cobren una importancia vital en el uso de sistemas informáticos conectados, tanto a usuarios aislados como a pequeñas y grandes redes.

Hoy en dia, dicha posibilidad de interconectarse a través de redes, ha abierto nuevos horizontes para explorar más allá de las fronteras nacionales, situación que ha llevado la aparición de nuevas amenazas en los sistemas computarizados y con ello, a que muchas organizaciones gubernamentales y no gubernamentales internacionales desarrollen documentos y directrices que orientan en el uso adecuado de destrezas tecnológicas y recomendaciones para obtener mayor provecho de las ventajas que esto propicia, así como evitar el uso indebido de la mismas, lo cual puede ocasionar serios problemas en los bienes y servicios de las empresas e instituciones en el mundo. Por otro lado las tecnologías basadas en hardware (dispositivos electrónicos), software (programas) y sus combinaciones, que son empleadas para “garantizar” la seguridad informática en las pequeñas, medianas y grandes empresas, pueden crear una falsa sensación de seguridad sobre la protección contra intrusos externos que desean alterar, apropiarse y utilizar posteriormente la información contenida en las computadoras.

Otros aspectos importantes son las tareas de análisis y gestión de riesgos que no son un fin en sí mismas sino que se encajan en la actividad continua de gestión de la seguridad. El análisis de riesgos permite determinar cómo es, cuánto vale y cuán protegidos se encuentran los activos; mientras que las actividades de gestión de riesgos coordinadas con los objetivos, estrategia y política de la Organización, permiten elaborar un plan de seguridad que, implantado y operado, satisfaga los objetivos propuestos con el nivel de riesgo que acepta la Dirección. Por lo que para tener éxito, la organización debe estar comprometida a tratar la gestión de riesgos de forma proactiva y consistente durante períodos determinados.

La gestión del riesgo se ha convertido en un escollo para la dirección estratégica de las organizaciones que confían en metodologías reconocidas para alimentar sus sistemas de gestión. Especialmente duro se hace gestionar el riesgo en aquellas empresas cuyos procesos críticos reposan en tecnologías de la información. La gestión de la seguridad de la información es muy extensa, pero sin duda alguna, uno de sus puntos claves es la adecuada gestión del riesgo.

La seguridad informática se encarga de la identificación de las vulnerabilidades del sistema y del establecimiento de contramedidas que eviten que las distintas amenazas posibles exploten dichas vulnerabilidades.

En Cuba, dada la importancia del tema de seguridad de la información, aún cuando no esté informatizado el proceso, se recomienda realizar el análisis de riesgos en aquellos lugares donde se archiva y custodia información sensible y que a mediano o largo plazo debe ser objeto de atención en el proceso de Informatización de la Organización. Una máxima de la seguridad informática es que: "No existe ningún sistema completamente seguro". Existen sistemas más o menos seguros y más o menos vulnerables, pero la seguridad nunca es absoluta.

La Universidad de las Ciencias Informáticas es una Organización que cuenta con un amplio movimiento productivo, posee un gran número de activos, los que pueden estar expuestos a innumerables amenazas, o incluso, aun protegidos no contar con las medidas de protección adecuadas, o en el peor de los casos no se conoce de estos el nivel de riegos. Igualmente hace uso de las TICs para la prestación de sus servicios y proteger su información. Sin embargo carece de una herramienta que permita minimizar los riesgos sobre sus sistemas informáticos y propicie la continuidad de los procesos.

Dada la importancia de proteger los bienes informáticos que conforman el sistema de información de una Organización y a la primordial necesidad de reducir los riesgos a que estos están sometidos diariamente, se llevará a cabo la puesta en marcha de esta investigación con el fin de resolver el siguiente problema: ¿Cómo minimizar los riesgos que afectan la seguridad informática en los sistemas informaticos de la UCI? Para ello, se plantea como objeto de estudio: los procesos de desarrollo de procedimientos para el análisis y gestión de riesgos, enmarcando el campo de acción: en el desarrollo de un procedimiento para el análisis y gestión de riesgos a los sistemas de información en las Organizaciones. Para poder darle solución al problema anteriormente planteado se define como objetivo general: Realizar análisis y diseño de una herramienta que permita realizar un correcto análisis y gestión de riesgos sobre los activos y/o sistemas informáticos de la UCI.

De acuerdo al objetivo formulado y en correspondencia con el alcance de este, se da respuesta a las siguientes preguntas científicas:

1. ¿Qué metodologías aplicables actuales en materia de seguridad informática se necesitan analizar para poder elaborar una propuesta adecuada?
2. ¿Qué herramienta pudiera proponerse que permita realizar una correcta identificación y un certero análisis de riesgos a los activos y/o sistemas informáticos de una organización?

Para dar respuesta a las preguntas anteriores fue necesario realizar las siguientes tareas de investigación:

- Estudio del estado del arte de metodologías y sistemas de Análisis y Gestión de Riesgos de los Sistemas de Información.
- Realizacion de análisis y diseño de una herramienta para el análisis y gestión de riesgos.

El presente trabajo esta estructurado por los siguientes capítulos:

- **Capitulo 1 Fundamentación Teórica:** En este capítulo se tratarán los principales conceptos y términos abordados en la investigación. Se describen y explican, además, las tendencias, herramientas, norma internacional y metodologías actuales que se han empleado para darle solución al problema planteado.
- **Capítulo 2 Características del sistema:** En este capítulo se realiza una descripción general del sistema propuesto. Se representa el Modelo de Dominio. Se realiza la especificación de los requisitos de software que debe cumplir dicho sistema. Se definen y describen los casos de uso y la relación con los actores que se modelan en el Diagrama de Casos de Uso del Sistema.
- **Capítulo 3 Análisis y Diseño del sistema:** En este capítulo se muestra cómo está definida la arquitectura candidata del sistema. Se representa una estructura global del mismo, así como la modelación de los artefactos necesarios para su construcción. Se representan los componentes de la aplicación tratados como clases y representados a través de diagramas de clases con estereotipos Web. Se presenta el Modelo lógico de datos y Modelo físico de datos.
- **Capítulo 4 Estudio de la factibilidad:** Este capítulo contiene el método de estimación por puntos de caso de uso, mediante el cual se obtiene el esfuerzo y costo del proyecto, el tiempo de desarrollo en meses, costo y la cantidad de personas que se necesitan para su desarrollo. Comprende además el análisis del costo y el análisis del beneficio tangible e intangible.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

1.1 Introducción.

En el presente capítulo se tratarán los principales conceptos y términos abordados en la investigación. Se describen y explican, además, las tendencias, herramientas, norma internacional y metodologías actuales que se han empleado para darle solución al problema planteado.

1.2 Conceptos Fundamentales.

1.2.1. ¿Qué es Seguridad Informática?

La seguridad informática puede definirse como el conjunto de reglas, planes y acciones que permiten asegurar la información contenida en un sistema computacional o la capacidad de mantener intacta y protegida la información de sistemas informáticos, permitiendo asegurar que los recursos del sistema de información generalmente (material informático o programas) de una organización sean utilizados de manera que no sea fácil de acceder por cualquier persona que no se encuentre acreditada.
Lógicamente, este término abarca mucho más que la protección de los datos siendo este, uno de los más importantes aspectos.[1]

Áreas que cubre la seguridad informática:

- Políticas de Seguridad
- Seguridad Física
- Autentificación
- Integridad
- Confidencialidad
- Control de Acceso
- Auditoría

1.2.2. ¿Qué significa Riesgo?

Del concepto de riesgo, hay muchas acepciones y diferentes usos de éste término, defendidas por diferentes autores, y visto desde diferentes puntos de vista.
Entre las variadas formas de interpretar el riesgo, podemos encontrar:

- contingencia o posibilidad de que suceda un daño, desgracia o contratiempo.
- conjunto de circunstancias que pueden disminuir el beneficio.
- Exposición. Peligro, es una contingencia inminente o muy probable, en tanto que riesgo y exposición pueden expresar desde la mera posibilidad a diversos grados de probabilidad.

- Esta fase tiene por objeto la identificación del riesgo delimitando su contenido y alcance para diferenciarlo de otros riesgos. Se basa en la identificación específica de sus elementos característicos como son: el bien y el daño.

En este trabajo, entenderemos por Riesgo: Estimación del grado de exposición a que una amenaza se materialice sobre uno o más activos causando daños o perjuicios a la Organización.[2]
El riesgo indica lo que podría pasar a los activos sino se protegieran adecuadamente. Es importante saber qué características son de interés en cada activo, así como saber en qué medidas estas características están en peligro, o sea, analizar el sistema.

1.2.3 ¿Qué significa Análisis de riesgos?

Como herramienta de diagnóstico para poder establecer la exposición real a los riesgos por parte de una organización se recurre a lo que se llama el Análisis de Riesgo. Es conocido como el corazón de toda actuación organizada en materia de seguridad y de la gestión global de seguridad, ya que permite el establecimiento de un nivel adecuado de seguridad tanto del software que se desea proteger como el que se desarrollará.

El Análisis de Riesgos implica:

- ✓ 1.-Determinar qué se necesita proteger.
- ✓ 2.- De qué hay que protegerlo.
- ✓ 3.-Cómo hacerlo.

El análisis de riesgo involucra un proceso de administración de riesgos, el cual es llevado en forma continua, dado que es necesario evaluar periódicamente si los riesgos identificados y la exposición a los mismos calculada en etapas anteriores se mantienen vigentes. La importancia del análisis de riesgo radica en que permite identificar los impactos futuros de todo proyecto en la estructura de riesgos de la organización.

En resumen, **Análisis de Riesgo:** Es el proceso sistemático para estimar la magnitud de los riesgos a que esta expuesta la Organización. Permite la determinación de cómo es, cuánto vale y cómo de protegidos se encuentran los activos y proporciona un modelo del sistema de información en términos de activos, amenazas y salvaguardas.[3]

1.2.4. ¿Qué es Gestión de riesgos?

Es la selección e implantación de salvaguardas para conocer, prevenir, reducir o controlar los riesgos identificados.

1.2.5. ¿En qué consiste el Proceso de Análisis y Gestión de Riesgos?

En el proceso de análisis y gestión de riesgos de la seguridad en los sistemas de información se puede identificar las siguientes etapas:

- Planificación.

En esta fase, se establece el objetivo del proyecto, el dominio de estudio y las restricciones generales. Deben también definirse las métricas con las que se valorarán los diferentes elementos de seguridad, de manera que los resultados finales de medición del riesgo sean definidos en función de los parámetros adecuados para cuantificar el riesgo por la organización (por ejemplo, definir la escala de frecuencias para medir la vulnerabilidad, definir las cantidades monetarias por las que cuantificar el impacto, etc.).

- Análisis de riesgos.

Una vez definido el dominio, los analistas de riesgos procederán a realizar las entrevistas al personal de la organización para la obtención de información. En esta fase se identificarán los activos de la organización, identificando las relaciones que se establecen entre activos. De esta forma se obtiene el "árbol de activos" que representan las distintas dependencias y relaciones entre activos, es decir, todos aquellos elementos que están "encadenados entre sí" en términos de seguridad.

También se identifican el conjunto de amenazas, estableciendo para cada activo, cual es la vulnerabilidad que presenta frente a dicha amenaza. Además, se cuantifica el impacto, para el caso en el que la amenaza se materializase.

Dado que los activos se encuentran jerarquizados y se encuentran establecidas las relaciones de dependencia entre los activos de las diferentes categorías, hemos conseguido de forma explícita documentar la "cadena de fallo" en caso de un incidente de seguridad.

La experiencia y la sucesiva revisión de la información generada en estudios de riesgos anteriores permitirán ajustar de forma más exacta las diferentes dependencias entre activos. Con toda esta información, tendremos una estimación del costo que podría producir la materialización de una amenaza sobre un activo. Teniendo en cuenta las relaciones funcionales y de dependencias entre activos, se hayan los valores de riesgo.

- Gestión de riesgos.

En esta fase, se procede a la interpretación del riesgo. Una vez identificados los puntos débiles, deben seleccionarse el conjunto de funciones de salvaguarda que podrían ser usados para disminuir los

niveles de riesgo a los valores deseados. Para ello, deberán especificarse los mecanismos de salvaguarda que se encuentran implantados hasta ese momento y cual es su grado de cumplimiento.
Este proceso se ayuda de la simulación. Se van probando selecciones de diferentes mecanismos de salvaguarda y se estudia en que medida reducen los niveles de riesgo a los márgenes deseados. Es muy importante realizar las correctas estimaciones de la efectividad de los diferentes mecanismos de salvaguarda para ajustar de forma precisa los valores de riesgo.

- Selección de mecanismos de salvaguarda.

Una vez obtenidos estos resultados, se establecen de nuevo reuniones con el equipo responsable del proyecto de la organización en estudio. De esta forma, se analizan los resultados obtenidos y se establece un plan de implantación de mecanismos.

1.3 Norma Internacional de Seguridad Informática a utilizar.

La serie ISO/IEC 27000: Es un conjunto de estándares desarrollados o en fase de desarrollo por ISO (International Organization for Standardization) e IEC (International Electrotechnical Commission), que proporcionan un marco de gestión de la seguridad de la información utilizable por cualquier tipo de organización, pública o privada, grande o pequeña. Con esta nueva serie, se busca dar un carácter auto consistente e integral al conjunto de normas de seguridad de la información.[4]

ISO 27002: 2005: Es el nuevo nombre de ISO 17799:2005, manteniendo 2005 como año de edición. Es una guía de buenas prácticas que describe los objetivos de control y controles recomendables en cuanto a seguridad de la información. No es certificable. Contiene 39 objetivos de control y 133 controles, agrupados en 11 dominios.[4]
Breve resumen del contenido:

- Introducción: conceptos generales de seguridad de la información y SGSI.
- Campo de aplicación: se especifica el objetivo de la norma.
- Términos y definiciones: breve descripción de los términos más usados en la norma.
- Estructura del estándar: descripción de la estructura de la norma.

- Evaluación y tratamiento del riesgo: indicaciones sobre cómo evaluar y tratar los riesgos de seguridad de la información.

- Política de seguridad: documento de política de seguridad y su gestión.

- Aspectos organizativos de la seguridad de la información: organización interna; terceros.

- Gestión de activos: responsabilidad sobre los activos; clasificación de la información.

- Seguridad ligada a los recursos humanos: antes del empleo; durante el empleo; cese del empleo o cambio de puesto de trabajo.

- Seguridad física y ambiental: áreas seguras; seguridad de los equipos.

- Gestión de comunicaciones y operaciones: responsabilidades y procedimientos de operación; gestión de la provisión de servicios por terceros; planificación y aceptación del sistema; protección contra código malicioso y descargable; copias de seguridad; gestión de la seguridad de las redes; manipulación de los soportes; intercambio de información; servicios de comercio electrónico; supervisión.

- Control de acceso: requisitos de negocio para el control de acceso; gestión de acceso de usuario; responsabilidades de usuario; control de acceso a la red; control de acceso al sistema operativo; control de acceso a las aplicaciones y a la información; ordenadores portátiles y teletrabajo.
- Adquisición, desarrollo y mantenimiento de los sistemas de información: requisitos de seguridad de los sistemas de información; tratamiento correcto de las aplicaciones; controles criptográficos; seguridad de los archivos de sistema; seguridad en los procesos de desarrollo y soporte; gestión de la vulnerabilidad técnica.

- Gestión de incidentes de seguridad de la información: notificación de eventos y puntos débiles de la seguridad de la información; gestión de incidentes de seguridad de la información y mejoras.

- Gestión de la continuidad del negocio: aspectos de la seguridad de la información en la gestión de la continuidad del negocio.

- Cumplimiento: cumplimiento de los requisitos legales; cumplimiento de las políticas y normas de seguridad y cumplimiento técnico; consideraciones sobre las auditorias de los sistemas de información.

- Bibliografía: normas y publicaciones de referencia.

1.3.1. ¿Qué beneficios brinda esta norma?

- Establecimiento de una metodología de gestión de la seguridad clara y estructurada.
- Reducción del riesgo de pérdida, robo o corrupción de información.
- Los clientes tienen acceso a la información a través medidas de seguridad.
- Los riesgos y sus controles son continuamente revisados.
- Confianza de clientes y socios estratégicos por la garantía de calidad y confidencialidad comercial.
- Las auditorias externas ayudan cíclicamente a identificar las debilidades del sistema y las áreas a mejorar.
- Posibilidad de integrarse con otros sistemas de gestión (ISO 9001, ISO 14001, OHSAS 18001…).
- Continuidad de las operaciones necesarias de negocio tras incidentes de gravedad.
- Conformidad con la legislación vigente sobre información personal, propiedad intelectual y otras.
- Imagen de empresa a nivel internacional y elemento diferenciador de la competencia.
- Confianza y reglas claras para las personas de la organización.
- Reducción de costes y mejora de los procesos y servicios.
- Aumento de la motivación y satisfacción del personal.
- Aumento de la seguridad en base a la gestión de procesos en vez de en la compra sistemática de productos y tecnologías.

1.4. Fundamentación de la metodología a utilizar.

Las metodologías de desarrollo de software son un conjunto de procedimientos, técnicas y ayudas a la documentación para el desarrollo de productos software. Además detallan la información que se debe producir como resultado de una actividad y la información necesaria para comenzarla.[5]

Es preciso tener claro que establecer contramedidas para mitigar absolutamente todos los riesgos es algo desmesurado, por cuestiones económicas y de índole operativa, y que tampoco sería correcto asumir la totalidad de los riesgos, sin invertir en ninguna medida de control de los mismos. Es necesario llegar a un equilibrio entre inversión y riesgo asumido voluntariamente, y éste es el objetivo principal de la gestión de los riesgos, tal y como se ha explicado.

Maneras de gestionar adecuadamente el riesgo hay muchas. Desde luego, siempre se aconseja emplear metodologías reconocidas ya que éstas emanan de una experiencia y un contraste que las hacen válidas a priori.

1.4.1. Metodologías y mejores prácticas.

Para llevar a cabo el proceso de análisis del riesgo, existen diversas metodologías que son utilizadas, entre algunas de las más importantes se puede mencionar a:

- ✓ 1. MAGERIT (Metodología de Análisis y Gestión de Riesgos de los Sistemas de Información). Elaborada por el Consejo Superior de Administración Electrónica de España, es una metodología de carácter público, perteneciente al Ministerio de Administraciones Públicas. Tiene como objetivo estudiar los riesgos que soporta un sistema de información y el entorno asociado a él. Está conformada por una serie de técnicas específicas para el análisis de riesgos, análisis mediante tablas, análisis algorítmico, árboles de ataque, técnicas generales, análisis costo-beneficio, entre otros.
- ✓ 2. EBIOS (Expression des Besoins et Identification des Objectifs de Sécurité - Expresión de las Necesidades e Identificación de los Objetivos de Seguridad). Es promovido por la Dirección Central de Seguridad de Sistemas de Información (DCSSI-Francia) como norma internacional. Es un software de asistencia bajo licencia libre. Su enfoque simple y modular le permite adaptarse a todos los contextos y a distintas acciones de seguridad. Permite apreciar y tratar los riesgos relativos a la seguridad de los sistemas de información (SSI).

- ✓ 3. OSSTMM (Open Source Security Testing Methodology Manual). Manual de metodología Abierta de Testeo de Seguridad es una metodología para realizar análisis de vulnerabilidad que está basada en entregar resultados cuantificables y que ha sido desarrollada dentro de un proyecto de comunidad Open Source. Es la respuesta cuando un analista de seguridad informática se pregunta por dónde empezar, qué y cómo analizar, cómo presentar los resultados.
- ✓ 4. OWASP (Open Web Application Security Project). Es un organismo sin ánimo de lucro creado en Estados Unidos y que cuenta con más de 60 capítulos locales repartidos en todo el mundo. Su objetivo es ayudar a las empresas a entender y mejorar la seguridad de sus aplicaciones y servicios Web. El proyecto crea documentación, herramientas y estándares Open Source sobre seguridad en Aplicaciones Web gracias a expertos de la comunidad internacional que, de forma voluntaria, colaboran en los distintos proyectos.

1.4.2. ¿Por qué usar la Metodología MAGERIT (Metodología de Análisis y Gestión de Riesgos de los Sistemas de Información)?

Entre las razones por las que se selecciona la metodología MAGERIT version 2.0, se debe citar que proporciona un buen número de herramientas para obtener un mapa de todos los riesgos que se desean controlar y representar, lo que facilita enormemente la toma de decisiones. Considera acertadamente que la gestión del riesgo es el "alma mater" de toda actuación organizada en materia de seguridad y, por tanto, de la gestión global de la misma. Otros de los motivos de esta selección se materializan en los objetivos de esta metodología los cuales se mencionan a continuación:

MAGERIT persigue los siguientes objetivos:

- Concientizar a los responsables de los sistemas de información de la existencia de riesgos y de la necesidad de atajarlos a tiempo.
- Ofrecer un método sistemático para analizar tales riesgos.
- Ayudar a descubrir y planificar las medidas oportunas para mantener los riesgos bajo control.
- Apoyar la preparación a la Organización para procesos de evaluación, auditoría, certificación o acreditación, según corresponda en cada caso.

Así mismo, se ha cuidado la uniformidad de los informes que recogen los hallazgos y las conclusiones de un proyecto de análisis y gestión de riesgos: modelo de valor, mapa de riesgos, evaluación de salvaguardas, estado de riesgo, informe de insuficiencias, y plan de seguridad.

En sus inicios esta metodología sirvió para darle solución al tema de gestión de riesgos pero el tiempo ha demostrado que su uso es ilimitado, no está sustentada prácticamente por algún sistema, o archivado en las instituciones sin que se le de un uso práctico.

La herramienta PILAR es un procedimiento informático-lógico para el análisis y la gestión de los riesgos de un sistema de información siguiendo la Metodología MAGERIT. Esta es de uso exclusivo en la administración pública española. Se puede descargar de http://www.ar-tools.com/pilar.

1.4.3. Rational Unified Process (RUP).

Rational Unified Process (RUP), es un proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado (UML), constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos.

La metodología que se emplea para el desarrollo de este trabajo es RUP (Proceso Unificado de Desarrollo). Es la metodología que mejor se ajusta a las necesidades que existen actualmente en el desarrollo de software, pues propone un Modelo iterativo e incremental, muy acorde con la naturaleza cambiante de los requisitos en muchos proyectos.

RUP es en esencia un proceso o metodología de desarrollo de software: Es una forma disciplinada de asignar tareas y responsabilidades en una empresa de desarrollo (Quién debe hacer Qué, Cuándo y Cómo debe hacerlo), y tiene como objetivo asegurar la producción de software de calidad dentro de plazos y presupuestos predecibles.

El ciclo de vida de RUP se caracteriza por ser: dirigido por casos de uso, centrado en la arquitectura e Iterativo (mini-proyectos) e Incremental (versiones). [6]

Además, cubre el ciclo de vida de desarrollo de un proyecto y toma en cuenta las mejores prácticas a utilizar en el modelo de desarrollo de software. Las cuales se muestran a continuación:

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

1.5. Herramientas existentes a considerar para la gestión de riegos.

1.5.1. Herramientas más usadas en el mundo.

Existe gran cantidad de herramientas de software de Gestión de Riesgos disponibles en el mercado y que siguen algunas metodologías. Estas herramientas se enfocan sólo en una categoría de riesgos (TRIMS – Technical Risk Identification and Mitigation System), o están orientadas a compañías maduras que poseen una amplia base de datos organizacional que les permita generar información de categorías propias de riesgos (Risk Track y WelcomRisk), o bien emplean un mecanismo que no se orienta al uso de taxonomías (ARM – Active Risk Manager) [FUENTE y LOVELLE, 2006].

Aún cuando son conocidas algunas herramientas que o bien se enfocan solo en una categoría de riesgos como TRIMS, o bien están orientadas a compañías que poseen una amplia base de datos organizacional que les permite generar información de categorías propias de riesgos; tal es el caso de RiskTrak y WelcomRisk, o bien emplean un mecanismo que no se orienta al uso de Taxonomías a manera de Active Risk Manager y que por lo general son propietarias.

Producto	Licencia:	Proveedor	Descripción	Sistemas Operativos	Plataforma
Active Risk Manager (ARM)	Propietario	Strategic Thought	Herramienta integrada de Administración de Riesgos que brinda una solución para la Identificación de Riesgos mediante la utilización de la información contenida en la WBS de proyecto.	Windows 98, ME, NT, 2000, y XP.	Web Based
Technical	Libre.	Best	Herramienta	Windows 98,	Win 32

Risk And Mitigation System (TRIMS)		Manufacturing Practices	integrada de Administración de Riesgos que emplea ingeniería de conocimiento y que se enfoca en la Identificación y medición de riesgos técnicos de proyecto	ME, NT, 2000, y XP.	
RiskTrak	Propietario	Risk Services & Technology	Herramienta integrada a la Administración de Riesgos que brinda una solución para la Identificación de Riesgos mediante el empleo de bases de datos	Windows 98, ME, NT, 2000, y XP.	Win 32
WelcomRisk	Propietario	Welcom	Herramienta integrada a la Administración de Riesgos que brinda una solución para la Identificación Sistemática de Riesgo	Windows 98, ME, NT, 2000, y XP.	Win32

			mediante la utilización de bibliotecas configurables de categorías de riesgo.		
Chichón – Análisis de Riesgo		Free	Chichón es una herramienta para analizar cuantitativamente el riesgo de un sistema (de información). La herramienta sigue el Modelo de Magerit 1.0	Multiplataforma	Java
PILAR	Propietario	Consejo Superior de Administración Electrónica. España. A.L.H. J. Mañas S.L.	Esta herramienta es un procedimiento informático-lógico para el análisis y la gestión de los riesgos de un sistema de información siguiendo la Metodología MAGERIT.	Multiplataforma	Java
MSAT	Libre	Microsoft	Es una	Windows 2000	Win32

(Microsoft Security Assessment Tool)			herramienta, para ayudar a organizaciones a evaluar las debilidades de su entorno de TI actual, relevar una lista de temas prioritarios y brindarle una guía que lo ayudará especialmente a minimizar los riesgos. Comienza el proceso con una reseña del estado actual de su seguridad y luego utiliza la MSAT para controlar en forma continua la capacidad de su infraestructura para responder a las amenazas a su seguridad.	Professional Edition ; Windows Vista; Windows XP Professional Edition	

Tabla 2.1 Resumen de herramientas.

1.5.2. Herramienta usada en Cuba.

Existe un sistema de medidas para la seguridad informática, denominado Softar v 2.3 [7], incluye el establecimiento de las políticas y procedimientos que conforman una estrategia de cómo tratar los aspectos de seguridad. El mismo está basado en la metodología para el documento Diseño de un sistema de seguridad informática, aplicado en Cuba desde 1999 hasta la fecha.

1.6. Herramientas CASE.

Las Herramientas CASE (Computer Aided Software Engineering, Ingeniería de Software Asistida por Ordenador) son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el coste de las mismas en términos de tiempo y de dinero. Estas herramientas nos pueden ayudar en todos los aspectos del ciclo de vida de desarrollo del software en tareas como el proceso de realizar un diseño del proyecto, cálculo de costes, implementación de parte del código automáticamente con el diseño dado, compilación automática, documentación o detección de errores entre otras.[8]

1.6.1. Visual Paradigm.

Visual Paradigm es una poderosa herramienta CASE que al igual que el Rational Rose utiliza UML para el modelado, es la aplicación por excelencia para ser utilizada en un ambiente de software libre. Permite crear tipos diferentes de diagramas en un ambiente totalmente visual. Es muy sencillo de usar, fácil de instalar y actualizar. Genera código para varios lenguajes. Tiene integrado el MS Visio y es compatible con otras ediciones. Un entorno de creación de diagramas para UML 2.0.[9]

Potencialidades de Visual Paradigm:

- Licencia: Gratuita y Comercial.
- Un entorno de creación de diagramas para UML 2.0.
- Diseño centrado en casos de uso y enfocado al negocio que generan un software de mayor calidad.
- Uso de un lenguaje estándar común a todo el equipo de desarrollo que facilita la comunicación.
- Capacidades de ingeniería directa e inversa.
- Modelo y código que permanece sincronizado en todo el ciclo de desarrollo.

- Disponibilidad de múltiples versiones, para cada necesidad.
- Disponibilidad de integrarse en los principales IDEs.
- Disponibilidad en múltiples plataformas.
- Posibilita la representación gráfica de los diagramas permitiendo ver el sistema desde diferentes perspectivas, como el de componentes, despliegue, secuencia, casos de uso, clase, actividad, estado, entre otros. Además, identifica requisitos y comunica información, se centra en cómo los componentes del sistema interactúan entre ellos, sin entrar en detalles excesivos, además, permite ver las relaciones entre los componentes del diseño y mejora la comunicación entre los miembros del equipo usando un lenguaje gráfico.

1.6.2. Embarcadero ERStudio 7.1.

Embarcadero ERStudio es una de las herramientas CASE de diseño de bases de datos que ayuda a generar, mantener alta calidad y gran rendimiento en las aplicaciones de la base de datos desde un modelo lógico de los requerimientos de información y las reglas de negocio que definen la base de datos al modelo físico optimizado por las características especificas de ésta. Permite visualizar la estructura, elementos clave y optimizar el diseño de las bases de datos, genera tablas u otras especificaciones en dependencia de la plataforma seleccionada.

Principales características:

- Arquitectura de diseño en capas: Erwin0 proporciona la flexibilidad necesaria para generar el modelo de datos que mejor satisface las necesidades de las entidades. Ofrece soporte para modelos lógicos y físicos independientes, así como para el modelo lógico/físico combinado tradicional.
- Tecnología de transformación: El diseño físico de una base de datos coincide raras veces con el diseño lógico inicial de los datos. Las limitaciones de las entidades imponen la necesidad de modificar tablas para cumplir los requisitos de rendimiento de las aplicaciones actuales. La tecnología de transformación permite implementar este tipo de cambios y a la vez mantener la integridad del diseño original.
- Definición de estándares: La herramienta ofrece soporte para la definición y el mantenimiento de estándares mediante el Diccionario de Dominios y el Editor de Estándares de tipos de datos, permite que el usuario defina los estándares para la asignación de tipos definidos por el usuario y predeterminadas a tipos de datos específicos de cada sistema gestor de base de datos.

- Gestión de modelos de gran tamaño: Proporciona una visión específica para usuarios individuales y dividen los modelos en subconjuntos más pequeños y manipulables.

Ventajas:

- Facilidades de diseño de diagramas Entidad-Relación y Entidad-Relación extendido y transformación de este al modelo relacional (en tercera forma normal, preservando las dependencias funcionales y sin pérdidas de información).
- Comparación comprensiva entre el modelo de datos y la base de datos.
- Soporta la separación del modelo lógico y del físico.

1.7. Plataforma y lenguaje de programación.

Para el desarrollo de esta propuesta de aplicación se ha escogido como lenguaje de programación a utilizar PHP 5 y como sistema gestor de base de datos PostgreeSQL. Esta decisión está basada y fundamentada en el estudio de las tendencias actuales de desarrollo de aplicaciones de este tipo y dadas las características y ventajas que estas herramientas posibilitan, las cuales se muestran a continuación.

1.7.1. Lenguaje de programación a utilizar.

Para la implementación de la aplicación web se propone como lenguaje de programación PHP5 ya que es uno de los más extendidos en la red de redes y ha sido aceptado precisamente por la simplicidad y potencia que lo caracteriza, ofrece gran variedad de funciones para la explotación de bases de datos sin grandes complicaciones. Es un lenguaje multiplataforma completamente gratuito que puede ser ejecutado en la mayoría de los sistemas operativos tales como UNIX, Windows y Mac OS X, y puede interactuar con los servidores web más populares, pues existe en versión CGI, módulo para Apache, e ISAPI.

Fue lanzado bajo la licencia BSD, no obliga a entregar el código fuente, pero sí impone la irritable cláusula publicitaria de este tipo de Licencia. Permite la conexión a diferentes tipos de servidores de bases de datos tales como: MySQL, PostgreSQL, Oracle, ODBC, DB2, Microsoft SQL Server, Firebird y SQLite; lo cual permite la creación de aplicaciones web muy robustas[10]. Es un lenguaje fácil de aprender y de aplicar, consume pocos recursos y con gran rapidez de ejecución, contiene funciones

para trabajar virtualmente con todas las tecnologías para la web existentes hoy, por lo que es muy empleado para el desarrollo de aplicaciones web.

PHP no es un lenguaje excesivamente complejo y con una curva de aprendizaje asequible, es bastante fácil de aprender pero también te permite aprender características de lenguajes más complejos como son los lenguajes de programación orientados a objetos.

Es posiblemente el lenguaje web más popular lo que hace que haya numerosos tutoriales y ejemplos de código que agilizan el periodo de aprendizaje.[11]

1.8. Sistema Gestor de Base de Datos (SGBD).

Los sistemas de gestión de base de datos son un tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan. Se compone de un lenguaje de definición de datos, un lenguaje de manipulación de datos y un lenguaje de consulta.

1.8.1. PostgreSQL.

PostgreSQL es un gestor de base de datos relacional libre, liberado bajo la licencia BSD. Es una alternativa a otros sistemas de bases de datos de código abierto (como MySQL, Firebird y MaxDB), así como sistemas propietarios como Oracle o DB2.

Algunas de sus principales características son:

- Claves ajenas también denominadas llaves ajenas o llaves foráneas (foreign keys) Disparadores (triggers).
- Vistas Integridad transaccional.
- Acceso concurrente multiversión (no se bloquean las tablas, ni siquiera las filas, cuando un proceso escribe).
- Capacidad de albergar programas en el servidor en varios lenguajes.
- Herencia de tablas, tipos de datos y operaciones geométricas.

1.9. Servidor Web.

El servidor web es un programa que corre sobre el servidor que escucha las peticiones HTTP que le llegan desde el cliente, en este caso los navegadores. Dependiendo del tipo de la petición, el servidor

web buscará una página web o bien ejecutará un programa en el servidor. De cualquier modo, siempre devolverá algún tipo de resultado HTML al navegador que realizó la petición.

El mundo está dividido por dos grandes grupos de servidores web, el IIS (Internet Information Server) de Microsoft, y el Apache un proyecto libre de la Fundación Apache, gratuito y de código abierto.

En nuestro caso hemos decidido usar el servidor web Apache ya que es uno de los servidores web más potentes del mercado, ofreciendo una perfecta combinación entre estabilidad y sencillez.

Hoy en día es el servidor web más utilizado del mundo, encontrándose muy por encima de sus competidores, tanto gratuitos como comerciales. Es un software de código abierto que funciona sobre cualquier plataforma, y se distribuye prácticamente con todas las implementaciones de Linux.

1.10 Lenguaje de modelado: UML.

Se utiliza UML (Lenguaje de Modelado Unificado) como lenguaje de modelado para el apoyo de esta metodología debido a que UML es un lenguaje que permite modelar, construir y documentar los elementos que forman un producto de software que responde a un enfoque orientado a objetos y puede también considerarse, como un lenguaje de modelamiento visual que permite una abstracción del sistema y sus componentes. Por otro lado, UML se ha convertido en el estándar internacional e industrial para definir, organizar y visualizar los elementos que configuran la arquitectura de una aplicación orientada a objetos. Puede utilizarse en todo el ciclo de vida de desarrollo de un software, lo que garantiza el modelado de todas las etapas de desarrollo.
En los últimos años, ha tenido pequeñas modificaciones introducidas en el idioma.

UML es un lenguaje estándar para especificar, visualizar, construir y documentar los artefactos (partes) de sistemas de software, así como para modelado en negocios y otros sistemas no basados en software. El UML es una parte muy importante del desarrollo de software orientado a objetos y en el proceso del desarrollo de dicho software. El UML utiliza en su mayoría notaciones gráficas para expresar el diseño de proyectos de software. Utilizar UML ayuda a los equipos de proyecto a comunicar, explorar diseños potenciales y validar el diseño de arquitectura del software.

1.11 Conclusiones parciales.

En este capítulo se desarrolló un estudio de las metodologías y tecnologías que serán utilizadas para el desarrollo del presente trabajo.Se propone la metodologia MAGERIT versión 2.0 como soporte principal del sistema y el uso de herramientas desarrolladas sobre software libre, muchas de las cuales tienen como principal ventaja que son multiplataformas.La herramientas a utilizar son las siguientes:
Herramienta CASE: Visual Parading
Servidor Web: Apache.
Lenguaje de programación: PHP5
Gestor de Base Datos: PostgreSQL 8.0

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA.

2.1. Introducción.

En el presente capítulo se realiza una descripción general del sistema que se propone y cómo debe funcionar el mismo. Se representa el Modelo de Dominio que se plantea automatizar mediante un diagrama de objetos. Se realiza la especificación de los requisitos de software que debe cumplir dicho sistema. Se definen y describen los casos de uso y la relación con los actores que se modelan en el Diagrama de Casos de Uso del Sistema.

2.2 Problema y situación problémica.

La Universidad de Ciencias Informáticas actualmente cuenta con una gran infraestructura de redes, numerables computadoras con informaciones importantes y confidenciales en algunos casos, para el desarrollo de proyectos productivos, un extenso e inmedible intercambio de información y/o transmisión de datos entre todos los usuarios que tienen acceso a los medios y sistemas informáticos con que dispone dicha organización, un amplio número de PC conectadas entre si y depende fundamentalmente de las tecnologías de la información para la prestación de sus servicios, ya sean internos, externos o a terceros. Sin embargo, no dispone de una herramienta que apoye el proceso de gestión de la seguridad informática y posibilite, esencialmente, gestionar con eficiencia los riegos que sobre sus activos inciden frecuentemente como consecuencia de lo explicado anteriormente; de manera que se pueda ofrecer una mayor y mejor protección sobre estos bienes informáticos y por consiguiente de reducir los riesgos a que estos se encuentran sometidos diariamente.

2.3 Objeto de automatización.

Teniendo en cuenta lo planteado anteriormente se hace necesario desarrollar un sistema para la gestión de riesgos, que permita analizar y gestionar los riegos que puedan incidir sobre los activos y bienes informáticos de la organización en cada una de sus áreas de desarrollo o dominios, aplicaciones en desarrollo, etc. Así como también, que sea capaz de brindar las indicaciones precisas u orientaciones objetivas de qué se debe hacer ante estos riesgos y cómo enfrentarlos exitosamente, que permita mantener bajo control la materialización de posibles amenazas sobre sus activos y que brinde la posibilidad de afrontar los impactos y riesgos identificados en los procesos de análisis y gestión que se consideren inaceptables, llevando los mismos a valores aceptables para la organización y su dirección.

2.4 Información que se maneja.

Documentos específicos que se procesen, detalles de la información que se manipule.

2.5 Propuesta de sistema.

De acuerdo a lo expuesto anteriormente como propuesta de solución a nuestro problema científico, se determinó crear un sistema que garantice la gestión de la información relativa a la institución en cuanto a seguridad de los medios informáticos (léase gestión de riesgos).
Esta solución de software está diseñada para lograr una independencia total entre sistemas operativos a través de la red, de ahí radica una de las ventajas de las aplicaciones web.
Dentro de las funcionalidades, se destacan las más importantes:

- Permitir al usuario efectuar informes personalizados de los riesgos correspondientes a los activos informáticos dentro de su organización o área.
- Consultar resultados.
- Permitir la comparación con reportes previos.
- Facilitar el funcionamiento y puesta en práctica de este servicio traerá grandes beneficios a la organización, notando gran mejoría en el proceso de análisis y gestión de riesgos del sistema de Información de la UCI.
- Permitir la identificación de riesgos capaces de afectar el funcionamiento normal del entorno informático.

2.6 Modelo del Dominio.

Para realizar la captura correcta de los requisitos y poder construir un sistema más idóneo y correcto es necesario tener un conocimiento exacto y concreto del funcionamiento del objeto de estudio. Debido a la complejidad de la estructura y que no están bien definidos los procesos del negocio en cuestión, se arriba a la conclusión de que no es posible realizar un modelo de negocio por lo cual se construye un modelo de dominio. Este permite de manera visual, mostrar al usuario los principales conceptos que se manejan en el dominio del sistema en desarrollo, lo cual ayuda a utilizar un vocabulario común para todas las personas involucradas en el desarrollo del proyecto, permitiendo que se pueda entender el contexto en que estructura y construye dicho sistema. Contribuye además, con una mejor comprensión del problema que el sistema resuelve en relación a su contexto. Para describir dicho

modelo, se realiza a un diagrama de clases UML, en el que se especifican las principales clases conceptuales que pueden intervenir en el sistema.

2.6.1 Conceptos del Dominio.

2.6.1.1 Directivo de Seguridad: Es el rol que se le concede a un miembro del Departamento de Seguridad Informática de la Organización con el objetivo de tener el control, dar soporte y administrar el sistema.

2.6.1.2 Responsable de Seguridad: Es un rol que se le concederá al responsable de la seguridad informática en cada dominio o a una persona determinada del mismo, con el objetivo de introducir y gestionar en el sistema los datos de los activos que corresponden a su dominio de seguridad.

2.6.1.3 Usuario: Está compuesto por los Responsables de Seguridad y Directivos de Seguridad, que hagan uso del sistema.

2.6.1.4 Dominio de seguridad: Es el objeto creado por el Directivo de Seguridad que permite a los usuarios, realizar las operaciones y visualizar los datos correspondientes a una colección de activos determinada, protegidos bajo una única autoridad.

2.6.1.5 Período de Evaluación: Es el objeto creado por el Responsable de Dominio, que permite establecer evaluaciones en distintos momentos al sistema de información que se analiza para conocer su evolución con respecto a los riegos.

2.6.1.6: Activos: Es el objeto que representa los recursos del sistema de información o relacionados con éste, necesarios para que la Organización funcione correctamente y alcance los objetivos propuestos por su dirección. Donde el usuario podrá:

- Identificar: Identificando los activos que conforman el sistema de información.
- Valorar: Asignando valores a activos individuales en cada dimensión de seguridad, teniendo en cuenta el interés de cada uno por su valía para la Organización. El valor puede ser propio, o puede ser acumulado. Se dice que los activos inferiores en un esquema de dependencias, acumulan el valor de los activos que se apoyan en ellos.
- Establecer la dependencia: Determinando la medida en que un activo superior se vería afectado por un incidente de seguridad en un activo inferior.
- Gestionar las opciones anteriores.

2.6.1.7 Amenazas: Es el objeto que representa las "cosas que ocurren" que pueden afectar a los activos y causar daños. Permite dar paso a la identificación y valoración de las amenazas que pueden afectar a cada activo:

- Identificación: Permite seleccionar o identificar la amenaza o grupo de amenazas posibles sobre los activos del sistema de información que se analiza.
- Valoración: Permite estimar cuán vulnerable es el activo, en dos sentidos:
  Degradación: cuán perjudicado resultaría el activo.
  Frecuencia: cada cuánto se materializa la amenaza.
- Gestionar las opciones anteriores.

2.6.1.8 Estimación del Impacto: Es el objeto que representa la medida del daño sobre el activo derivado de la materialización de una amenaza, sin tener en cuenta las salvaguardas actualmente desplegadas.

2.6.1.9 Impacto Acumulado: Es el objeto que se calcula para cada activo, por cada amenaza y en cada dimensión de valoración, siendo una función del valor acumulado y de la degradación causada. Es el que se calcula sobre un activo teniendo en cuenta:

- Su valor acumulado (el propio mas el acumulado de los activos que dependen de él).
- Las amenazas a que está expuesto.

2.6.1.10 Impacto Repercutido: Es el objeto que se calcula para cada activo, por cada amenaza y en cada dimensión de valoración, siendo una función del valor propio y de la degradación causada. Es el que se calcula sobre un activo teniendo en cuenta:

- Su valor propio.
- Las amenazas a que están expuestos los activos de los que depende.

2.6.1.11 Estimación del Riesgo: Este objeto es la medida del daño probable sobre un sistema. Conociendo el impacto de las amenazas sobre los activos, se puede derivar el riesgo sólo teniendo en cuenta la frecuencia de ocurrencia, pero no las salvaguardas actualmente desplegadas.

2.6.1.12 Riesgo Acumulado: Es el objeto que se calcula para cada activo, por cada amenaza y en cada dimensión de valoración, siendo una función del valor acumulado, la degradación causada y la frecuencia de la amenaza. Es el que se calcula sobre un activo teniendo en cuenta:

- El impacto acumulado sobre un activo debido a una amenaza y
- La frecuencia de la amenaza

El riesgo acumulado se calcula para cada activo, por cada amenaza y en cada dimensión de valoración, siendo una función del valor acumulado, la degradación causada y la frecuencia de la amenaza. Permite determinar las salvaguardas de que hay que dotar a los medios de trabajo: protección de los equipos, copias de respaldo, etc.

2.6.1.13 Riesgo Repercutido: Es el objeto que se calcula para cada activo, por cada amenaza y en cada dimensión de valoración, siendo una función del valor propio, la degradación causada y la frecuencia de la amenaza. Es el calculado sobre un activo teniendo en cuenta:

- El impacto repercutido sobre un activo debido a una amenaza y
- La frecuencia de la amenaza

2.6.1.14 Salvaguardas o contra medidas: Es el objeto que identifica los procedimientos o mecanismos tecnológicos que reducen el riesgo. Permiten hacer frente a las amenazas. El usuario podrá:

- Identificar salvaguarda: permite agregar o retirar del análisis aquellas salvaguardas que se consideren aplicables o no, respectivamente, para el nuevo proyecto.
- Valoración salvaguarda: Objeto que permite determinar la eficacia de las salvaguardas desplegadas, valorando su efectividad tomando en consideración: su idoneidad para el fin perseguido, la calidad de la implantación, la formación de los responsables de su configuración y operación, la formación de los usuarios, la existencia de controles de medida de su efectividad, la existencia de procedimientos de revisión regular.

2.6.19 Resultados: Este objeto permite interpretar los resultados anteriores de impacto y riesgo, así como establecer relaciones de prioridad por activos o grupos de activos, bien por orden de impacto o por orden de riesgo.

2.6.1.20 Informes: Es el elemento encargado de visualizar todos los informes que se encuentra en el sistema, identificados a lo largo del proceso de análisis y gestión de riesgos.

2.6.1.21 Perfil de Seguridad: Es el objeto que permite ver la posición de seguridad del sistema desde el punto de vista de un perfil dado (estándar de seguridad de la información). Una vez que el usuario interactúe con el perfil se hace una presentación (parcial) de las salvaguardas evaluadas, y un resumen de la cobertura de los objetivos o de los controles de seguridad en el perfil seleccionado.

2.6.2 Modelo de Dominio.

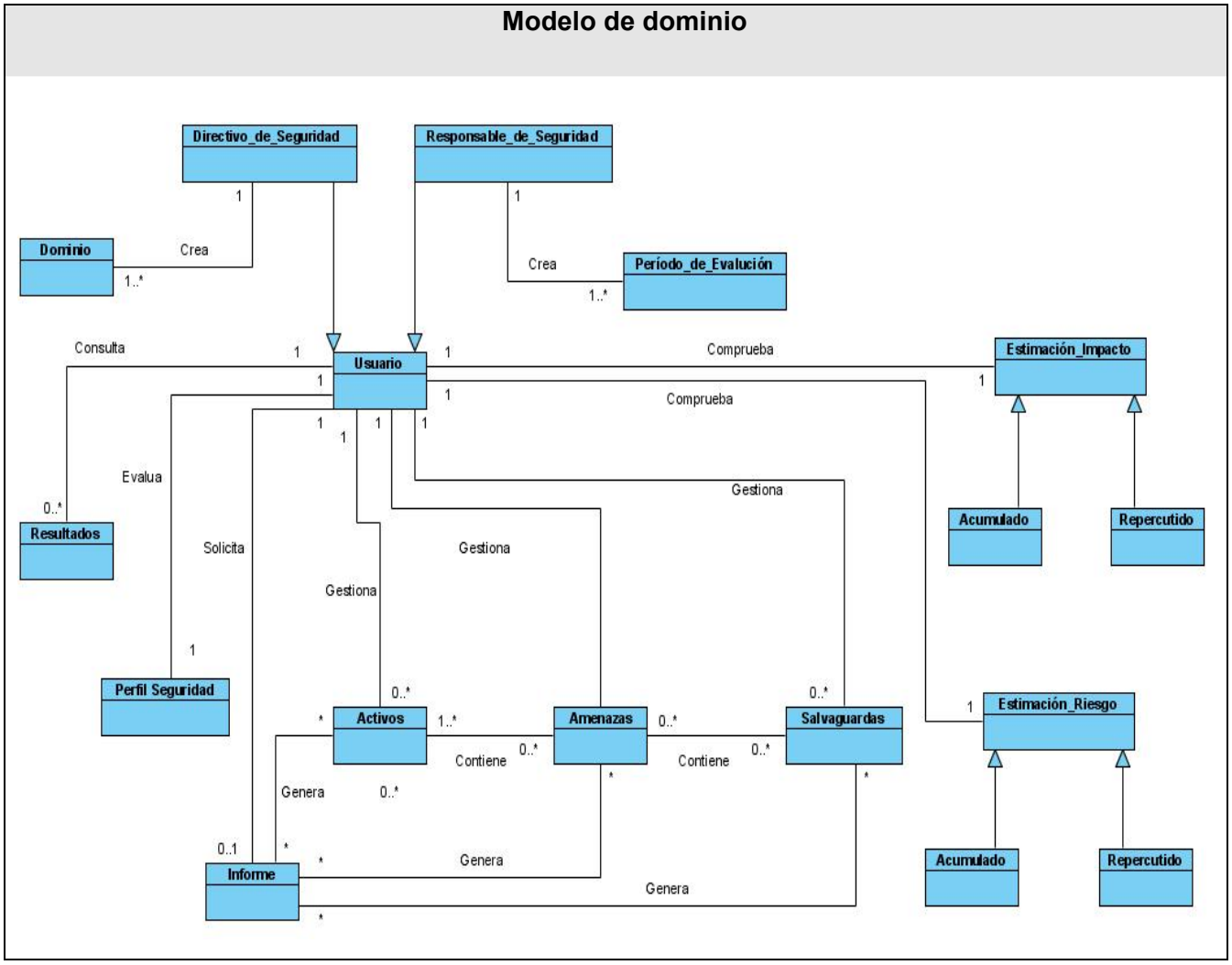


Figura 2.1 Diagrama de Modelo de Dominio.

2.3 Especificación de los requisitos de software.

Los requerimientos son una representación documentada de las condiciones o capacidades que debe alcanzar un sistema o componente de un sistema para satisfacer un contrato, estándar, u otro documento impuesto formalmente, que tiene como propósito, establecer un entendimiento común entre el usuario y el proyecto de software sobre los requisitos que solicita el usuario, los cuales serán abordados en dicho proyecto.

Se agrupan en dos grandes categorías:

- Requerimientos funcionales.
- Requerimientos no funcionales.

A continuación se muestran los requerimientos funcionales y no funcionales definidos para la realización y desarrollo del sistema que se propone. La distribución de estos requerimientos se ha definido a partir de las funcionalidades que componen dicho sistema y de los servicios que este debe ofrecer.

2.3.1 Requerimientos funcionales.

Los requerimientos funcionales son los encargados de especificar las acciones que el sistema debe ser capaz de realizar, sin que se tome en consideración ningún tipo de restricción física. Por lo que se hace una definición clara y libre de ambigüedades.

RF 1. Autenticar usaurio.
RF 2. Cargar Configuracion.
RF 3. Gestionar usuario.
RF 4. Registrar Dominio de Seguridad.
RF 5. Modificar Dominio.
RF 6. Eliminar Dominio.
RF 7. Ver detalles del Dominio.
RF 8. Adicionar Período de Evaluación.
RF 9. Mostrar Período de Evaluación.
RF 10. Modificar Período de Evaluación.
RF 11. Eliminar Período de Evaluación.
RF 12. Ver Detalles del Período de Evaluación.
RF 13. Adicinar Activo.
RF 14. Modificar Activo.
RF 15. Buscar Activo.
RF 16. Ver Detalles de Activos.
RF 17. Eliminar Activo.
RF 18. Establecer Dependencia de Activo.
RF 19. Valorar Activo.
RF 20. Identificar Amenazas.

RF 21. Valorar Amenazas.
RF 22 Ver Detalles de Amenaza.
RF 23 Modificar amenaza.
RF 24. Identificar Salvaguardas.
RF 25. Valorar Salvaguardas.
RF 26. Calcular Impacto Potencial Acumulado.
RF 27. Calcular Impacto Potencial repercutido.
RF 28. Calcular Riesgo Potencial Acumulado.
RF 29. Calcular Riesgo Potencial Repercutido.
RF 30. Calcular Impacto Acumulado Residual.
RF 31. Calcular Impacto Repercutido Residua.l
RF 32. Calcular Riesgo Acumulado Residual.
RF 33. Calcular Riesgo Repercutido Residual.
RF 34. Interpretar Resultados.
RF 35. Evaluar Perfil de Seguridad.
RF 36. Generar Informe.
RF 37. Cerrar sección.

2.3.2 Requerimientos no funcionales.

Los requerimientos no funcionales son propiedades o cualidades que debe tener el producto. Constituyen las características que hacen a dicho producto atractivo, usable, rápido y confiable. Normalmente están vinculados a requerimientos funcionales y forman una parte significativa de la especificación. Son importantes para la determinación entre un producto bien aceptado y uno con poca aceptación, por lo que se consideran esenciales en el éxito del producto.

2.3.2.1 Usabilidad.

El sistema podrá ser usado desde cualquier plataforma o entorno de usuario que soporte el ambiente web.

2.3.2.3 Rendimiento

Se debe garantizar que el tiempo de respuesta del sistema ante las solicitudes hechas por los usuarios para cada acción a realizar, sea el mínimo posible de 2 segudos.

2.3.2.3 Soporte.

La aplicación está basada en la metodología MAGERIT, la cual está sustentada en las siguientes normas y estándares: ISO 27002, SP-80, ISO/IEC 13335-1:2004, ISO/IEC TR 13335-3:1998, ISO/IEC TR 13335-4:2000, ISO/IEC TR 18044:2004.

2.3.2.4 Portabilidad

Se podrá acceder al servicio desde cualquier PC conectada a la red que disponga de un navegador web. En la actualidad, la mayoría de los entornos de usuario tienen una aplicación para esta función.
El sistema está diseñado para soportar su funcionalidad tanto en sistemas operativos libres como propietarios.

2.3.2.5 Seguridad y Privacidad

Se establecerá un acceso limitado a usuarios determinado por el rol dentro de la organización, para evitar acceso no autorizado a los servicios.

2.3.2.6 Software

- Se requiere tener instalado el Navegador: Mozilla Firefox 2.0 o superior o Internet Explorer 6.0 o superior.
- Se utilizará como servidor web Apache 2.0.
- Utilizará como base de datos PostgreeSQL.

2.3.2.7 Hardware

Requiere como mínimo de RAM 512 MB.
El disco duro requiere como mínimo 3 GB para almacenar la Base de Datos.

2.3.2.8 Restricciones en el diseño y la implementación.

Lenguaje de programación PHP.
Librería ADOdb.

2.3.2.9 Legales

La aplicación y toda la documentación generada pertenecen a la Dirección de Seguridad Informática UCI y la Universidad de las Ciencias Informáticas.

2.3.2.10 Confiabilidad

La aplicación debe estar disponible las 24 horas de forma tal que se pueda acceder a todas sus funcionalidades.

2.3.2.11 Interfaz.

- Se ajusta a los estándares establecidos para el desarrollo de un buen diseño.
- Es simple y de fácil uso para que el usuario no tenga dificultad al interactuar con el mismo.
- Trata de que la aplicación sea lo más iterativa posible.
- Está diseñada para que el usuario pueda ir de un punto a otro con gran facilidad dentro de ella.

2.3.2.12 Ayuda.

El sistema deberá contar con una ayuda que sea capaz de auxiliar y guiar al usuario.

2.4 Modelo de Casos de Uso del Sistema.

Los casos de uso constituyen una técnica narrativa para describir el comportamiento del sistema y sus funcionalidades. Cada caso de uso puede describir una o más funcionalidades requeridas para el sistema por parte del usuario.

2. 4.1 Definición de los actores del sistema a automatizar.

En la siguiente tabla se muestra una descripción del rol que desempeña cada uno de los actores del sistema.

Actores	**Justificación**
Directivo de Seguridad.	Es la persona encargada de administrar los usuarios del sistema y asignarles el rol que desempeñarán. Es el responsable de gestionar y llevar el control de los resultados y la información proyectados en cada dominio de la Organización y de darle soporte al sistema. Requerimientos funcionales asociados: RF3, RF4, RF5, RF6, RF7.
Responsable de Seguridad.	Es la persona que responde por la seguridad informática en cada área a la que pertenece y quien se encarga de

	interactuar con el sistema y gestionar los datos referentes a su dominio. Requerimientos funcionales asociados: RF8, RF9, RF10, RF11, RF12, RF 13, RF 14, RF 15, RF 16, RF 17, RF 18, RF 18, RF 19, RF 20, RF 21, RF 22, RF 23, RF 24, RF 25, RF 26, RF 27, RF 28, RF 29, RF 30, RF 31, RF 32, RF 33. RF 34, RF 35, RF 36.
Usuario.	Es aquel que solicita entrar al sistema y seguidamente se le permite el acceso a determinadas funcionalidades del sistema que solo correponden realizar a este, según su rol establecido previamente por el Directivo Seguridad Informática. Como Usuario tiene acceso a las siguientes funciones: Mostar Perfil e Imprimir los informes resultantes. Requerimientos funcionales asociados: RF1, RF 37.

Tabla 2.2 Definición y descripción de los actores del sistema.

2.4.2 Diagrama de caso de uso del sistema a automatizar.

En la figura 2.2 se muestra una representación gráfica de los casos de uso del sistema así como su relación con los actores del sistema, los cuales son los encargados de inicializar e interactuar con cada caso de uso.

Diagrama de Paquetes de Casos de Uso del Sistema.

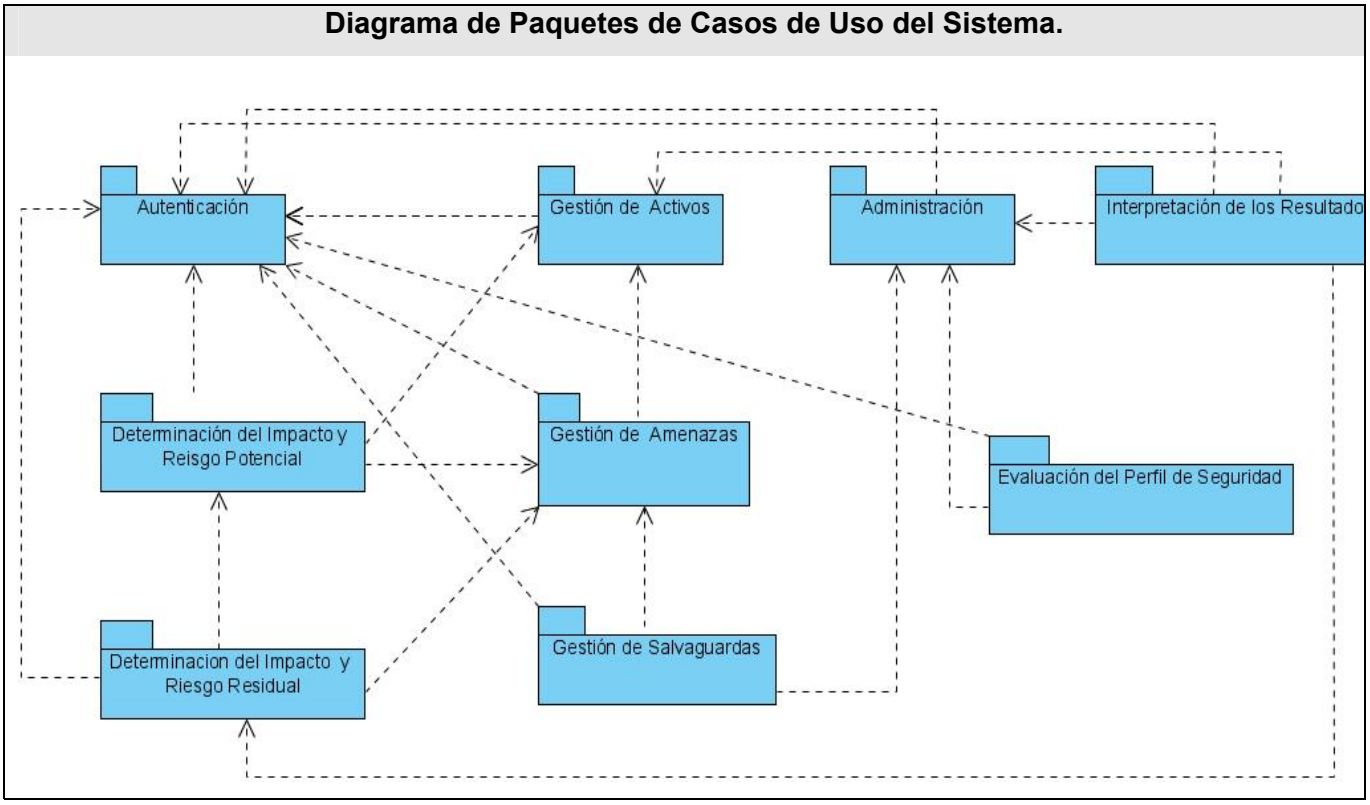


Figura 2.2 Diagrama de Paquete de Casos de Uso del Sistema.

Diagrama de Caso de Uso del Sistema.

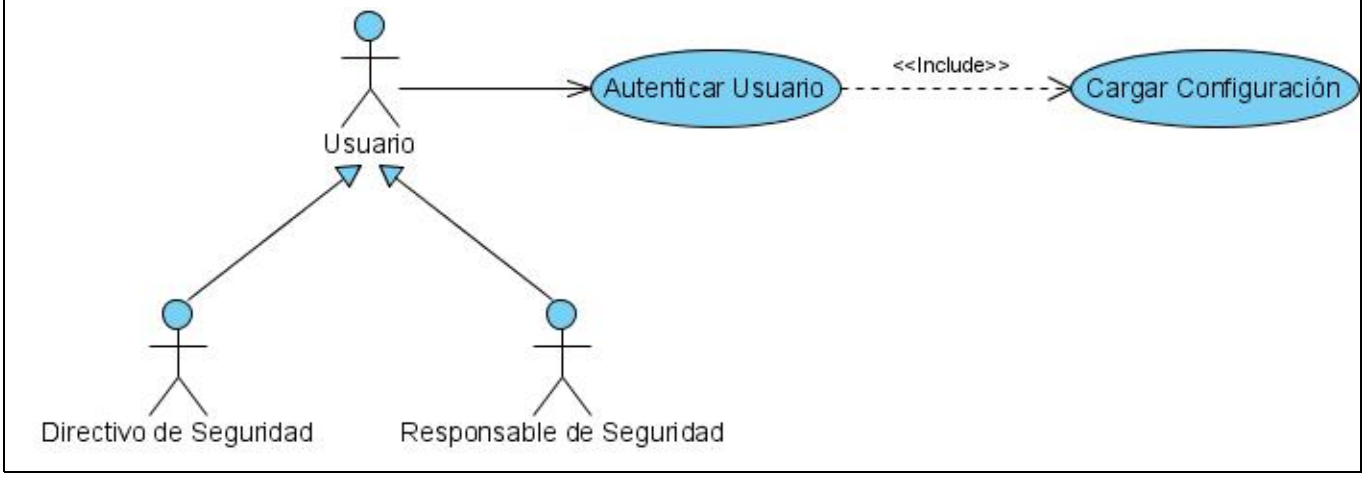


Figura 2.3 Diagrama de Casos de Uso del Sistema. Paquete de Autenticación.

Diagrama de Caso de Uso del Sistema.

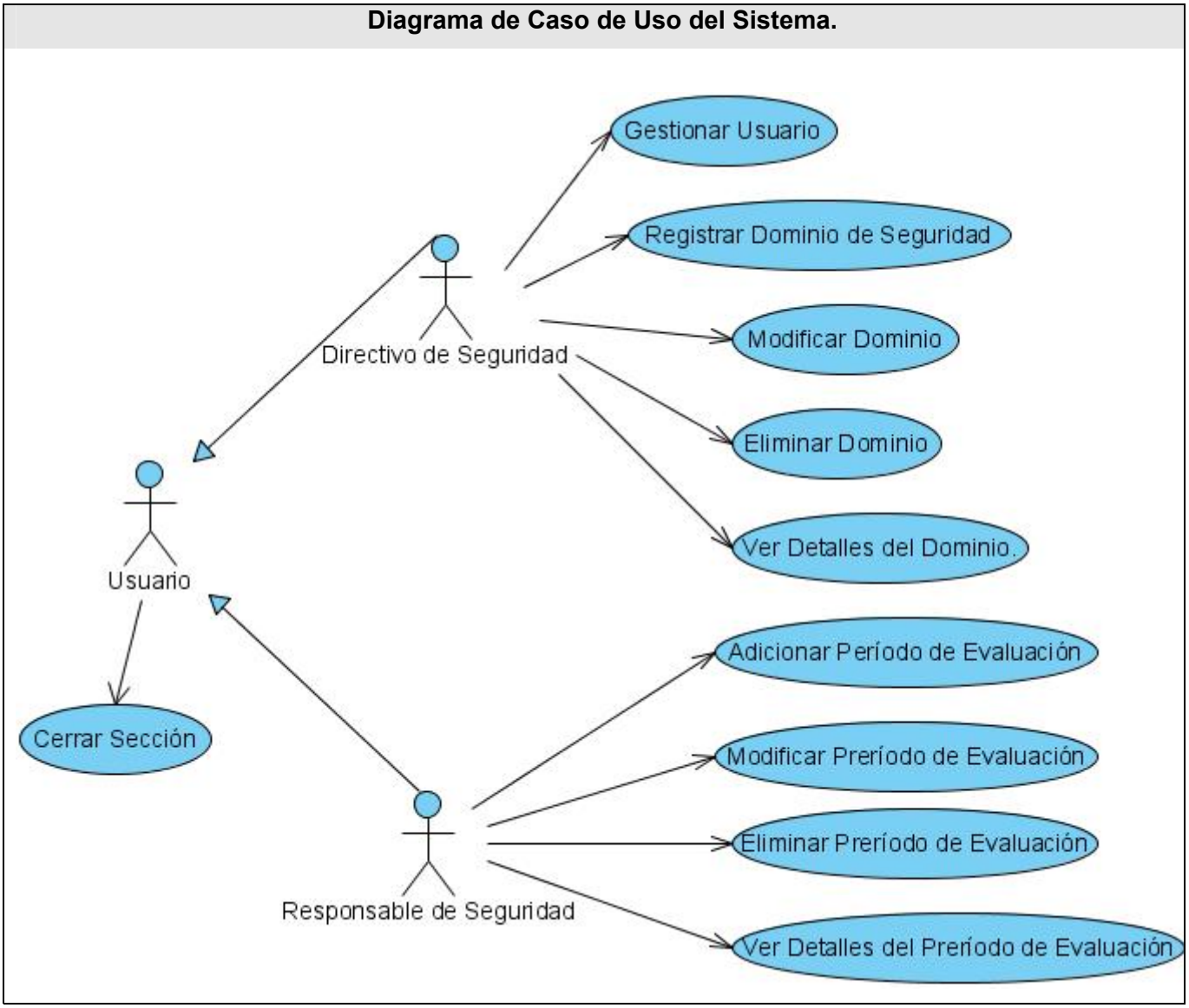


Figura 2.4 Diagrama de Casos de Uso del Sistema. Paquete de Administración.

Diagrama de Caso de Uso del Sistema.

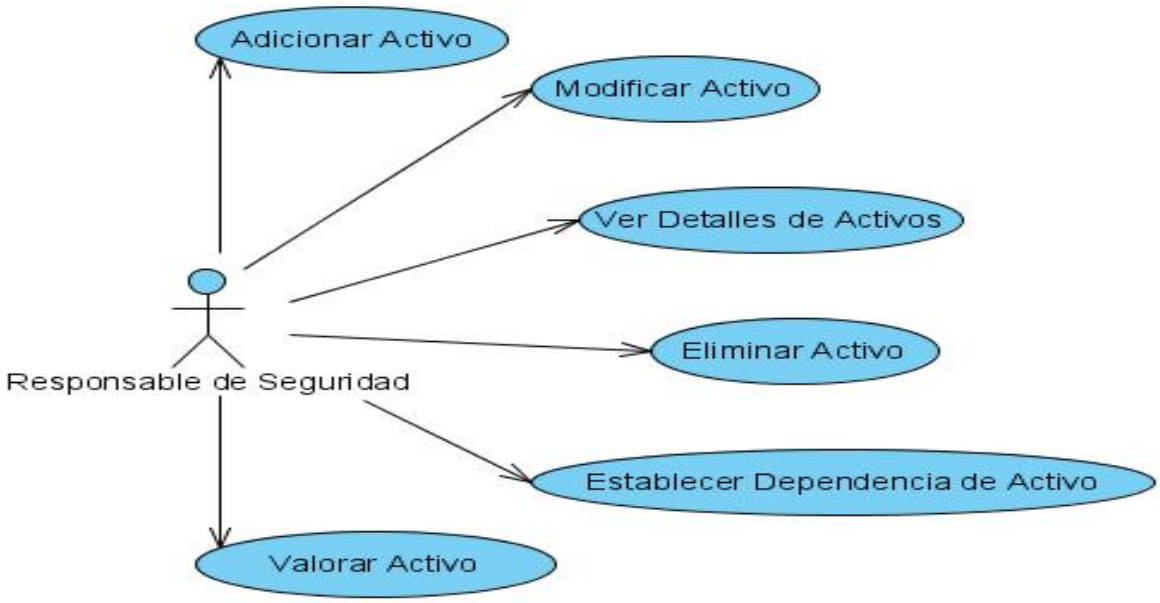


Figura 2.5 Diagrama de Casos de Uso del Sistema. Paquete Gestión de Activos.

Diagrama de Caso de Uso del Sistema.

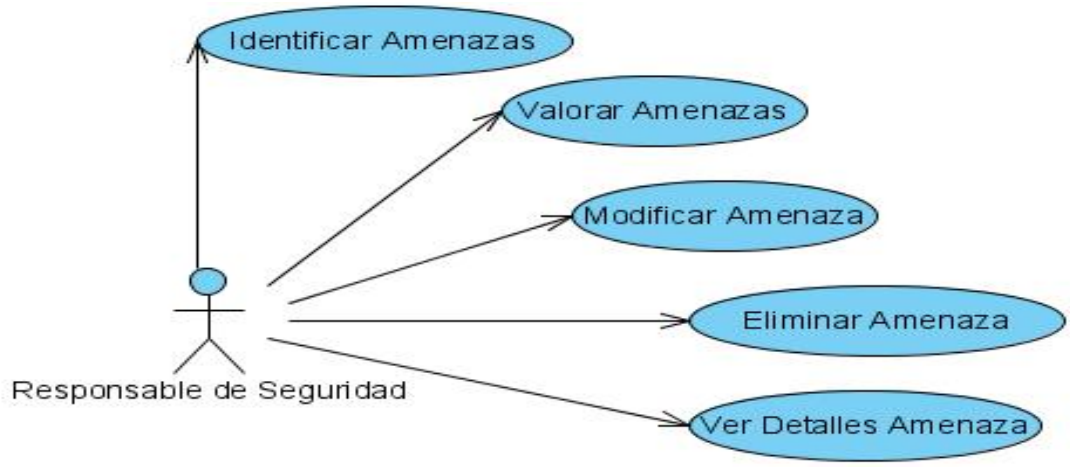


Figura 2.6 Diagrama de Casos de Uso del Sistema. Paquete Gestion de Amenazas .

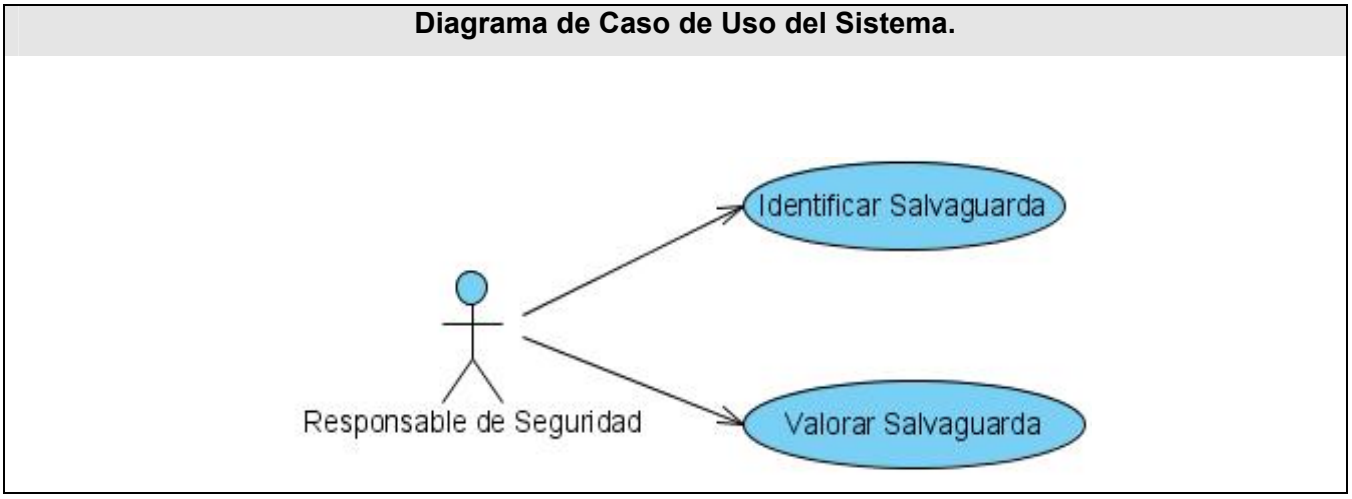


Figura 2.7 Diagrama de Casos de Uso del Sistema. Paquete Gestión de Salvaguardas.

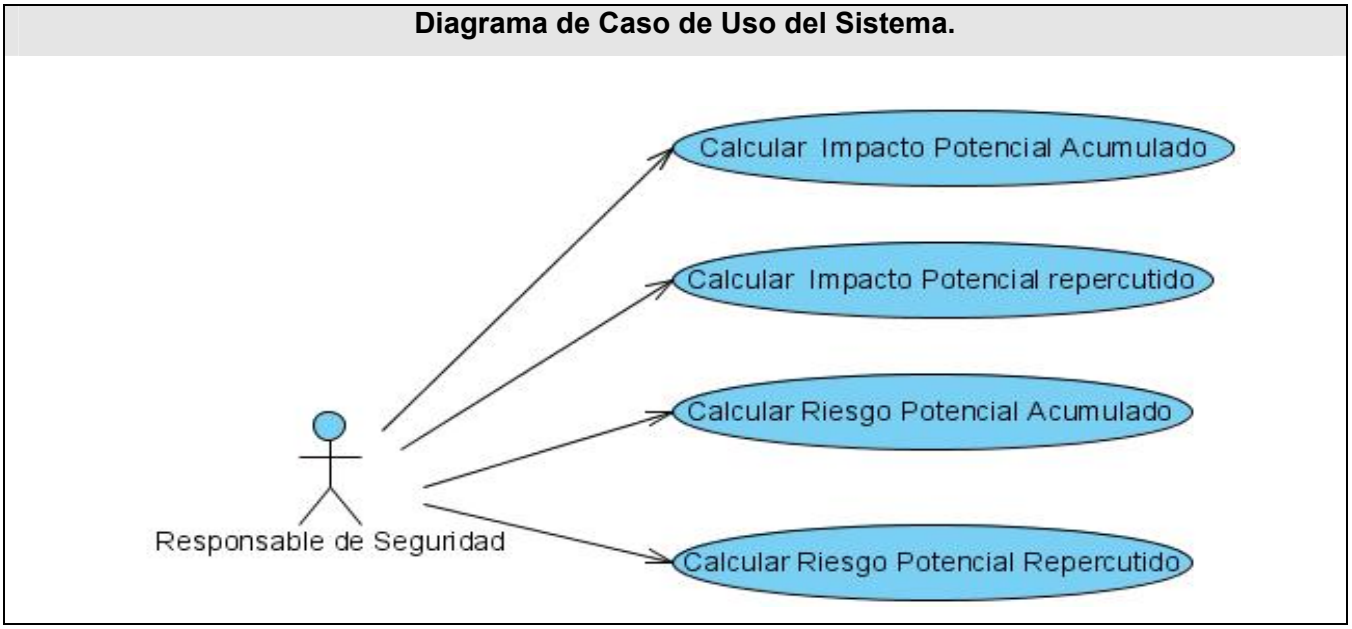


Figura 2.8 Diagrama de Casos de Uso del Sistema. Paquete Determinación de Impacto y Riesgo Potencial.

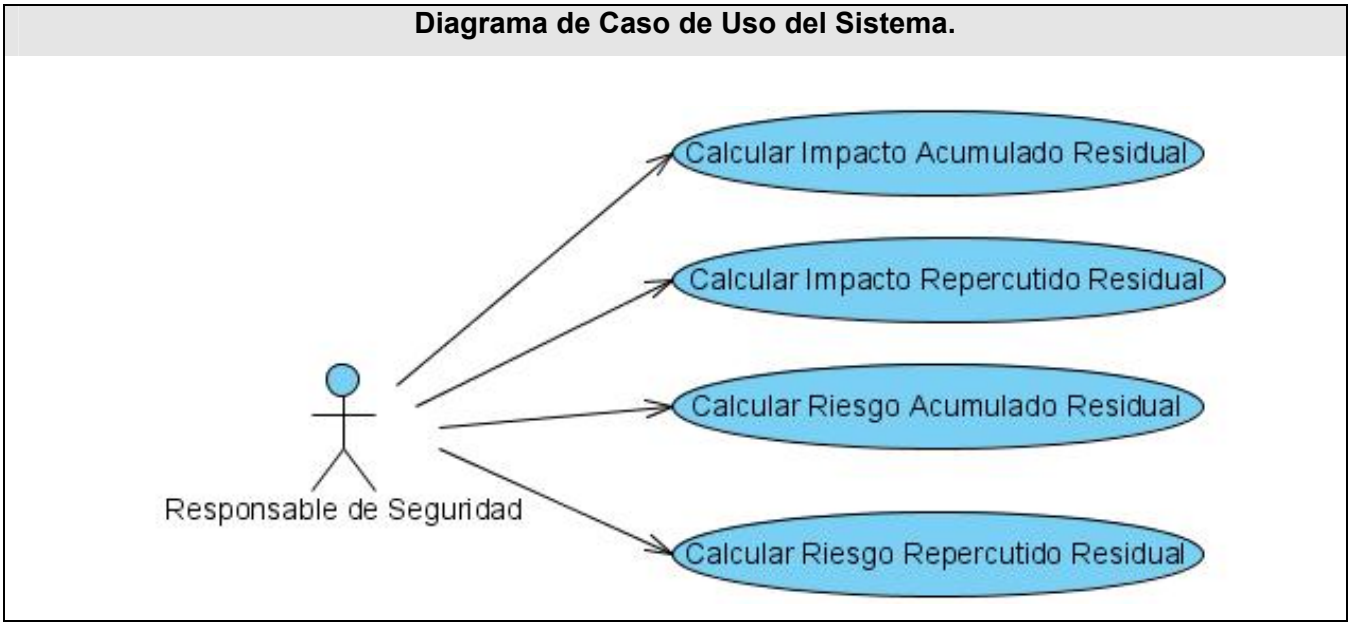


Figura 2.9 Diagrama de Casos de Uso del Sistema. Paquete Determinación de Impacto y Riesgo Residaul.

Figura 2.10 Diagrama de Caso de Uso del Sistema. Paquete Interpretación de Resultados.

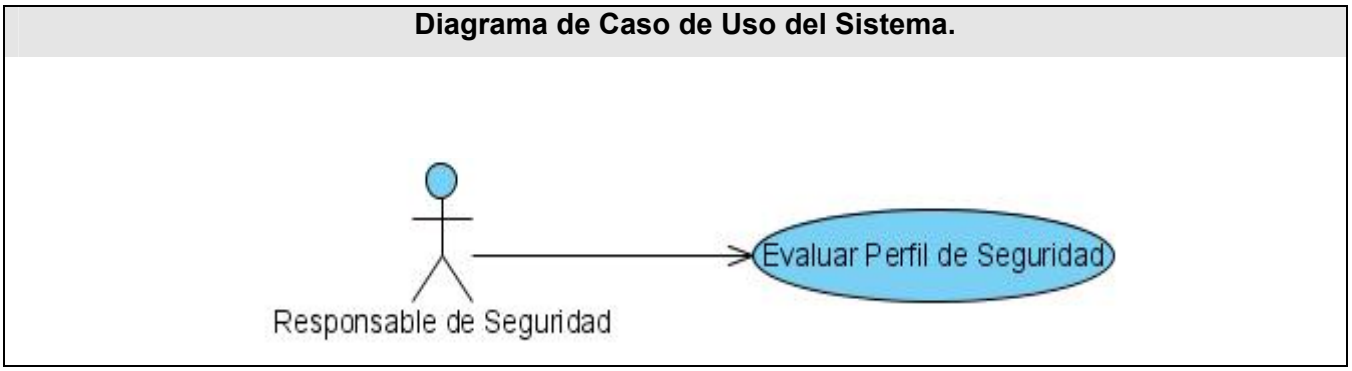


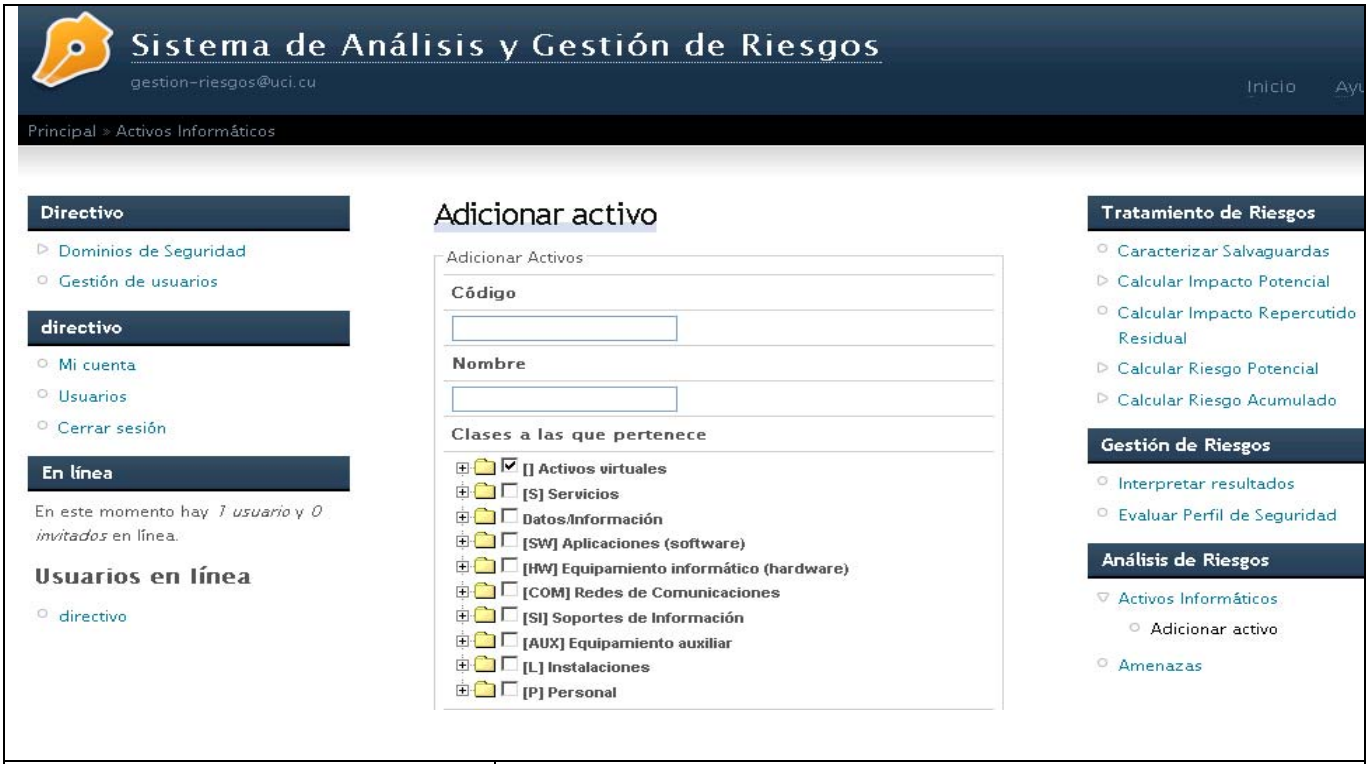
Figura 2.11 Diagrama de Casos de Uso del Sistema. Paquete Evaluación del Perfil de Seguridad.

2.4.3 Descripción de los Casos de uso.

En las siguientes tablas se realiza una descripción detallada de los casos de uso críticos del sistema, además de mostrar en secciones una secuencia de las acciones realizadas por los actores y las respuestas del sistema para cada una de las funcionalidades del caso de uso. Cada tabla puede tener una o más secciones en dependencia de la complejidad de cada caso de uso.

<table>
<tr><th colspan="3">CU-1 Adicionar Activo</th></tr>
<tr><td>Actores</td><td colspan="2">Responsable de Seguridad (inicia).</td></tr>
<tr><td>Propósito</td><td colspan="2">Reconocer los activos que componen los procesos.</td></tr>
<tr><td colspan="3">Resumen: El Responsable de Dominio selecciona la opción “Adicionar activos”. Se muestra un formulario con los datos que se deben insertar. El Responsable de Dominio introduce los datos especificados. Se almacenan los datos en la BD.</td></tr>
<tr><td>Referencias</td><td colspan="2">RF13.</td></tr>
<tr><td>Precondiciones</td><td colspan="2"></td></tr>
<tr><td colspan="3">Flujo Normal de Eventos.</td></tr>
<tr><td>Acciones del Actor</td><td colspan="2">Respuesta del Sistema</td></tr>
<tr><td>1. El Responsable de Seguridad selecciona la opción “Adicionar activos”.</td><td colspan="2">2. Obtiene de la BD los activos existentes.</td></tr>
<tr><td></td><td colspan="2">3. Muestra el formulario Adicionar activos.</td></tr>
<tr><td>4. El Responsable de Seguridad introduce o selecciona:
Nombre.
Código.
Clases a la que pertenece.
Capa organizativa.
Descripción.
Tipo de clase.
Ubicación.
Cantidad (si procede).
Descripción.
Otras caracterísitcas.</td><td colspan="2"></td></tr>
<tr><td>5. El Responsable de Seguridad acciona el botón</td><td colspan="2">6. Comprueba que se han introducido los datos</td></tr>
</table>

"Aceptar".	obligatorios. • Nombre. • Código. • Clases a la que pertenece. • Capa organizativa. • Tipo de clase.
	7. Comprueba que los datos introducidos son correctos. • El nombre solo admite letras.
	8. Comprueba que el código y nombre del activo no esté registrado en la BD.
	9. El sistema registra la información en su base de datos.
Flujos Alternos	
6. a Campos Obligatorios.	
	6. a.1 Comprueba que se hayan introducido los campos obligatorios.
	6. a.2 Muestra un mensaje indicando los campos obligatorios.
7. a Datos Introducidos Incorrectos	
	7. a.1. Comprueba que los datos introducidos son correctos.
	7. a.2 Indica los datos que están incorrectos.
8. a Datos repetidos	
	8. a.1 Comprueba si el activo está registrado en la BD usando el mismo nombre o código.
	8. a.2 Muestra un mensaje indicando que el activo se encuentra registrado.



Poscondiciones	

Tabla 2.3 Descripción del CUS "Adicionar Activo".

CU-2 Modificar Activo	
Actores	Responsable de Dominio (inicia).
Propósito	Modificar los datos de un activo.
Resumen: El Responsable de Seguridad selecciona el elemento que desea modificar. Usando el código del activo seleccionado se muestra un formulario con los datos asociados al activo de forma editable. El Responsable de Seguridad modifica los datos. Se actualizan los datos en la BD.	
Referencias	RF14.
Precondiciones	
Flujo Normal de Eventos.	
Acciones del Actor	**Respuesta del Sistema**
1. El Responsable de Seguridad selecciona	2. Obtiene de la BD, usando el código del activo

la opción “Modificar activo”.	seleccionado, los siguientes datos: • Nombre. • Código. • Tipo de clase. • Capa organizativa. • Descripción.
	3. Muestra el formulario Modificar activos.
4. El Responsable de Seguridad selecciona el activo y modifica los datos necesarios.	
5. El Responsable de Seguridad acciona el botón Aceptar.	6. Comprueba que se han introducido los datos obligatorios.
	7. Comprueba que los datos introducidos son correctos. • El nombre solo admite letras.
	8. Comprueba que el código y nombre del activo no esté registrado en la BD.
	9. El sistema registra la información en su base de datos.
Flujo Alterno	
6. a Campos Obligatorios.	
	6. a.1 Comprueba que se hayan introducido los campos obligatorios.
	6. a.2 Muestra un mensaje indicando los campos obligatorios.
7. a Datos Introducidos Incorrectos	
	7. a.1 Comprueba que los datos introducidos son correctos.
	7. a.2 Indica los datos que están incorrectos.
8. a Datos repetidos	
	8. a.1 Comprueba si el activo está registrado en la BD usando el mismo nombre o código.
	8. a.2 Muestra un mensaje indicando que el activo se

	encuentra registrado.
	
Poscondiciones	

Tabla 2.4 Descripción del CUS "Modificar Activo".

CU-3 Ver Detalles de Activo.		
Actores	Responsable de Seguridad.	
Propósito	Mostrar los detalles de un activo.	
Resumen: Obtiene de la BD los datos del activo seleccionado. Muestra los datos obtenidos.		
Referencias	RF16	
Precondiciones		
Flujo Normal de Eventos.		
Acciones del Actor	**Respuesta del Sistema**	
1. El Responsable de Seguridad selecciona un activo para ver sus detalles.	2. Obtiene de la BD, usando el código del activo seleccionado, los siguientes datos: • Nombre • Descripción	

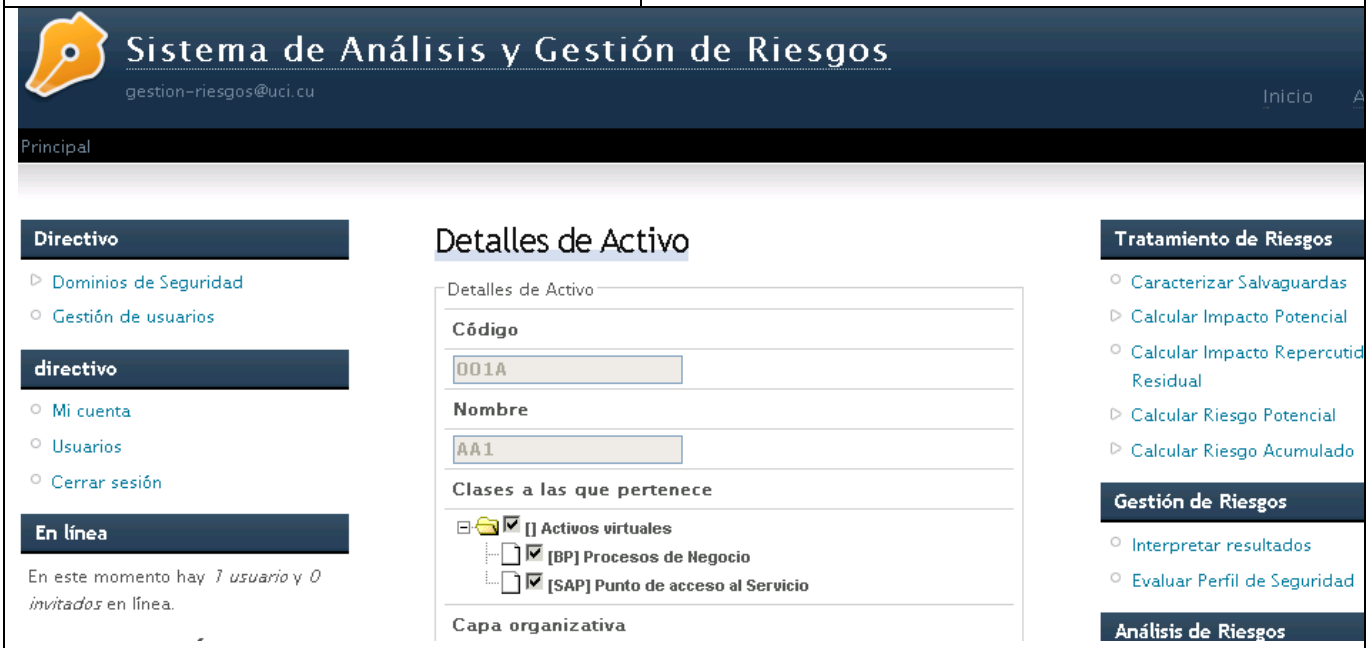
	• Tipo de clase • Capa organizativa • Código • Valor acumulado • Listado de amenazas.
	3. Muestra los datos obtenidos.
	
Poscondiciones	

Tabla 2.5 Descripción del CUS "Ver Detalles de Activo".

CU-4 Eliminar Activo	
Actores	Responsable de Seguridad
Propósito	Eliminar un activo del sistema.
Resumen: El caso de uso se inicia Responsable de Seguridad selecciona un activo para eliminar. Se le muestra un mensaje de confirmación de eliminación. El Responsable de Dominio acepta la eliminación. Se eliminan de la BD el activo seleccionado.	
Referencias	RF17

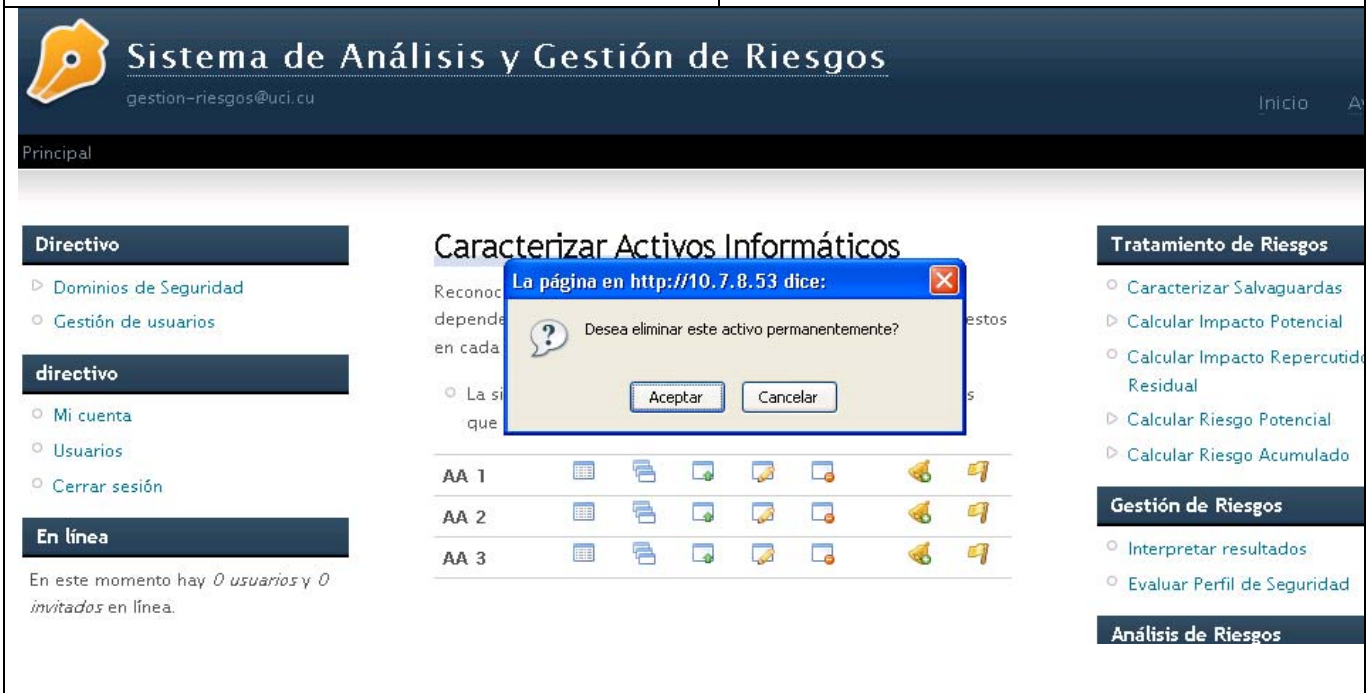
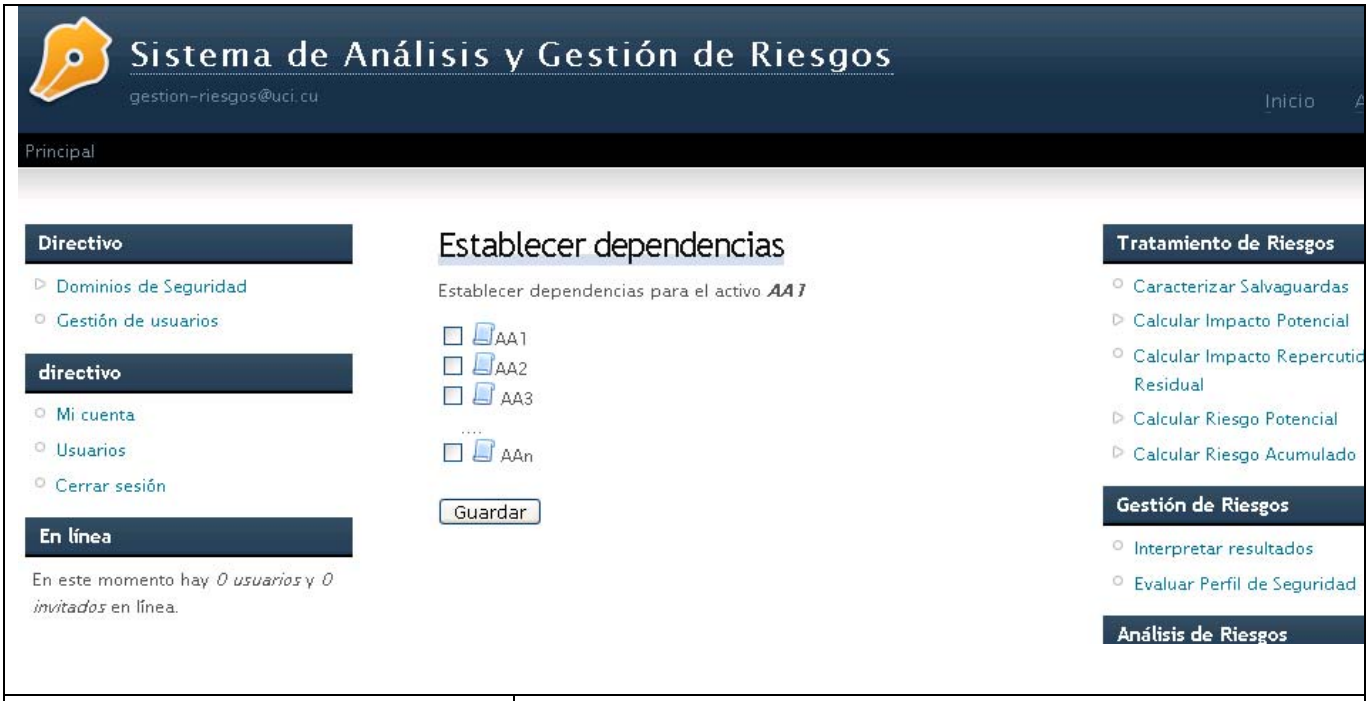
Precondiciones	
Flujo Normal de Eventos.	
Acciones del Actor	**Respuesta del Sistema**
1. El Responsable de Seguridad selecciona la opción “Eliminar” del activo que se desea eliminar.	2. Muestra un mensaje de confirmación de eliminación.
3. El Responsable de Seguridad selecciona la opción “Aceptar”.	4. Elimina en la BD los datos asociados al activo seleccionado usando el código del mismo.
	5. Actualiza la lista de activos mostrada.
Flujo Alterno	
3. a Opción “Cancelar”	
3. a.1 El Responsable de Seguridad selecciona la opción “Cancelar”.	3. a.2 Cierra el mensaje mostrado.
	
Poscondiciones	

Tabla 2.6 Descripción del CUS “Eliminar Activo”.

CU-5 Establecer Dependencia de Activo	
Actores	Responsable de Dominio.
Propósito	Identificar la dependencia entre activos.

Resumen: El caso de uso se inicia cuando el Responsable de Seguridad selecciona la opción "Establecer Dependencia". El sistema, muestra el listado de los tipos de activos que pueden tener dependencias con uno previamente seleccionado y agregar una nueva dependencia entre activos.	
Referencias	RF18.
Precondiciones	
Flujo Normal de Eventos.	
Acciones del Actor	**Respuesta del Sistema**
1. El Responsable de Dominio selecciona la opción "Establecer Dependencias".	2. Obtiene de la BD el listado de activos que pueden tener dependencia con el activo seleccionado, teniendo en cuenta el tipo de clase de activos identificados con los que puede establecer dependencia el activo seleccionado.
	4. Muestra el formulario "Establecer Dependencias".
5. El Responsable de Seguridad selecciona el(los) activo(s) para establecer la dependencia.	
11. El Responsable de Seguridad acciona el botón Guardar.	6. Agregar a la lista de activos dependientes la selección realizada por el usuario.
	7. Muestra la lista de los activos seleccionados.
	13. Almacena los datos introducidos en la BD.
	14. Muestra mensaje indicando que la operación se realizó con éxito.
Flujo Alterno	
2. a No Existen activos	
	2. a.2. Muestra un mensaje indicando que no existen activos con esos criterios de búsqueda.



Poscondiciones	No aplica

Tabla 2.7 Descripción del CUS "Establecer Dependencia entre Activos".

CU-6 Valorar Activo	
Actores	Responsable de seguridad.
Propósito	Identificar en que dimensión es valioso el activo.
Resumen: El caso de uso se inicia cuando el responsable de Seguridad selecciona la opción "Valorar Activo". El sistema, muestra el formulario que permite valorar el activo seleccionado en una dimesion específica, asi como introducir un comentario que justifica el por qué de la valoración. El responsable de seguridad acciona el botón Aceptar para guardar el valor establecido. El sistema calcula el valor acumulado del activo seleccionado y guarda en la BD los valor propio asignado por al actor y el valor acumulado, determinado por el mayor valor entre el propio y el de cualquiera de sus superiores que este soporta, finalizando asi el caso de uso.	
Referencias	RF19.
Precondiciones	Identificar activo. Establecer dependecia del activo.
Flujo Normal de Eventos.	
Acciones del Actor	**Respuesta del Sistema**
1. El Responsable de Seguridad selecciona la opción	3. Muestra el formulario Valor activo.

"Valorar Activo".	
4. El Responsable de Seguridad introduce: • Dimensión: Disponibilidad. Integridad de los datos. Confiabilidad de los datos. Autenticidad de los usuarios del servicio. Autenticidad del origen de los datos. Trazabilidad del servicio. Trazabilidad de los datos. • Valor. 10 muy alto. 7 - 9 altos. 4 - 6 medio. 1 - 3 bajo. 0 despreciable. • Comentario.	
5. El Responsable de Seguridad acciona el botón Aceptar.	6. Comprueba que se han introducido los datos obligatorios.
	7. Obtiene de la BD los valores propios de los activos superiores que dependen del seleccionado.
	8. Calcula el valor acumulado del activo; determinado por el mayor valor entre el propio y el de cualquiera de sus superiores que este soporta.
	9. Almacena los datos en la BD.
Flujo Alterno.	
6. a Campos Obligatorios.	
	6. a.1 Comprueba que se hayan introducido los campos obligatorios.
	6. a.2 Muestra un mensaje indicando los

	campos obligatorios.
	
Poscondiciones	

Tabla 2.8 Descripción del CUS "Valorar Activo".

CU-7 Identificar Amenazas	
Actores	Responsable de Seguridad (inicia).
Propósito	Identificar las amenazas relevantes sobre cada activo.
Resumen: El caso de uso se inicia cuando el Responsable de Seguridad selecciona la opcion "Identificar Amenazas". El sistema muestra un formulario que permite establecer una nueva amenaza o retirar aquellas que se han identificado previamente para el activo seleccionado, adiciona la nueva amenaza identificada a la lista de amenazas previas y finalmente almacena los nuevos datos.	
Referencias	RF20.
Precondiciones	Debe estar identificado el activo.
Flujo Normal de Eventos.	
Acciones del Actor	**Respuesta del Sistema**
1. El Responsable de Dominio selecciona la opción "Identificar Amenazas".	2. Muestra el formulario identificar amenazas para el activo seleccionado.

3. El Responsable de Seguridad selecciona el(los) tipo(s) de amenaza(s) deseada(s).	
4. El Responsable de Seguridad acciona el botón "Guardar".	5. Adiciona la(s) nueva(s) amenaza(s).al listado de amenazas identificadas para el activo seleccionado.
	6. Almacena los datos en la BD.
Flujo Alterno	
4. a Operación "Cancelar".	
4. a.1 El Responsable de Seguridad acciona el botón "Cancelar".	4. a.2 Elimina la nueva selección hecha por el usuario.
5. a Datos repetidos	
	5. a.1 Comprueba si se ha identificado previamente la amenaza seleccionada
	5. a.2 Muestra un mensaje indicando que la amenaza se encuentra identificada.

"Interfaz Identificar amenazas".

Poscondiciones	

Tabla 2.9 Descripción del CUS "Identificar amenazas".

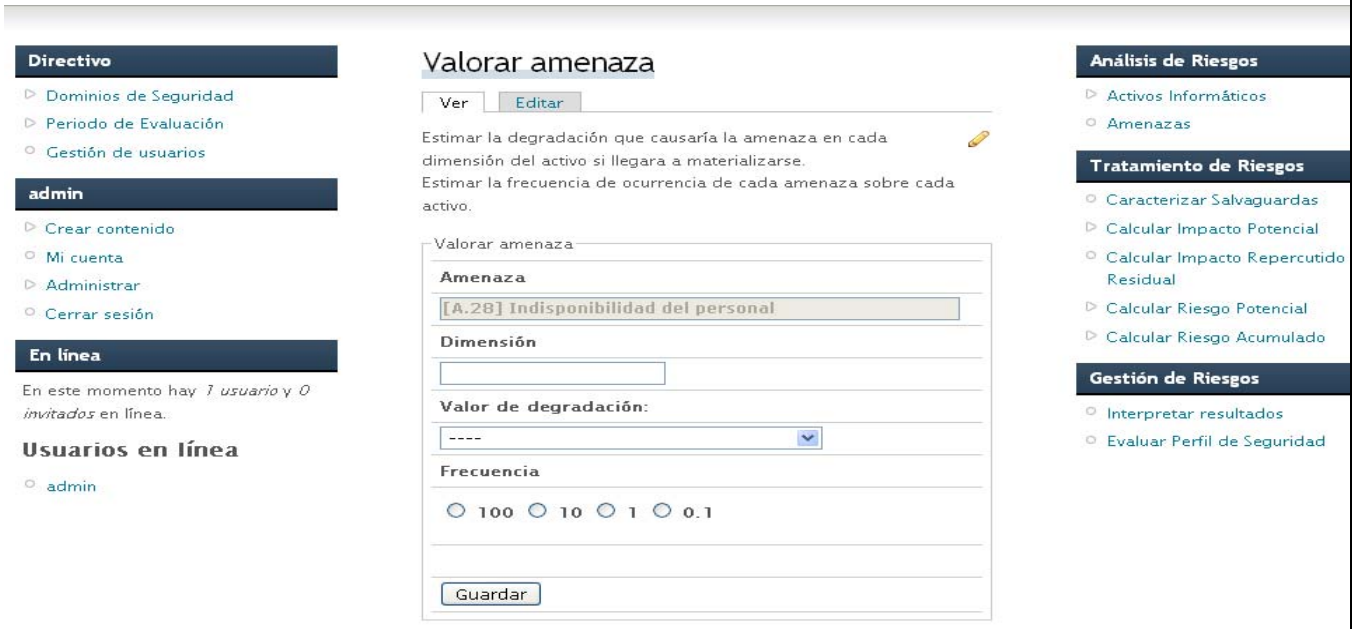
CU-8 Valorar Amenazas	
Actores	Responsable de Seguridad (inicia).

Propósito	Estimar la degradación que causaría la amenaza en cada dimensión del activo si llegara a materializarse. Estimar la frecuencia de ocurrencia de cada amenaza sobre cada activo.
Resumen: El caso de uso se inicia cuando el Responsable de Seguridad selecciona la opción "Valorar Amenazas". El sistema muestra un formulario que permite modificar el valor de las amenazas previamente identificadas para un activo. El actor especifica los valores: diemension y valor y ordena actualizar en la BD los nuevos valores insertados. El sistema actualiza la BD y la lista de amenazas mostradas.	
Referencias	RF21.
Precondiciones	Debe estar identificada la amenaza.

Flujo Normal de Eventos.	
Acciones del Actor	**Respuesta del Sistema**
1. El Responsable de Dominio selecciona la opción "Valorar Amenazas".	2. Muestra el formulario Valorar Amenazas.
5. El Responsable de Seguridad introduce: • Dimensión. Disponibilidad. Integridad de los datos. Confiabilidad de los datos. Autenticidad de los usuarios del servicio. Autenticidad del origen de los datos. Trazabilidad del servicio. Trazabilidad de los datos. • Valor de degradación: 0% insignificante. 1% impacto pequeño. 10% impacto apreciable. 90% el activo casi todo su valor. 100% el activo pierde todo su valor. • Frecuencia (100/10/1/0.1).	
6. El Responsable de Seguridad acciona el botón	7. Comprueba que se han introducido los

Guardar	datos obligatorios.
	8. Almacena los datos en la BD.
	9. Actualiza la lista de amenazas mostradas.
Flujo Alterno.	
7. a Campos Obligatorios.	
	7. a.1 Comprueba que se hayan introducido los campos obligatorios.
	7. a.2 Muestra un mensaje indicando los campos obligatorios.

"Interfaz Valorar Amenazas"



Poscondiciones	

Tabla 2.10 Descripción del CUS "Valorar amenazas".

CU-9 Ver Detalles de Amenaza	
Actores	Responsable de Seguridad
Propósito	Mostrar una lista de activos relacionados con la amenaza.
Resumen: El caso de uso se inicia Responsable de Seguridad selecciona la opción Ver Detalles de Amenaza. El sistema muestra una lista de los activos.	

Referencias	RF22
Precondiciones	Debe estar identificada la amenaza.
Flujo Normal de Eventos.	
Acciones del Actor	**Respuesta del Sistema**
2. a.1 El Responsable de Seguridad selecciona la opción “Ver Detalles de Amenaza.”.	2. Obtiene de la BD los activos relacionados con la amenaza seleccionada.
	3. Muestra listado de activos relacionados.
Flujo Alterno	
3. a Ver detalles de activo relacionado.	
El Responsable de Seguridad selecciona un activo para ver sus detalles.	**3. a.2** Muestra los detalles del activo seleccionado.

“Interfaz Ver Detalles de Amenaza”

Poscondiciones	

Tabla 2.11 Descripción del CUS “Ver Detalles de Amenaza”.

CU-10 Modificar Amenaza	
Actores	Responsable de Dominio (inicia).
Propósito	Modificar los valores previos de la amenaza.

Resumen: El caso de uso se inicia cuando el Responsable de Seguridad selecciona la opción Modificar valor de la amenaza. El sistema muestra un formulario con los valores asociados a la amenaza seleccioanda, usando el código de la misma. El Responsable de Seguridad modifica los datos.Se actualizan los datos en la BD.	
Referencias	RF23.
Precondiciones	Debe estar previamente identificada la amanaza.
Flujo Normal de Eventos.	
Acciones del Actor	**Respuesta del Sistema**
1. El Responsable de Seguridad selecciona la opción “Modificar amenaza”.	2. Obtiene de la BD las amenazas identificadas previamente, teniendo en cuenta el activo seleccionado.
	3. Muestra el formulario Modificar Valor de la Amenaza.
4. El Responsable de Seguridad modifica los campos deseados.	
5. El Responsable de Seguridad acciona el botón Aceptar.	6. Comprueba que se han introducido los datos obligatorios.
	7. Guarda la nueva información introducida en la BD.
Flujo Alterno	
6. a Campos Obligatorios.	
	6. a.1 Comprueba que se hayan introducido los campos obligatorios.
	6. a.2 Muestra un mensaje indicando los campos obligatorios.
“Interfaz Modificar Valor de la Amenaza”.	

Poscondiciones	

Tabla 2.12 Descripción del CUS "Modificar Amenaza".

CU-11 Identificar Salvaguardas	
Actores	Responsable de Dominio (inicia).
Propósito	Identificar las salvaguardas que se han previsto y desplegado hasta la fecha y determinar la eficacia de aquellas que se encuentran desplegadas.
Resumen: El caso de uso se inicia cuando el Responsable de Seguridad selecciona la opción "Identificar Salvaguardas". El sistema muestra un formulario que permite retirar del análisis aquellas salvaguardas que se considere que están fuera de lugar y por tanto no son aplicables, asi como introducir comentarios que justifican la retirada de la salvaguarda del sistema de información objeto de análisis.	
Referencias	RF24.
Precondiciones	Deben estar caracterizados los activos y las amenazas.

	Debe estar determinado el nivel de riesgo e impacto potencial.
Flujo Básico de Eventos.	
Acciones del Actor	**Respuesta del Sistema**
1. El Responsable de Seguridad selecciona la opción “Identificar salvaguardas”.	2. Obtiene de la BD el estado de cada una de las salvaguardas identificadas previamente para el activo seleccionado.
	3. Muestra el formulario identificar salvaguardas, que permite seleccionar o retirar la(s) salvaguarda(s) deseda(s). Indicando las salvaguardas desplegadas. Hablita el link Comentar Salvaguarda, usando el nombre de la misma.
4. El Responsable de Seguridad selecciona o deselecciona la(s) salvaguarda(s) deseada(s).	
5. El Responsable de Seguridad acciona el botón Guardar.	6. Registra la información en la BD.
Flujo Alterno	
5. a Comentar Salvaguarda.	
5. a.1 El Responsable de Seguridad acciona el link para introducir el comentario de la salvaguarda correspondiente.	5. a.2 Muestra la interfaz Comentar Salvaguarda.
5. a.3 El Responsable de Seguridad introduce comentario y acciona el botón Guardar comentario.	5. a.4 Almecena la información en la BD.
5. b “Cancelar la opción Guardar”.	
5. b.1 El Responsable de Dominio acciona el botón “Cancelar”.	5. b.2 Elimina los nuevos cambios introducidos.

Poscondiciones	No aplica.

Tabla 2.13 Descripción del CUS "Identificar Salvaguardas".

CU-12 Valorar Salvaguardas	
Actores	Responsable de Seguridad (inicia).
Propósito	Determinar la eficacia de las salvaguardas desplegadas.
Resumen: El caso de uso se inicia cuando el Responsable de Seguridad selecciona la opción "Valorar Salvaguardas". El sistema muestra un formulario que permite valorar aquellas salvaguardas identificadas previamente en un periodo de evaluación determinado, así como introducir comentarios que explican el por qué de la valoración de eficacia hecha sobre dicha salvaguarda.	
Referencias	RF25.
Precondiciones	Deben estar caracterizados los activos y las amenazas. Debe determinado el nivel de riesgo e impacto potencial. Debe estar creado el período de evaluación.
Flujo Básico de Eventos.	

Acciones del Actor	**Respuesta del Sistema**
1. El Responsable de Dominio selecciona la opción "Valorar salvaguardas".	2. Obtiene de la BD los datos de las salvaguardas previamente valoradas con su

	nivel de recomendación correspondiente a cada fase. Nota: El nivel de recomendación se define teniendo en cuenta la eficacia de la salvaguarda en cada fase. Los criterios definidos son: • Escasa información para hacer una recomendación. • La salvaguarda no se aplica. • La madurez es demasiado pobre: debe ser mejorada urgentemente. • La madurez es pobre: debe ser mejorada • La madurez es suficiente. • Si la eficacia de la salvaguarda está marcada como: ¿…? (necesidad de saber).
	3. Muestra el formulario valorar salvaguardas.
4. El Responsable de Seguridad introduce los datos a la salvaguarda selecciona: • Periodo de evaluación. • Valor de la eficacia. (0%, 10%, 50%, 90%, 95%, 100%, ¿…?). • Comentario.	
5. El Responsable de Seguridad acciona el botón Guardar.	6. Comprueba que se han introducido los datos obligatorios.
	7. Almacena los datos en la BD.

	8. Actualiza la lista de salvaguardas mostradas.
Flujo Alterno	
5. a "Cancelar".	
8. a.1 El Responsable de seguridad acciona el botón "Cancelar".	8. a.2 Limpia los campos seleccionados previamente.
9. a Campos Obligatorios.	
	9. a.1 Comprueba que se hayan introducido los campos obligatorios.
	9. a.2 Muestra un mensaje indicando los campos obligatorios.

Poscondiciones	No aplica

Tabla 2.14 Descripción del CUS "Valorar Salvaguardas".

CU-13 Calcular Impacto Potencial Acumulado	
Actores	Responsable de seguridad.
Propósito	Determinar el impacto potencial acumulado.
Resumen: El caso de uso se inicia cuando el Responsable de seguridad solicita al sistema	

<table>
<tr><td colspan="3">determinar el impacto potencial acumulado. El sistema muestra el valor de impacto acumulado calculado sobre los activos, por cada amenaza y en cada dimensión de seguridad.</td></tr>
<tr><td>Referencias</td><td colspan="2">RF 26</td></tr>
<tr><td>Precondiciones</td><td colspan="2">Deben estar valorados los activos y las amenazas.</td></tr>
<tr><td colspan="3">Flujo Normal de Eventos.</td></tr>
<tr><td>Acciones del Actor</td><td colspan="2">Respuesta del Sistema</td></tr>
<tr><td>1. El Responsable de Seguridad selecciona la opción “Impacto Acumulado”.</td><td colspan="2">2. Obtiene de la BD los valores que le permiten calcular el valor de impacto acumulado:
Para los activos, el valor acumulado.
Para las amenazas, la degradación causada por esta.</td></tr>
<tr><td></td><td colspan="2">3. Calcula el impacto acumulado.</td></tr>
<tr><td></td><td colspan="2">4. Muestra el resultado en una tabla teniendo en cuenta:
Activos.
Amenazas relacionadas.
Dimensiones (muestran el impacto calculado en cada una de las dimensiones).</td></tr>
<tr><td></td><td colspan="2">5. Habilita en botón “Guardar”.</td></tr>
<tr><td>6. El Responsable de Seguridad acciona el botón “Guardar”.</td><td colspan="2">7. Almacena los datos en la BD.</td></tr>
</table>

Poscondiciones	No aplica.

Tabla 2.15 Descripción del CUS "Determinar Impacto Potencial Acumulado".

CU-14 Calcular Impacto Potencial Repercutido	
Actores	Responsable de seguridad.
Propósito	Determinar el impacto potencial acumulado.
Resumen: El caso de uso se inicia cuando el Responsable de seguridad solicita al sistema determinar el impacto potencial repercutido. El sistema muestra el valor de impacto repercutido calculado para cada activo, por cada amenaza y en cada dimensión de seguridad.	
Referencias	RF27.
Precondiciones	Deben estar valorados los activos y las amenazas.
Flujo Normal de Eventos.	
Acciones del Actor	**Respuesta del Sistema**
1. El Responsable de Seguridad selecciona la opción "Impacto Repercutido".	2. Obtiene de la BD los valores que le permiten calcular el valor de impacto repercutido: • Para los activos, su valor propio. • Para las amenazas, que están expuestos los activos de los que depende
	3. Calcula el impacto repercutido.

	4. Muestra el resultado en una tabla teniendo en cuenta: • Activos. • Amenazas relacionadas. • Dimensiones (muestran el impacto calculado en cada una de las dimensiones).
	5. Habilita en botón “Guardar”.
6. El Responsable de Seguridad acciona el botón “Guardar”.	7. Almacena los datos en la BD.

Poscondiciones	No aplica.

Tabla 2.16 Descripción del CUS “Calcular impacto potencial repercutido”.

CU-15 Calcular Riesgo Potencial Acumulado	
Actores	Responsable de seguridad
Propósito	Determinar el riesgo potencial, al que está sometido el sistema de información.
Resumen: El caso de uso se inicia cuando el Responsable de seguridad selecciona la opción riesgo potencial acumulado. El sistema muestra el valor de riesgo acumulado, calculado para cada activo,	

por cada amenaza y en cada dimensión de seguridad.	
Referencias	RF28.
Precondiciones	Deben estar valorados los activos y las amenazas. Debe estar calculado el impacto acumulado.
Flujo Normal de Eventos.	
Acciones del Actor	**Respuesta del Sistema**
1. El Responsable de Seguridad selecciona la opción “Riesgo Acumulado”.	2. Activa la opción “Leyenda”.
	3. Obtiene de la BD los valores que le permiten calcular el valor de riesgo acumulado: • El impacto acumulado de cada uno de los activos. • La frecuencia de las amenazas correspondientes.
	4. Calcula el riesgo acumulado.
	5. Muestra el resultado en una tabla teniendo en cuenta: • Activos. • Amenazas relacionadas. • Dimensiones (muestran el impacto calculado en cada una de las dimensiones).
Flujo Alterno	
2. a Seleccionar opción “Leyenda”.	
	2. a.1 Muestra una tabla que muestra los criterios asociados a cada nivel de criticidad de los riesgos: Critico (5) Muy alto (4). Alto (3). Medio (2). Bajo (1). Muy Bajo (0).

Poscondiciones	No aplica.

Tabla 2.17 Descripción del CUS "Calcular riesgo potencial Acumulado".

CU-16 Calcular Riesgo Potencial Repercutido	
Actores	Responsable de seguridad (inicia).
Propósito	Determinar el riesgo potencial acumulado, al que está sometido el sistema de información.
Resumen: El caso de uso se inicia cuando el Responsable de seguridad selecciona la opción riesgo potencial acumulado. El sistema muestra el valor de riesgo repercutido, calculado para cada activo, por cada amenaza y en cada dimensión de seguridad.	
Referencias	RF29.
Precondiciones	• Debe estar calculado el valor de impacto repercutido. • Debe estar determinada la frecuencia de la amenazas.
Flujo Normal de Eventos.	
Acciones del Actor	**Respuesta del Sistema**
1. El Responsable de Seguridad selecciona la opción "Riesgo Repercutido".	2. Activa la opción "Leyenda".
	3. Obtiene de la BD los valores de impacto que le

	permiten calcular el valor de riesgo repercutido: • El impacto repercutido de cada uno de los activos. • La frecuencia de las amenazas correspondientes.
	4. Calcula el Riesgo Repercutido.
	5. Muestra el resultado en una tabla teniendo en cuenta: • Activos. • Amenazas relacionadas. • Dimensiones (muestran el impacto calculado en cada una de las dimensiones).
Flujo Alterno	
2. a Seleccionar opción "Leyenda".	
	2. a.1 Muestra una tabla que muestra los criterios asociados a cada nivel de criticidad de los riesgos: Critico (5) Muy alto (4). Alto (3). Medio (2). Bajo (1). Muy Bajo (0).

Poscondiciones	No aplica.

Tabla 2.18 Descripción del CUS "Calcular Riesgo Repercutido Potencial".

CU-17 Calcular Impacto Acumulado Residual	
Actores	Responsable de seguridad (inicia).
Propósito	Determinar el impacto acumulado residual al que está sometido el sistema de información.
Resumen: El caso de uso se inicia cuando el Responsable de seguridad selecciona la opción impacto residual acumulado, El sistema muestra los valores de impacto residual sobre los activos y las amenazas, en cada dimensión de seguridad.	
Referencias	RF30.
Precondiciones	• Debe estar identificada la amenaza. • Debe estar identificadas la salvaguarda. • Deben estar identificados el activo.
Flujo Normal de Eventos.	
Acciones del Actor	**Respuesta del Sistema**
1. El Responsable de Dominio selecciona la	2. Obtiene de la BD los valores de impacto que le

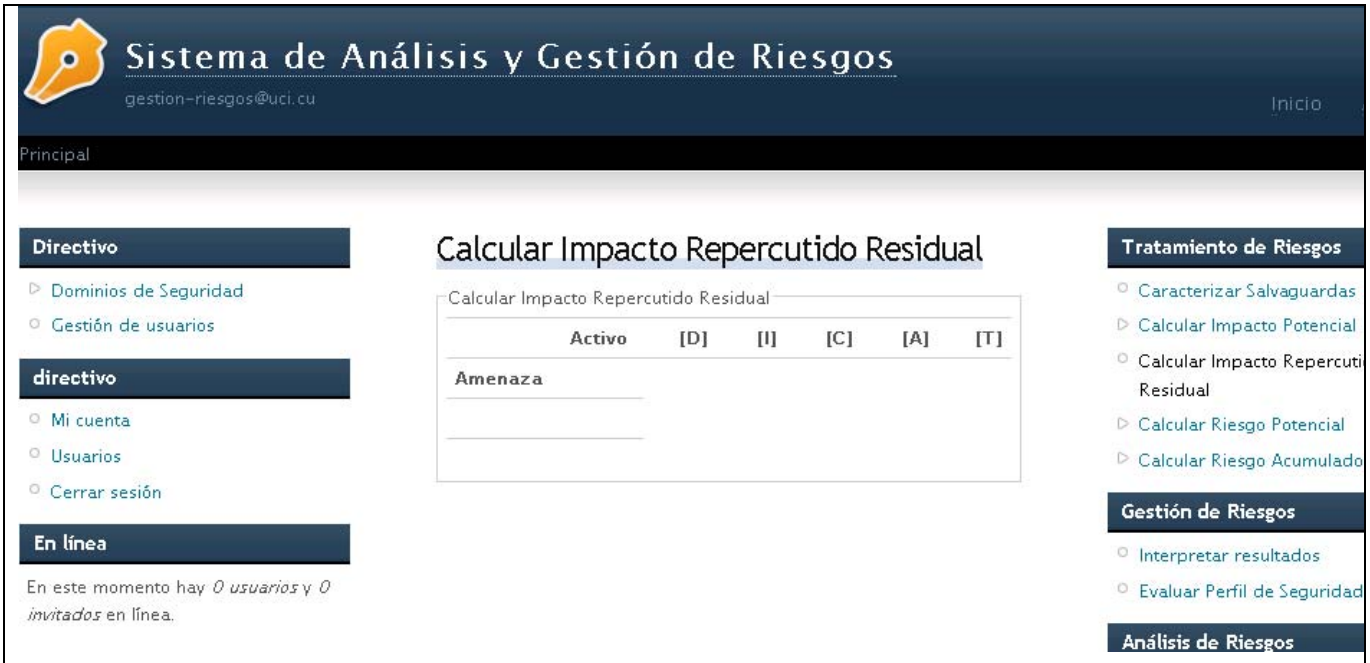
opción “Impacto Acumulado Residual”.	permiten calcular el valor de impacto repercutido: • Para los activos, el valor acumulado. • Para las amenazas, la degradación usada por esta. • Para las salvaguardas, el valor de la eficacia.
	3. Calcula la degradación residual.
	4. Calcula el impacto acumulado residual.
	5. Muestra el resultado en una tabla teniendo en cuenta: • Activos. • Amenazas relacionadas. • Dimensiones (muestran el impacto calculado en cada una de las dimensiones).
	6. Guarda los datos en la BD.

Poscondiciones	No aplica.

Tabla 2.19 Descripción del CUS “Calcular Impacto Acumulado Residual”.

CU-18 Calcular Impacto Repercutido Residual	
Actores	Responsable de seguridad (inicia).

Propósito	Determinar el impacto repercutido residual al que está sometido el sistema de información.
Resumen: El caso de uso se inicia cuando el Responsable de seguridad selecciona la opción Impacto Repercutido Residual, El sistema muestra los valores de impacto repercutido residual sobre los activos y las amenazas, en cada dimensión de seguridad.	
Referencias	RF31
Precondiciones	Debe estar calculado el impacto acumulado.
Flujo Normal de Eventos.	
Acciones del Actor	**Respuesta del Sistema**
1. El Responsable de Dominio selecciona la opción “Impacto Repercutido Residual”.	2. Obtiene de la BD los valores de impacto que le permiten calcular el valor de impacto repercutido residaul: • Para los activos, el valor propio. • Para las amenazas, la degradación de esta.
	3. Calcula la degradación residual.
	4. Calcula el impacto residual repercutido.
	5. Muestra el resultado en una tabla teniendo en cuenta: • Activos. • Amenazas relacionadas. • Dimensiones (muestran el impacto calculado en cada una de las dimensiones).
	6. Guarda los datos en la BD.



Poscondiciones	No aplica.

Tabla 2.20 Descripción del CUS "Calcular Impacto Repercutido Residual".

CU-19 Calcular Riesgo Acumulado Residual	
Actores	Responsable de seguridad (inicia).
Propósito	Determinar el riesgo acumulado residual al que está sometido el sistema.
Resumen: El caso de uso se inicia cuando el Responsable de seguridad solicita al sistema calcular del riesgo acumulado residual. El sistema, muestra los valores de riesgo acumulado residual sobre los activos y las amenazas, en cada dimensión de seguridad.	
Referencias	RF32.
Precondiciones	Deben estar valorados los activos, Deben estar valoradas las amenazas Deben estar valoradas las salvaguardas desplegadas.
Flujo Normal de Eventos.	
Acciones del Actor	**Respuesta del Sistema**
1. El Responsable de Dominio selecciona la opción "Calcular Riesgo Acumulado	2. Obtiene de la BD los valores de impacto que le permiten calcular el valor de riesgo acumulado:

Residual”.	• El impacto residual acumulado de cada uno de los activos. • La frecuencia de las amenazas correspondientes. • Eficacia de la salvaguarda.
	3. Calcula la frecuencia residual.
	4. Calcula el riesgo acumulado residual.
	5. Guarda los datos en la BD.
	6. Muestra el resultado en una tabla teniendo en cuenta: • Activos. • Amenazas relacionadas. • Dimensiones (muestran el impacto calculado en cada una de las dimensiones).
	7. Habilita la opción “Leyenda”.
Flujo Alterno.	
7. a Seleccionar opción “Leyenda”.	
7. a.1 El Responsable de Dominio selecciona la opción “Leyenda”.	7. a.2 Muestra una tabla que muestra los criterios asociados a cada nivel de criticidad de los riesgos: Critico (5) Muy alto (4). Alto (3). Medio (2). Bajo (1). Muy Bajo (0).

Poscondiciones	Impacto residual por activo.

Tabla 2.21 Descripción del CUS "Calcular Riesgo Acumulado Residual".

CU-20 Calcular Riesgo Repercutido Residual	
Actores	Responsable de seguridad (inicia).
Propósito	Determinar el riesgo repercutido residual al que está sometido el sistema.
Resumen: El caso de uso se inicia cuando el Responsable de seguridad solicita al sistema calcular el riesgo repercutido residual. El sistema, muestra los valores de riesgo repercutido residual sobre los activos y las amenazas, en cada dimensión de seguridad.	
Referencias	RF 33
Precondiciones	Deben estar valorados los activos, las amenazas y salvaguardas desplegadas.
Flujo Normal de Eventos.	
Acciones del Actor	**Respuesta del Sistema**
1. El Responsable de Seguridad selecciona la opción "Riesgo Repercutido Residual".	2. Obtiene de la BD los valores que le permiten calcular el valor de riesgo repercutido: • El impacto residual repercutido de cada uno

	de los activos. • La frecuencia la ocurrencia de la amenaza sobre el activo. • Eficacia de la salvaguarda mitigando la frecuencia de ocurrecia de la amenaza.
	3. Calcula la frecuencia residual.
	4. Calcula el riesgo repercutido residual.
	5. Guarda los datos en la BD.
	6. Muestra el resultado en una tabla teniendo en cuenta: • Activos. • Amenazas relacionadas. • Dimensiones (muestran el impacto calculado en cada una de las dimensiones).
	7. Habilita la opción “Leyenda”.
Flujo Alterno	
7. a Seleccionar opción “Leyenda”.	
	7. a.1 Muestra un menú con los criterios asociados a cada nivel de criticidad de los riesgos: Critico (5) Muy alto (4). Alto (3). Medio (2). Bajo (1). Muy Bajo (0).

Poscondiciones	

Tabla 2.22 Descripción del CUS "Calcular Riesgo Repercutido Residual."

2.5 Conclusiones parciales.

En este capítulo se han descrito las características principales del sistema. Con la realización del modelo de dominio se han definido los principales conceptos del entorno, así como las relaciones que se establecen entre ellos, lo que permitirá una mejor comprensión del sistema. Se determinaron los actores, se realizó el modelo de casos de uso del sistema y las descripciones textuales de dichos casos de uso, también se determinaron los principales requerimientos que el sistema deberá cumplir, obteniendo varios requisitos funcionales y algunos no funcionales clasificados en seguridad, soporte, hardware, usabilidad, entre otros.

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

3.1 Introducción

El capítulo presente, muestra en detalle, cómo está definida la arquitectura candidata del sistema. Permite representar una estructura global del mismo por medio de un Modelo de clases de Análisis, así como la modelación de los artefactos necesarios para su construcción. Se representan los componentes de la aplicación tratados como clases y representados a través de diagramas de clases con estereotipos Web. Se presenta el Modelo lógico de datos, a través del diagrama de clases persistentes y Modelo físico de datos, mediante el diagrama entidad relación, que constituyen la base principal para construir finalmente la base de datos que soportará todo el trabajo que debe realizar el sistema en general.

3.2 Descripción de la arquitectura utilizada.

Una arquitectura común de los sistemas de información que abarcan una interfaz para el usuario y el almacenamiento persistente de datos se conoce con el nombre de arquitectura en tres capas.
La descrición clásica de las capas se muestra a continuación:
1. Presentación: ventanas, reportes, etc.
2. Lógica de aplicaciones: tareas y reglas que rigen el proceso.
3. Almacenamiento: mecanisnmo de almacenamiento persistente.

La calidad tan especial de la arquitectura de tres capas consiste en aislar la lógica de la aplicación y en convertirla en una capa intermedia bien definida y lógica del software.
En la capa de presentación se realiza relativamente poco procesamineto de la aplicación; las ventanas envian a la capa intermedia peticiones de trabajo y este se comunica con la capa de almacenamiento del extremo posterior.

3.3 Patrón de arquitectura que se emplea.

El patrón utilizado para la realización de esta propuesta de software es: Modelo Vista Controlador (MVC) el cual separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos. Este patrón se ve frecuentemente en aplicaciones web, donde la vista es la página HTML y el código que provee de datos dinámicos a la página.

Son muchas las empresas que deciden pasar sus aplicaciones a la arquitectura modelo vista controlador para documentar más fácilmente el código, ahorrar espacio de tiempo y en caso de no

disponer de diseñadores web, poder contratar los servicios de un diseñador que no sepa mucho de programación que les haga las vistas.
El Modelo es todo acceso a datos, y las funciones que llevan lo que se llama "lógica de negocio", o sea datos y reglas de negocio. Lleva un registro de las vistas y controladores del sistema. Cada acceso a datos se pone en su función individual porque, de esta forma, si se cambia de gestor de bases de datos este cambio sólo afecta a estas funciones, no al resto de la aplicación.

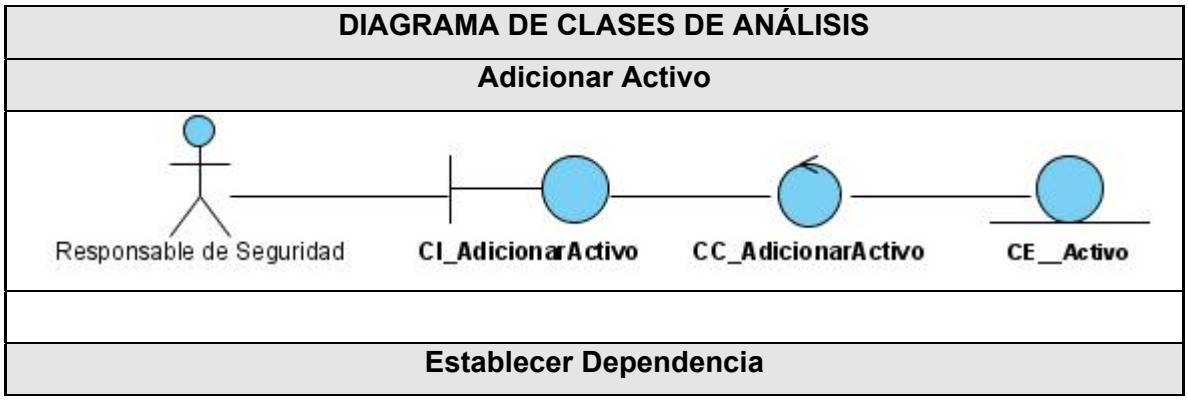
La Vista, en una aplicación web, es el HTML y lo necesario para convertir datos en HTML. O sea muestra la información del modelo al usuario. Tienen un registro de su controlador asociado (normalmente porque además lo instancia).

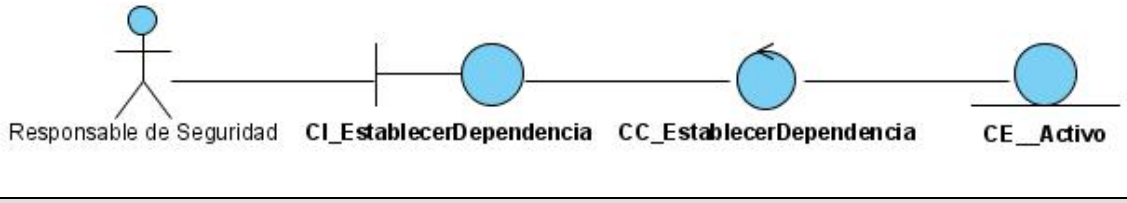
El Controlador es lo que une la vista y el modelo. Por ejemplo, son las funciones que toman los valores de un formulario, consultan la base de datos (a través del modelo) y producen valores, que la vista tomará y convertirá en HTML. En resumen, gestiona las entradas del usuario. Recibe los eventos de entrada (un clic, un cambio en un campo de texto, etc.). Contiene reglas de gestión de eventos, del tipo "SI Evento X, entonces Acción Y". Estas acciones pueden suponer peticiones al modelo o a las vistas. De este modo, el código que "hace algo" está perfectamente separado del código dedicado a crear HTML, lo que ayuda a evitar la unión de ambas partes.

3.4 Análisis.

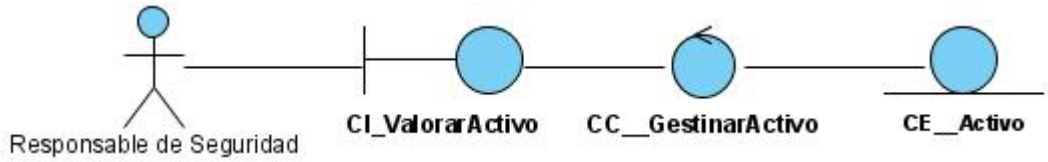
3.4.1 Definición del modelo de análisis. Modelo de clases de análisis.

A continuación se muestran los diagramas de clases de análisis y de diseño de los casos de uso críticos que conformarán funcionalidades del sistema propuesto, una vez que se desarrolle la fase de implementación.

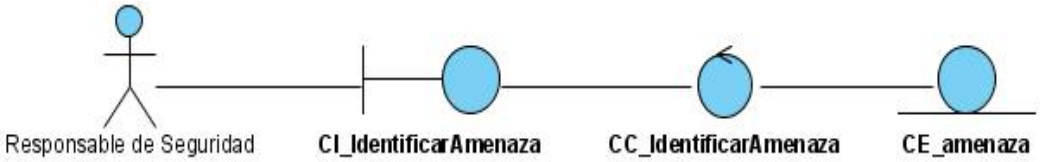




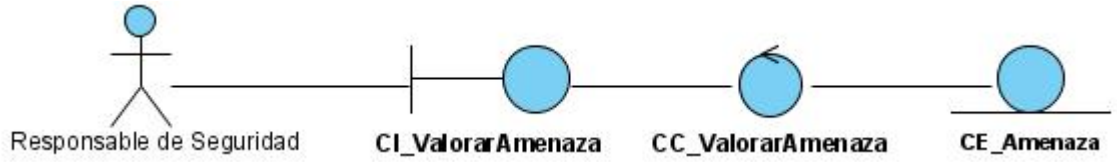
Valorar Activo



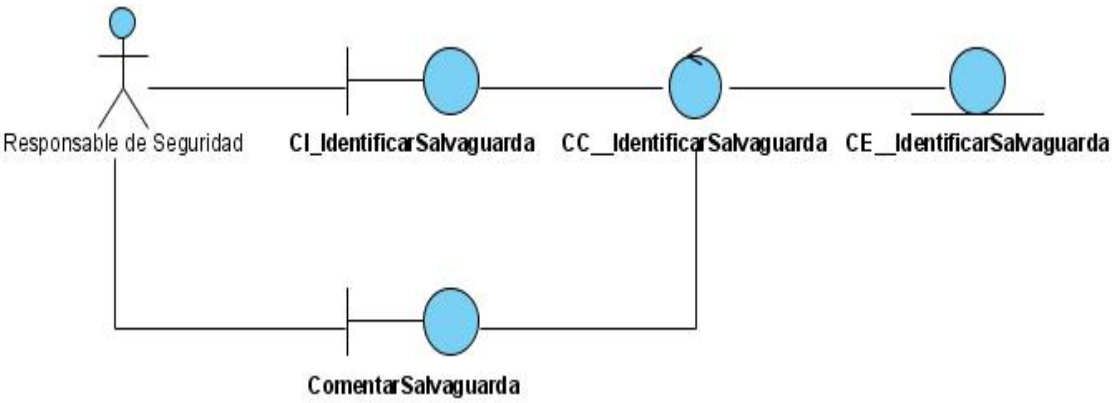
Identificar Amenaza



Valorar Amenaza



Identificar Salvaguarda



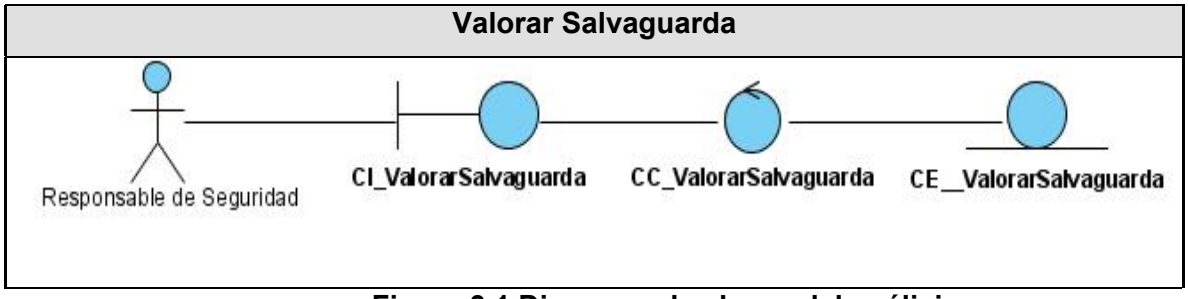


Figura 3.1 Diagrama de clases del análisis.

3.5 Diseño.

3.5.1 Diagramas de interacción.

Otro objetivo de esta etapa es la elaboración de los diagramas de interacción que muestran gráficamente como lo objetos se comunican entre sí. Esta interacción se puede expresar mediante diagramas de colaboración o diagramas de secuencia. Estos últimos son utilizados en el presente trabajo de tesis para detallar las secuencias de interacciones ordenadas en el tiempo de los objetos. Ver Anexo I

3.5.2 Diagrama de Clases Web.

El diagrama de clases para las Aplicaciones Web difiere un poco del resto de las aplicaciones que se acostumbra a construir puesto que en ellas son más importantes la modelación de la lógica y estado del negocio que los detalles de presentación. Para obtener un nivel correcto de abstracción y detalle que permita obtener un resultado final es mejor modelar los artefactos del sistema, es decir: modelar las páginas, los enlaces entre estas, todo el código que irá creando las páginas, así como el contenido dinámico de estas, una vez que estén en el navegador del cliente; estos son los artefactos que se necesitan modelar para que el desarrollador los implemente luego y obtener así el producto final. Se elaboró un diagrama de clases Web para cada caso de uso de forma tal que se facilite la comprensión de cómo se relacionan los distintos elementos en la realización de cada uno de ellos.

3.5.3 Diagrama de clases del diseño.

Diagrama de clase Adicionar Activo

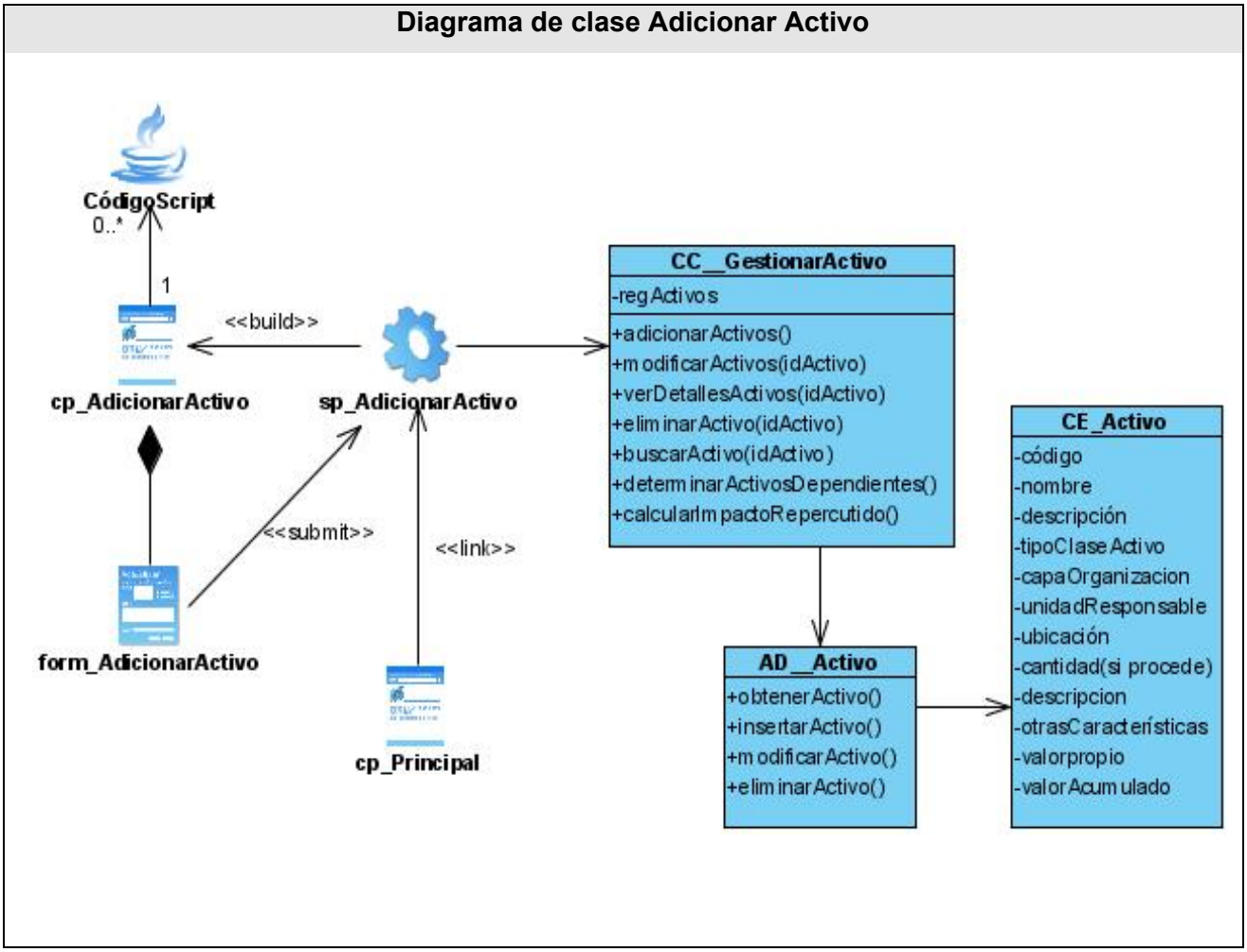


Figura 3.2 Diagrama de clase de diseño Adicionar Activo

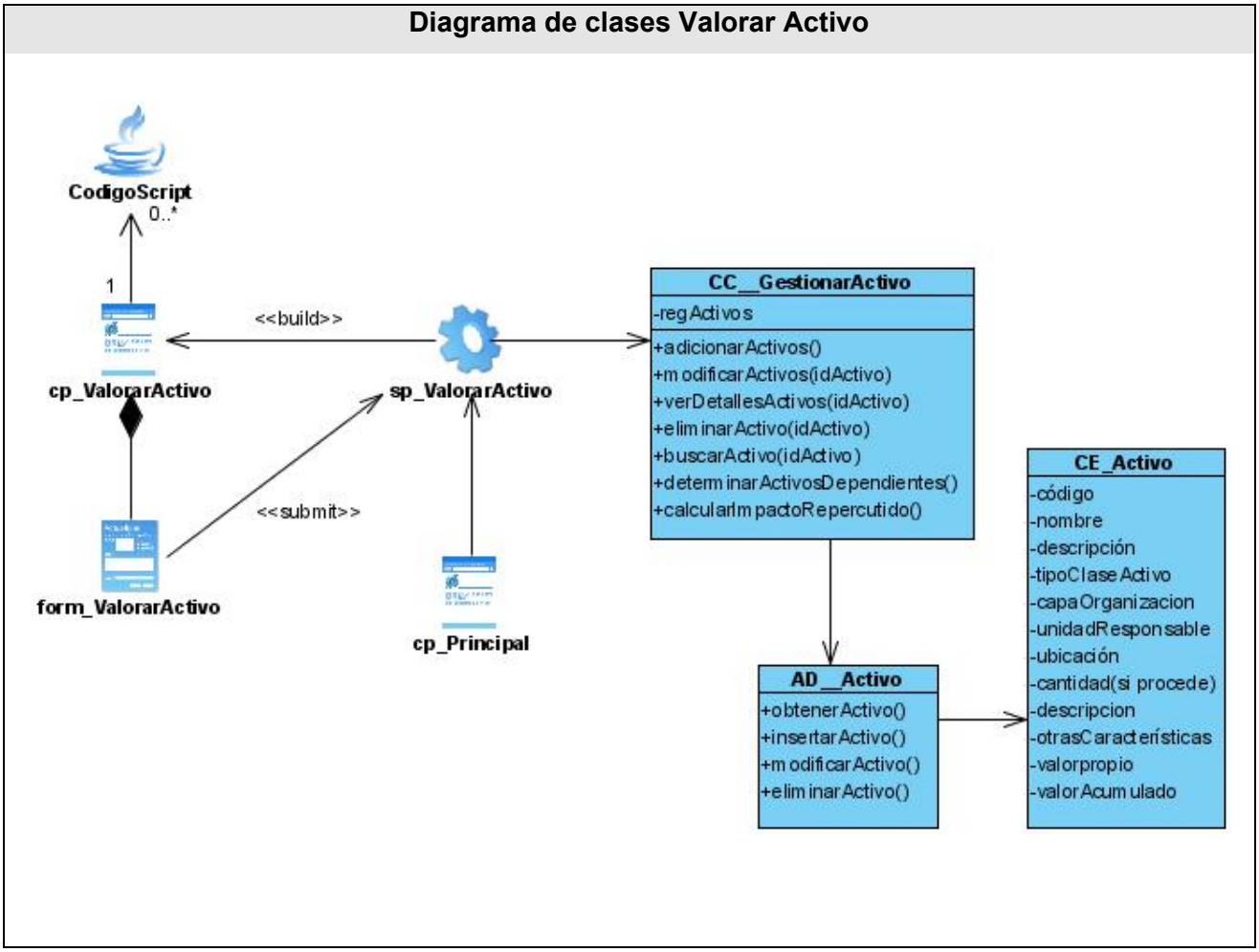


Figura 3.3 Diagrama de clase de diseño Valorar Activo.

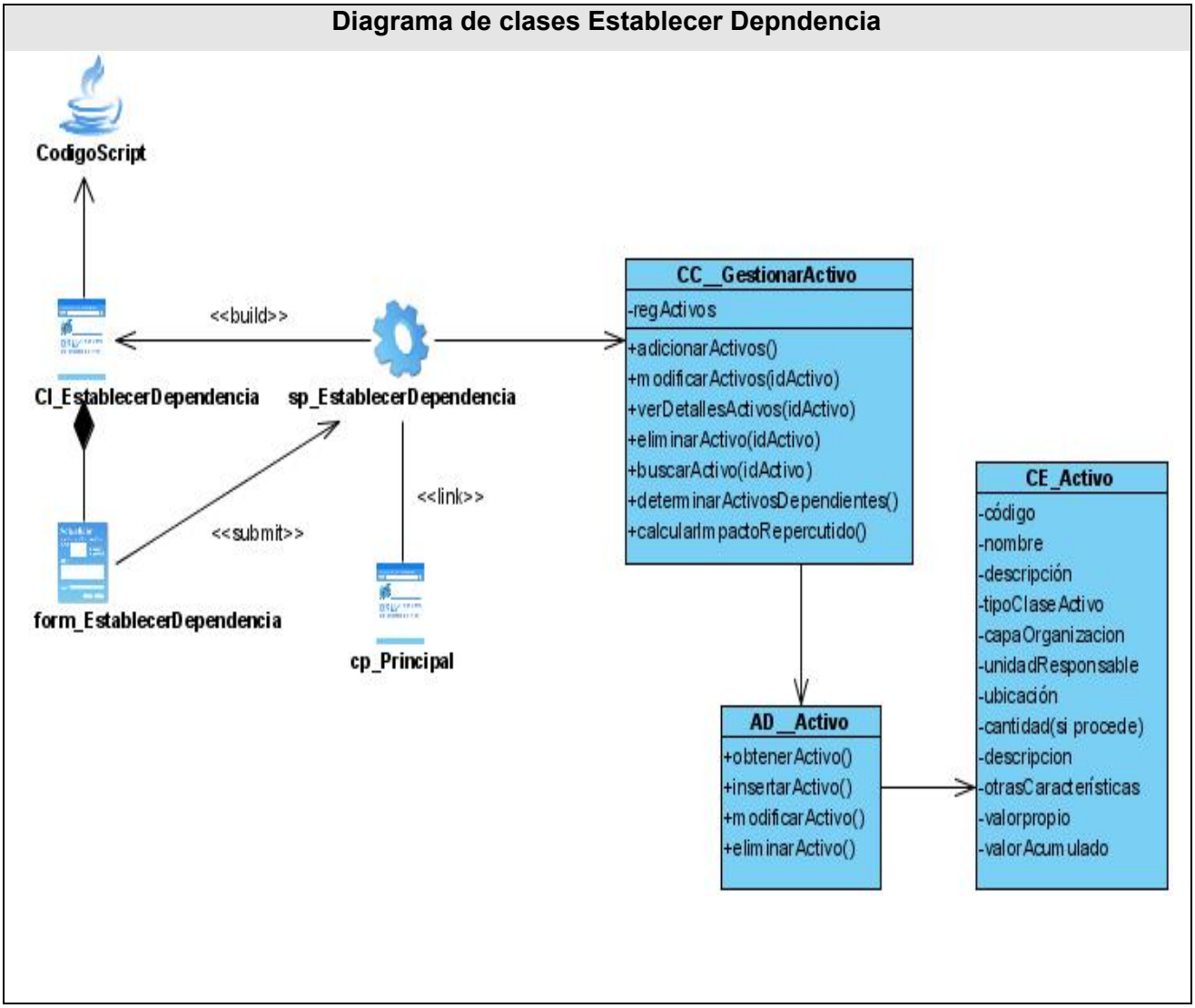


Figura 3.4 Diagrama de clase de diseño Establecer Depndencia.

Diagrama de clases Indentificar Amenaza

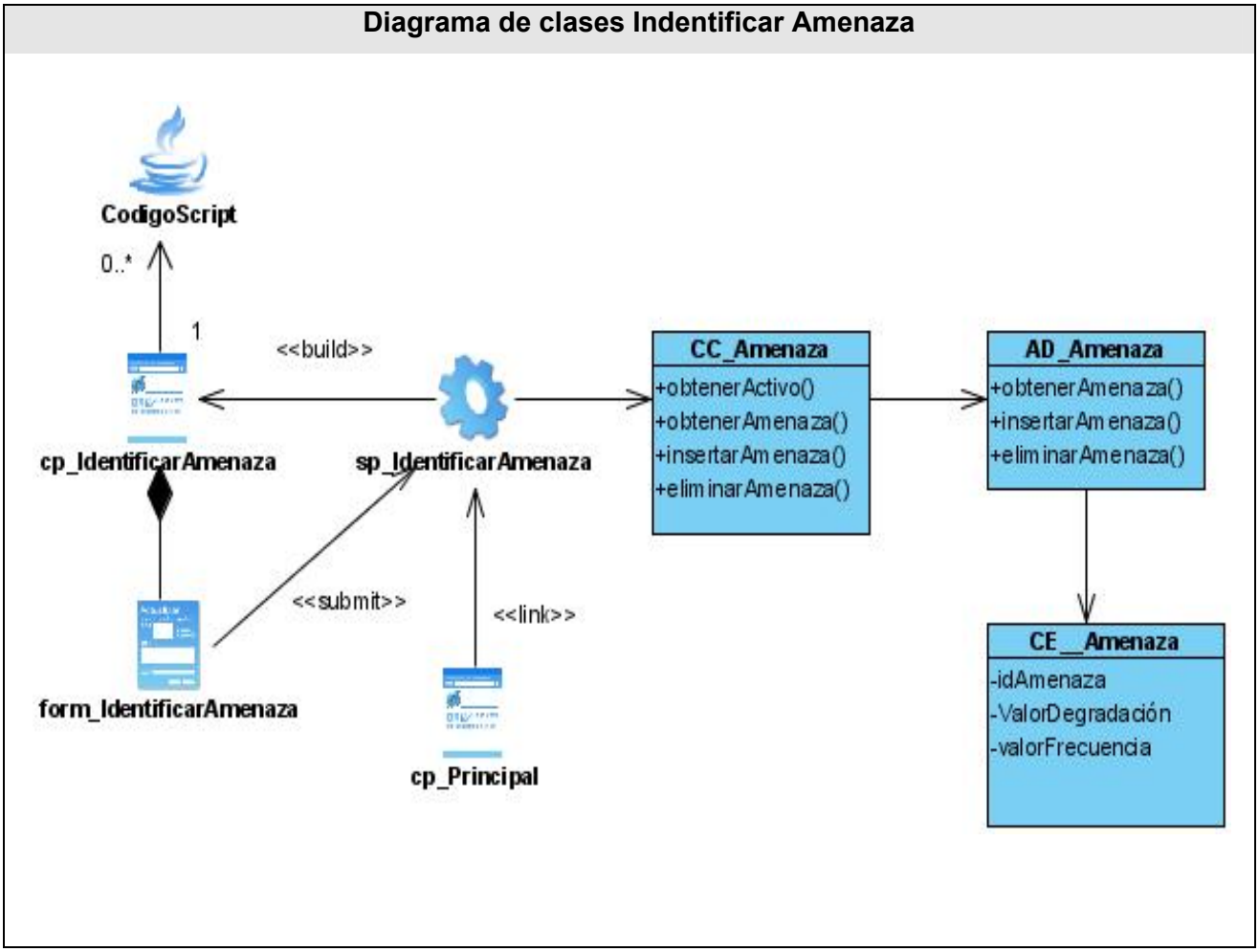


Figura 3.5 Diagrama de clase de diseño Indentificar Amenaza.

Diagrama de clases Valorar Amenaza

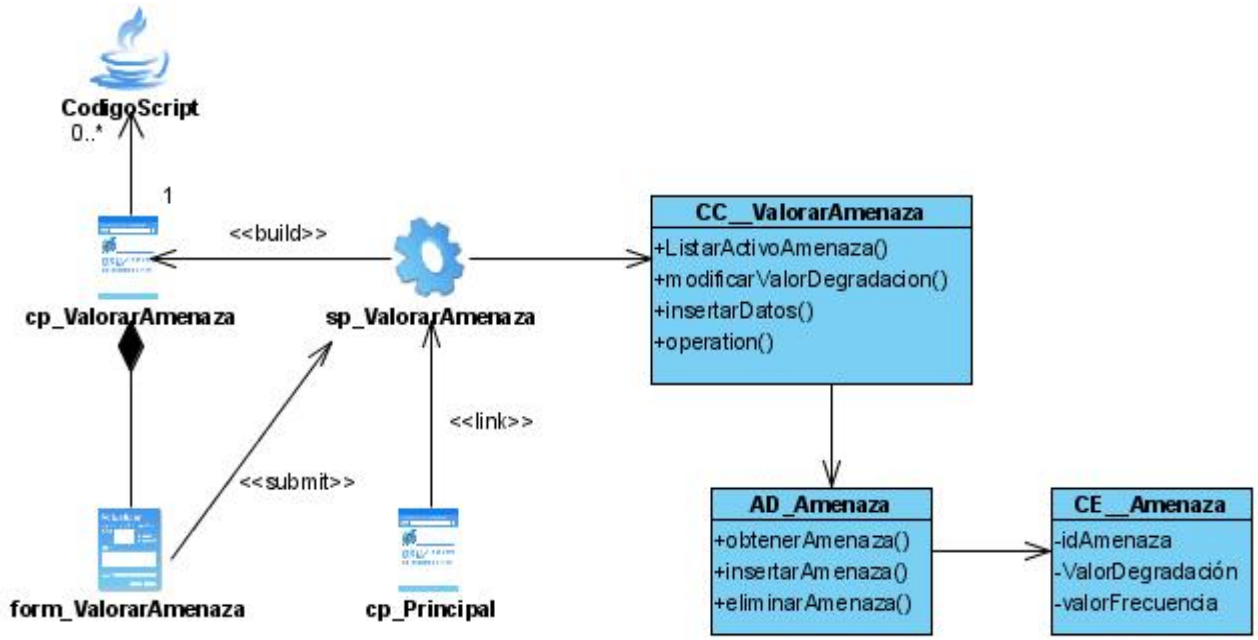


Figura 3.6 Diagrama de clase de diseño Valorar Amenaza.

Diagrama de clases Identificar Salvaguarda

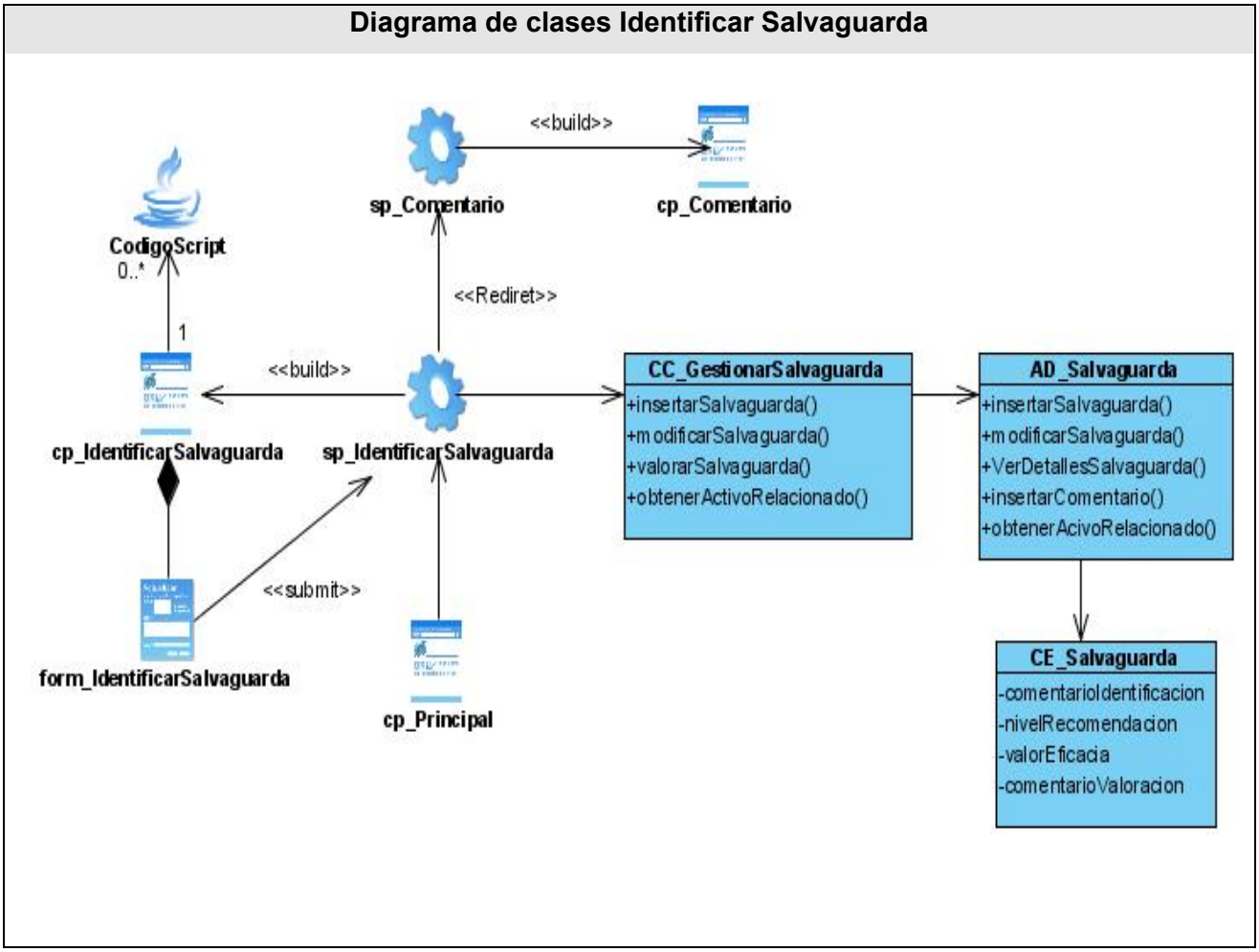


Figura 3.7 Diagrama de clase de diseño Identificar Salvaguarda.

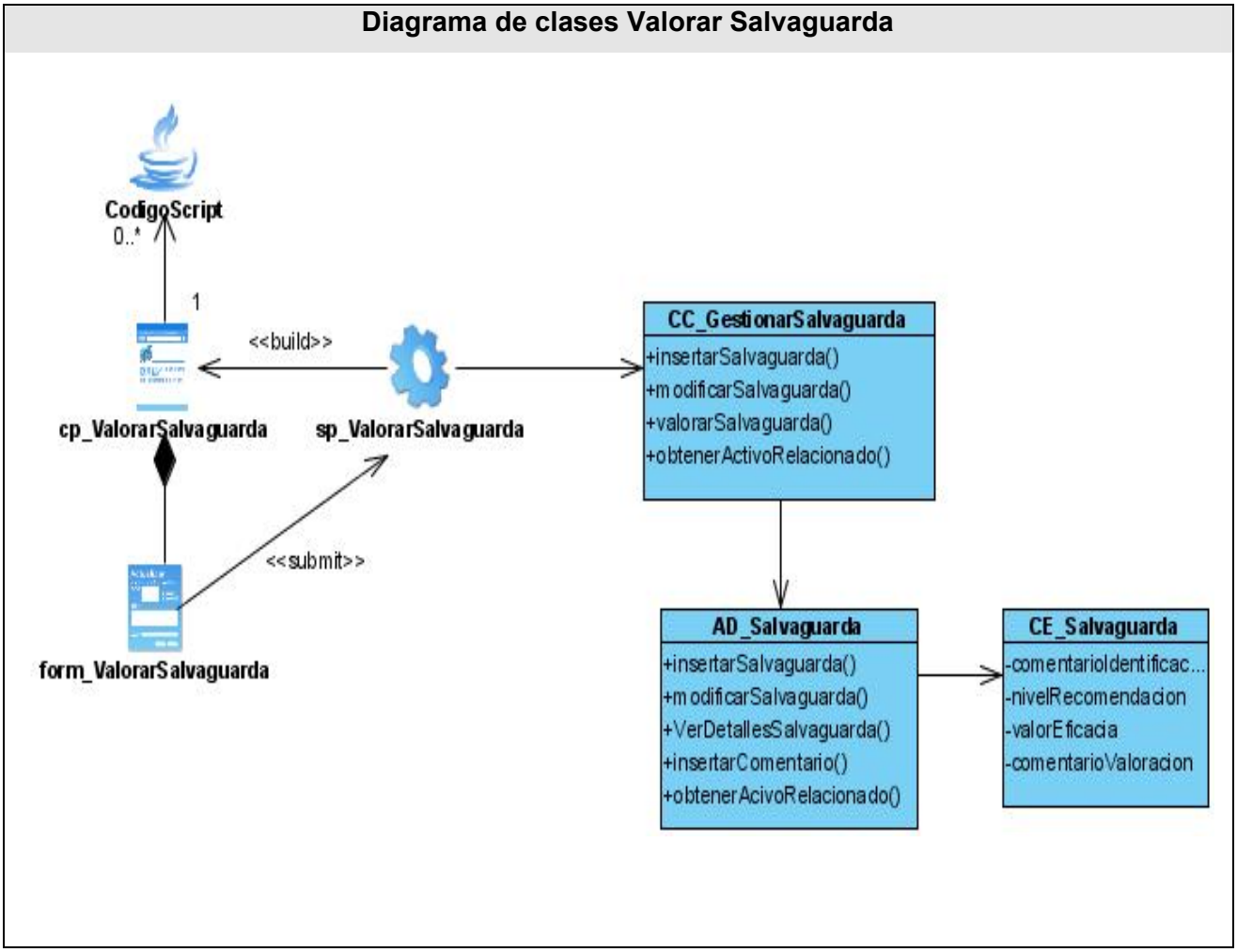


Figura 3.8 Diagrama de clase de diseño Valorar Salvaguarda.

3.6 Diseño de la base de datos.

3.6.1 Modelo lógico de datos.

Este modelo contiene el diagrama de clases persistentes que serán las tablas que compondrán la base de datos del sistema. A partir de este modelo se genera el modelo físico que es el que representa las relaciones de las tablas de la base de datos después de haber pasado por un proceso de mapeo donde las relaciones se transforman según su naturaleza y pueden o no incrementar el número de tablas de la base de datos.

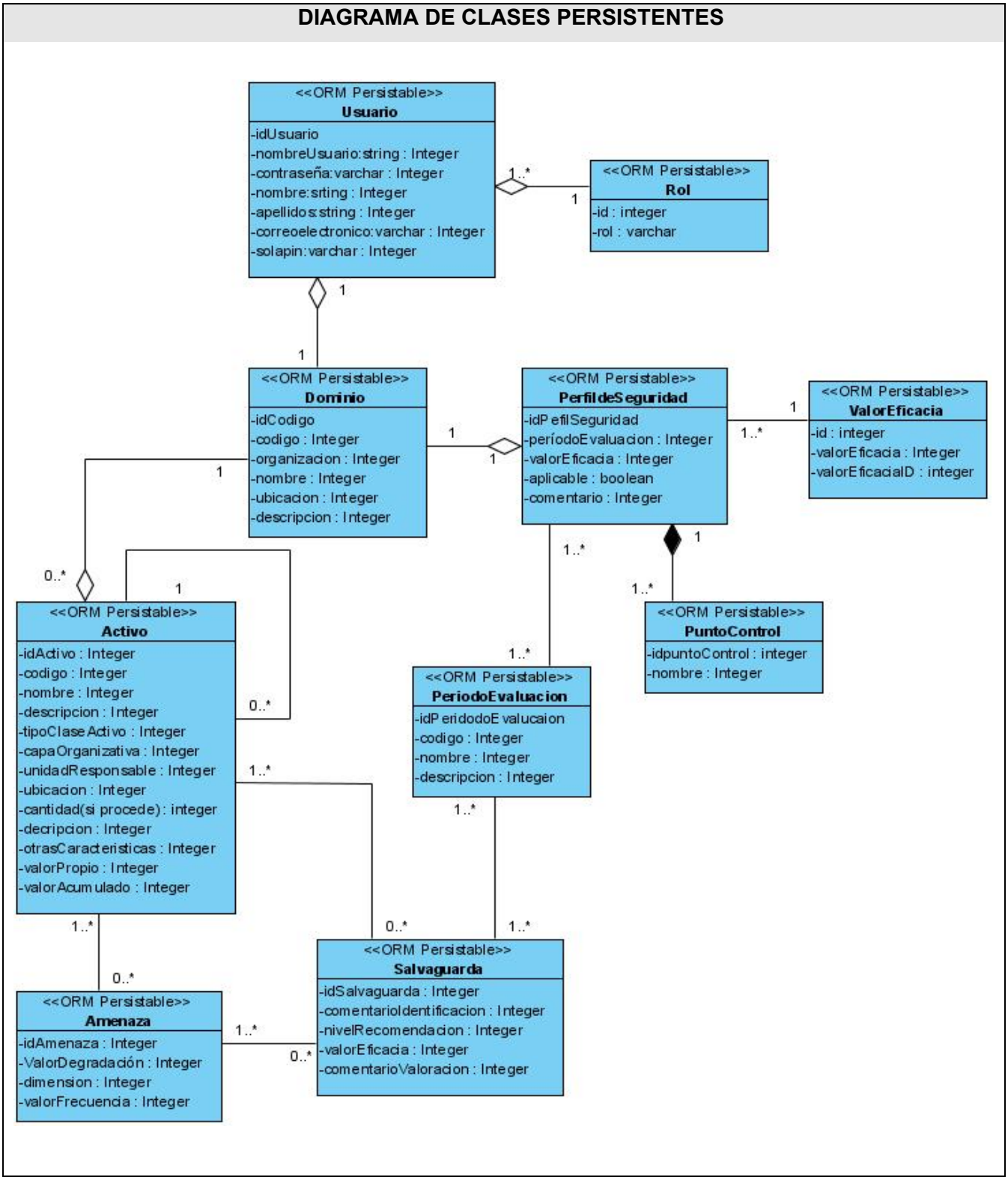


Figura 3.9 Diagrama de Clases Persistentes.

3.6.2 Modelo físico de datos.

Son las tablas que contiene la base de datos del sistema con sus atributos y relaciones entre las tablas y es donde se identifica el atributo identificador de la tabla. Este modelo de datos se realizó a partir del diagrama de clases persistentes, las cuales fueron identificadas en el diagrama de clases del diseño.

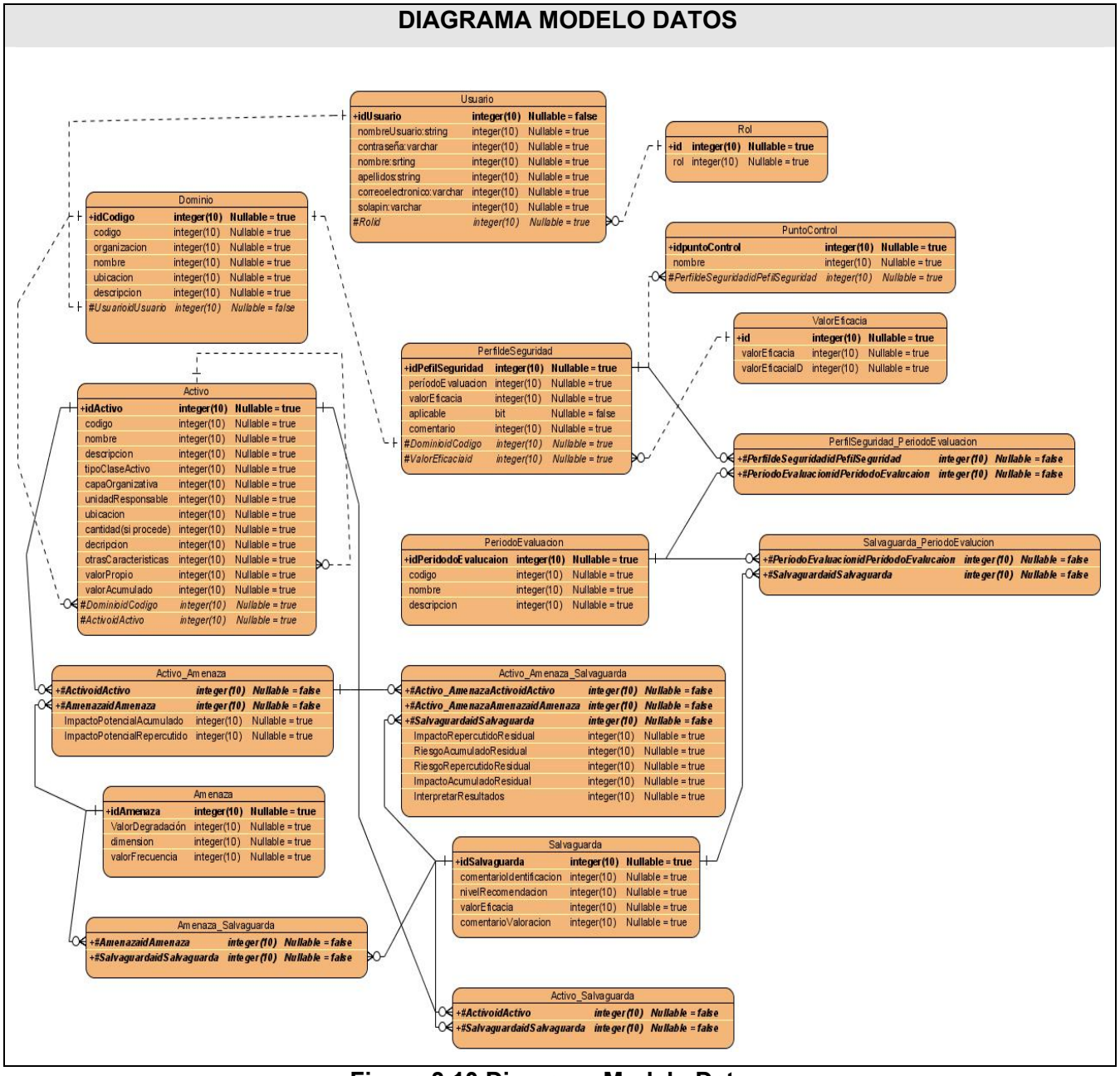


Figura 3.10 Diagrama Modelo Datos.

3.7 Modelo de despliegue.

El modelo de despliegue, es usado para modelar lo topología del hardware sobre el que se ejecuta el sistema y su distribución física, los distintos dispositivos y procesadores se representan a través de

nodos. En este caso el sistema debe disponer de computadoras clientes, un servidor para aplicaciones Web, un servidor para Base de Datos y la impresora.

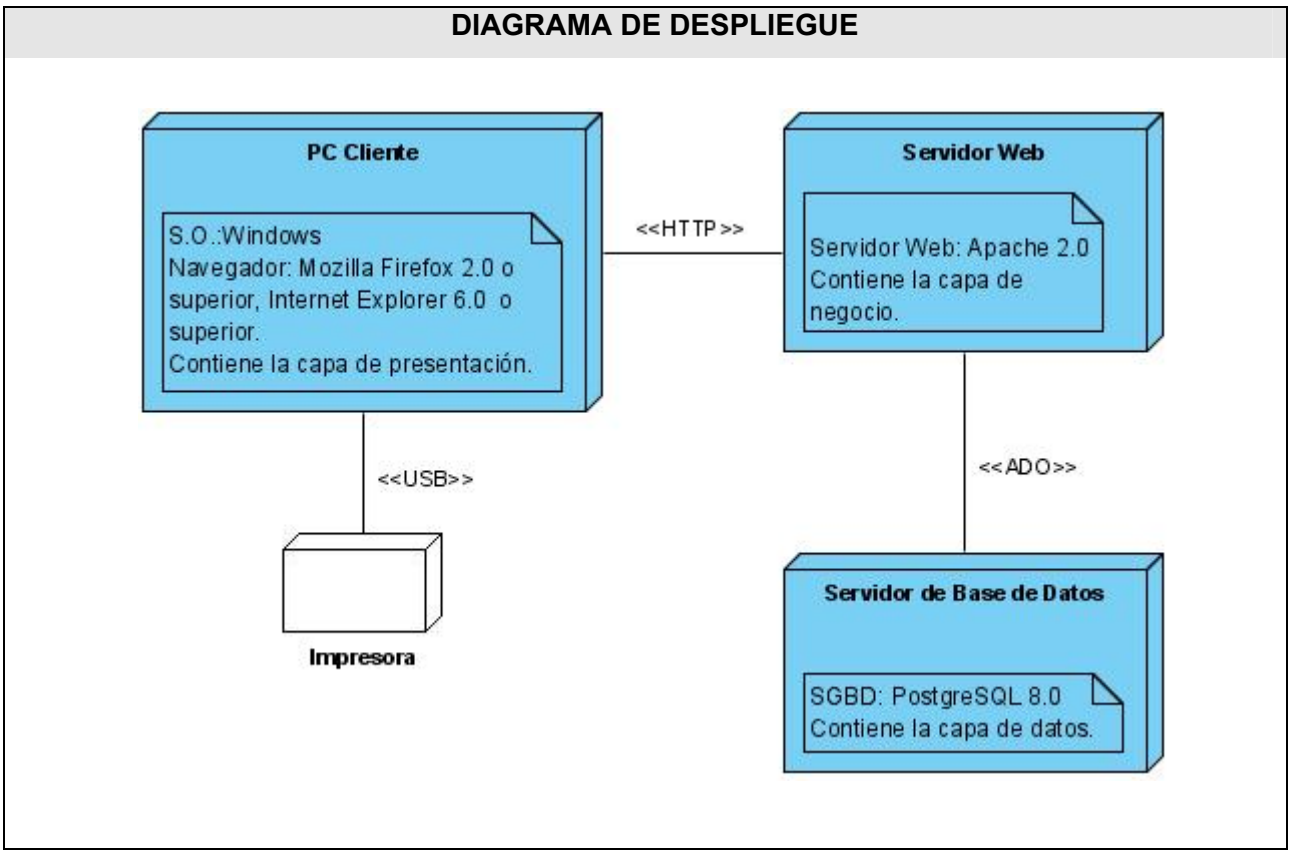


Figura 3.11 Diagrama de Desplieugue.

3.8 Conclusiones parciales.

Con el flujo de trabajo de análisis y diseño se ha logrado un acercamiento a la programación de la propuesta de solución, se modelaron los casos de usos con la utilización de la herramienta Visual Parading, y se presentaron a través de diagramas de clases. Se realizó además el diseño de la base de datos.

Todo el resultado obtenido constituirá una base para la adecuada implementación del sistema. Posteriormente se pasaría a construir el sistema, dándoles cumplimiento a todos los requisitos y las funcionalidades que son necesarias.

CAPÍTULO 4: ESTUDIO DE FACTIBILIDAD.

4.1 Introducción

Una de las principales tareas para asumir la realización de un proyecto es el análisis de factibilidad y viabilidad del proyecto, debido a la importancia de visualizar los beneficios que reportará. El estudio de la factibilidad de un proyecto es muy importante ya que evita la pérdida innecesaria de esfuerzo, tiempo y dinero, así como también permite establecer procedimientos razonables para poder desarrollar la Ingeniería de Software y operar todos los cambios producidos por los proyectos de Software. Para realizar una buena estimación se debe tener en cuenta la actividad de estimar los resultados del proyecto y los valores de costo, tiempo y recursos requeridos.

Este capítulo contiene el método de estimación por puntos de caso de uso, mediante el cual se obtiene el esfuerzo y costo del proyecto, el tiempo de desarrollo en meses costo y la cantidad de personas que se necesitan para su desarrollo. Comprende además el análisis del costo y el análisis del beneficio tangible e intangible.

4.2. Planificación basada en puntos de Casos de Uso.

La estimación mediante el análisis de Puntos de Casos de Uso se trata de un método de estimación del tiempo de desarrollo de un proyecto mediante la asignación de "pesos" a un cierto número de factores que lo afectan, para finalmente, contabilizar el tiempo total estimado para el proyecto a partir de esos factores.

A continuación, se detallan los pasos a seguir para la aplicación de éste método.

4.2.1. Calculo de puntos de Casos de Uso sin ajustar.

Consiste en el cálculo de los Puntos de Casos de Uso sin ajustar. Este valor, se calcula a partir de la siguiente ecuación:

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.

Calculando el factor de peso de los actores sin ajustar (UAW).

Este valor se calcula mediante un análisis de la cantidad de actores presentes en el sistema y la complejidad de cada uno de ellos.

Tipo de actor	Descripción	Factor de peso	Actor	Total
Simple	Otro sistema que interactúa con el sistema a desarrollar mediante una interfaz de programación	1	0	0
Medio	Otro sistema que interactúa con el sistema a desarrollar mediante un protocolo o una interfaz basada en texto	2	0	0
Complejo	Una persona que interactúa con el sistema mediante una interfaz gráfica	3	2	6

UAW = ∑ Factor de peso * Cantidad de actores

UAW =6

Calculando factor de Peso de los Casos de Uso sin ajustar (UUCW).

Este valor se calcula mediante un análisis de la cantidad de Casos de Uso presentes en el sistema y la complejidad de cada uno de ellos. La complejidad de los Casos de Uso se establece teniendo en cuenta la cantidad de transacciones efectuadas en el mismo, donde una transacción se entiende como una secuencia de actividades atómica, es decir, se efectúa la secuencia de actividades completa, o no se efectúa ninguna de las actividades de la secuencia. Los criterios se muestran en la siguiente tabla:

Tipo de CU	Descripción	Factor de peso	Cant de CU	Total
Simple	El caso de uso tiene de 1 a 3 transacciones.	5	31	155
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	0	0

UUCW=∑ Factor de peso*Cant de CU

UUCW= 195

UUCP = UAW + UUCW

UUCP= 6+195

UUCP= 201

4.2.2. Cálculo de Puntos de Casos de Uso ajustados.

Una vez que se tienen los Puntos de Casos de Uso sin ajustar, se debe ajustar éste valor mediante la siguiente ecuación:

UCP = UUCP x TCF x EF

donde,

UCP: Puntos de Casos de Uso ajustados

UUCP: Puntos de Casos de Uso sin ajustar (111)

TCF: Factor de complejidad técnica

EF: Factor de ambiente

Calculando Factor de complejidad técnica (TCF).

Este coeficiente se calcula mediante la cuantificación de un conjunto de factores que determinan la complejidad técnica del sistema. Cada uno de los factores se cuantifica con un valor de 0 a 5, donde 0 significa un aporte irrelevante y 5 un aporte muy importante. En la siguiente tabla se muestra el significado y el peso de cada uno de éstos factores:

Factor	Descripción	Peso	Valor asignado	Total
T1	Sistema distribuido	2	0	0
T2	Tiempo de respuesta	1	3	3
T3	Eficiencia del usuario final	1	4	4
T4	Funcionamiento Interno complejo	1	5	5
T5	El código debe ser reutilizable	1	4	4
T6	Facilidad de instalación	0.5	3	1.5
T7	Facilidad de uso	0.5	3	1.5
T8	Portabilidad	2	4	8
T9	Facilidad de cambio	1	3	3

T10	Concurrencia	1	5	5
T11	Incluye objetivos especiales de seguridad	1	3	3
T12	Provee acceso directo a terceras partes	1	0	0
T13	Se requieren facilidades especiales de entrenamiento de usuarios	1	4	4

∑= 42

El Factor de complejidad técnica se calcula mediante la siguiente ecuación:

$TCF = 0.6 + 0.01 \times \Sigma (Peso_i * Valor\ asignado_i)$

TCF= **0.6 + 0.01 x 42**

TCF= 1.02

Calculando el Factor Ambiente (EF).

Las habilidades y el entrenamiento del grupo involucrado en el desarrollo tienen un gran impacto en las estimaciones de tiempo. Estos factores son los que se contemplan en el cálculo del Factor de ambiente.

- Para los factores E1 al E4, un valor asignado de 0 significa sin experiencia, 3 experiencia media y 5 amplia experiencia (experto).
- Para el factor E5, 0 significa sin motivación para el proyecto, 3 motivación media y 5 alta motivación.
- Para el factor E6, 0 significa requerimientos extremadamente inestables, 3 estabilidad media y 5 requerimientos estables sin posibilidad de cambios.
- Para el factor E7, 0 significa que no hay personal part-time (es decir todos son full-time), 3 significa mitad y mitad, y 5 significa que todo el personal es part-time (nadie es full-time).
- Para el factor E8, 0 significa que el lenguaje de programación es fácil de usar, 3 medio y 5 que el lenguaje es extremadamente difícil.

Factor	**Descripción**	**Peso**	**Valor asignado**	**Total**
E1	Familiaridad con el modelo de proyecto	1.5	5	7.5

	utilizado			
E2	Experiencia en la aplicación	0.5	0	0
E3	Experiencia en la orientación a objetos	1	4	4
E4	Capacidad del analista líder	0.5	5	2.5
E5	Motivación	1	5	5
E6	Estabilidad de requerimientos	2	3	6
E7	Personal Part–Time	-1	3	-3
E8	Dificultad del lenguaje de programación	-1	3	-3

$EF = 1.4 - 0.03 * \sum (Peso_i * Valor\ asignado_i)$

EF= 1.4 – 0.03 * 19

EF= 0.83

Finalmente, los Puntos de Casos de Uso ajustados resultan:

UCP = 201 * 1.02 * 0.83

UCP = 170.16

4.2.3. Estimación del esfuerzo.

El esfuerzo en horas-hombre viene dado por:

E = UCP * CF

donde,

E: esfuerzo estimado en horas-hombre

UCP: Puntos de Casos de Uso ajustados (170.16)

CF: factor de conversión (20)

E= 170.16 * 20

E= 3403.33 Horas-Hombre.

Actividad	**Porcentaje %**	**Horas-Hombres**
Análisis	10	850.83
Diseño	20	1701.67
Implementación	40	3403.33
Pruebas	15	1276.25
Sobrecarga (otras actividades)	15	1276.25
Total	100	8508.32

Análisis de costo.

Esfuerzo Total horas-hombre (E_{THH}) →8508.32

Esfuerzo Total mes-hombre (E_{TMH}) → Este valor se obtiene de la división del E_{THH} entre el total de horas que se trabajan al mes. Como se trabajan un total de 8 horas diarias al mes esto representa un valor de 240 horas, entonces:

$E_{TMH} = E_{THH} / 240$

=8508.32 / 240

= 35.45

Salario Promedio (S_M)→ 100

Cantidad de Hombres (C)→ 3

Costo Hombre-Mes (C_{HM}) → Este valor es el resultado de la multiplicación del S_M por C:

$C_{HM} = 300$

Costo Total (T) → El costo total se obtiene multiplicando E_{TMH} por el C_{HM}, entonces:

$T = E_{TMH} * C_{HM}$

= 35.45 * 300

= 10 635

4.3. Beneficios tangibles e intangibles.

El sistema de análisis y gestión de riesgos, no se ha realizado con fines comerciales puesto que su objetivo fundamental es: Analizar y gestionar los riesgos que puedan incidir sobre los activos y bienes informáticos de la UCI en cada una de sus áreas de desarrollo o dominios, inicialmente.
Por lo que los beneficios intangibles son:

- Disminución del nivel de impacto y riesgo en el sistema de información.
- Facilitar la evaluación de riesgos.
- Analizar y visualizar en línea los riesgos de TI en la UCI.
- Registrar ordenadamente los activos del dominio.

4.5 Conclusiones parciales.

Finalizado el estudio de la factibilidad del sistema que se ha propuesto, teniendo en cuenta el costo estimado y los beneficios que aportará su desarrollo, se puede concluir, que es factible la realización de dicho sistema, lo cual se puede apreciar en los valores satisfactorios obtenidos que especifican la duración de desarrollo del proyecto y la cantidad de personas necesarias para su realización.

CONCLUSIONES

Se espera que el documento haya servido para la comprensión teórica de la situación problémica existente y la solución propuesta, así como el desarrollo de las diferentes etapas de la misma hasta fase de análisis y diseño usando la metodología RUP.

Se alcanzó, satisfactoriamente, el objetivo propuesto: realizar el análisis y diseño de una herramienta que permita realizar un correcto análisis y gestión de riesgos sobre los activos y/o sistemas informáticos de la UCI.

Se obtuvieron además los siguientes resultados:

1. Se ha demostrado la eficacia y necesidad de las metodologías, lenguajes y tecnologías utilizadas para el desarrollo del sistema.
2. Se realizó el análisis y diseño del sistema.
3. La solución propuesta ha sido acertada, los requerimientos planteados soportan al sistema y los casos de uso satisfacen las necesidades funcionales del mismo.
4. Se han seguido los principios básicos de diseño descritos para el desarrollo del sistema.
5. Se logra una seguridad y protección de los datos consecuente con el nivel de seguridad requerido.
6. Se ha elaborado la propuesta basada en la Metodología MAGERIT versión 2.0, que permitirá realizar el análisis de los riesgos del sistema de información de la UCI.

RECOMENDACIONES

Se recomienda:

1. Que se lleve a cabo la implementación y aplicación del sistema que se propone en la UCI.

2. Elaborar un procedimiento o sintaxis que permita publicar periódicamente las actualizaciones de las amenazas que conforman el sistema, dado que las amenazas pueden evolucionar en el tiempo para adaptarse a la evolución tecnológica.

3. Que en cada formulario se muestre una ayuda de cada elemento. Por ejemplo, una URL que vaya a la descripción del elemento.

4. Que se definan las dominios por grupos de activos de acuerdo a su importancia para la Organización, lo cual garantice una definición más exacta de los activos más valiosos y con ello lograr mayor eficiencia en los servicios prestados por la entidad ya sean internos o a terceros.

5. Que los riesgos que se puedan identificar se realicen a partir de roles para garantizar un análisis más detallado y profundo de los mismos.

6. Que se desarrolle el módulo de análisis cuantitativo, de manera que se integre al proyecto y permita hacer valoraciones simultáneas entre el modelo cuantitativo y el cualitativo, lo cual facilitaría la toma de desiciones por parte de la Organización.

REFERENCIA BIBLIOGRAFICA

[1] Conferencias de Seguridad Informática, curso 07-08, UCI.

[2] La gestión del riesgo. Cosultado en marzo 2008, disponible en: http://209.85.215.104/search?q=cache:liYVU3QNmdwJ:www.hispasec.com/unaaldia/2417+qu%C3%A9+es+gestion+de+riesgos+%2B+seguridad+informatica&hl=es&ct=clnk&cd=5&gl=cu&client=firefox-a [3] MAGERIT 2.0, Metodologia

[4] ISO/IEC 27002:2005, Cosultado en marzo, 2008http://www.iso.org/iso/iso_catalogue/catalogue_tc/ catalogue_detail.htm?csnumber=50297

[5] METODOLOGÍAS DE DESARROLLO DE SOFTWARE, 2005 / 2006.

[6] RUP: Rational Unified Process (Proceso Unificado de desarrollo).

[7] Softar Versión 2.3,autores: Ing. Gloria Naola Oduardo González, Ing. Ragnar Bermudez La O, Oficina de Seguridad para las Redes Informáticas.

[8] Herramientas CASE, Cosultado en marzo 2008, http://www.csi.map.es/csi/silice/Sgcase5.html

[9] Visual Paradigm en la Wiki de Producción. Publicado 28 marzo 2007. Disponible en http://wiki.prod.uci.cu/index.php/Visual_Paradigm.

[10] Lenguajes de Programación para la Web. Cosultado en marzo 2008. Disponible en: http://www.tejedoresdelweb.com/307/article-1883.html

[11] ¿Es PHP un buen primer lenguaje de programación? Consultado 17 mayo 2008. Disponible en http://php.uci.cu/?q=node/286

BIBLIOGRAFÍA

1. UML y Patrones, Introducción al analisis y diseño orientado a objetos, volumen 1, autor Craig Larman. Consultado abril 2008.
2. J. C. P. y. A. Cid, Herramienta de autor para la creación y gestión de objetos de aprendizaje, 2006.
3. L. Welicki, Patrones y Antipatrones: una Introducción –Parte II.
4. En este apartado se resumen las distintas normas que componen la serie ISO 27000 y se indica cómo puede una organización implantar un sistema de gestión de seguridad de la información (SGSI) basado en ISO 27001. Disponible en: <http://www.iso27000.es/iso27000.html> [Fecha de consulta 28 marzo 2008].
5. Utilizar UML ayuda a los equipos de proyecto a comunicar, explorar diseños potenciales y validar el diseño de arquitectura del software. Disponible en: <http://www.taringa.net/posts/downloads/1025883/Libros-y-Cursos-de-UML.html> [Fecha de consulta 04 marzo 2008].
6. Análisis y gestión de riesgos de la seguridad de los sistemas de la información. Disponible en: <http://www.cii-murcia.es/informas/abr05/articulos/Analisis_gestion_riesgos_seguridad_sistemas_informacion.php> [Fecha de consulta 29 marzo 2008].
7. Gestión de Riesgos de los Sistemas de Información. Disponible en: <http://www.web.eiconet.es/content/view/49/9/> [Fecha de consulta 29 marzo 2008].
8. Lenguajes de Programación para la Web. Disponible en: <http://www.tejedoresdelweb.com/307/article-1883.html > Publicado el: febrero 2007 de última actualización: marzo 2008.
9. ¿Es PHP un buen primer lenguaje de programación? Disponible en> <http://php.uci.cu/?q=node/286> Publicado17 mayo 2008.
10. Peralta, Mario. Estimación del esfuerzo basada en casos de uso. Centro de Ingienería del Software e Ingienería del Conocimiento, Buenos Aires, Argentina.
11. Larman, Craig. UML y Patrones. Introducción al análisis y diseño orientado a objetos y al proceso unificado. Segunda Edición por Prentice Hall.
12. MAGERIT-versión 2. Metodología de Análisis y Gestión de Reisgos de los Sistemas de Información. I Método. II Catálogo de Elementos. III Guía de Técnicas.

ANEXOS

Anexo I Diagramas de Interacción (Diagramas de secuencia del diseño).

Diagrama de de secuencia CU Adicionar Activo

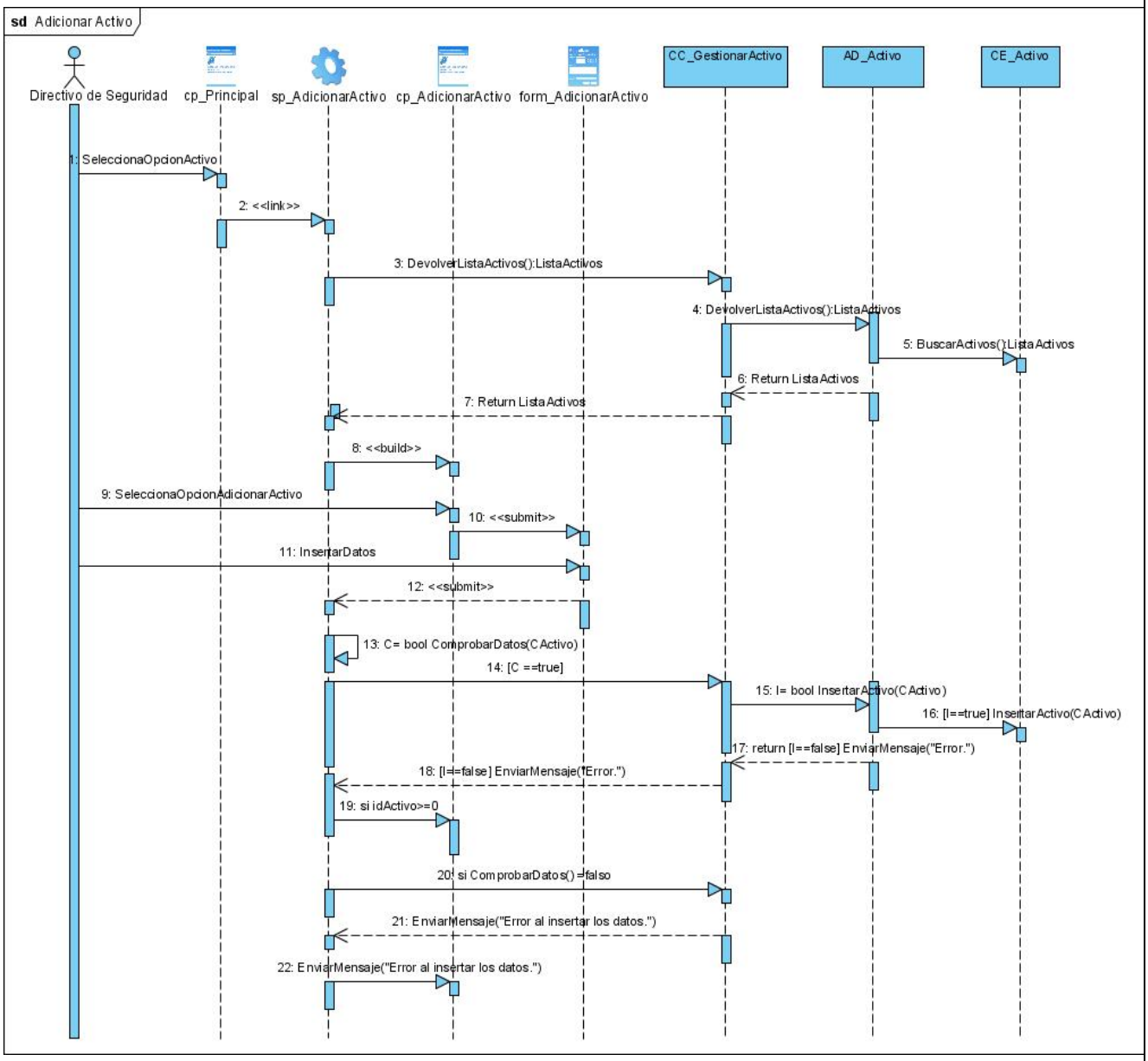


Diagrama de secuencia CU Valorar Activo

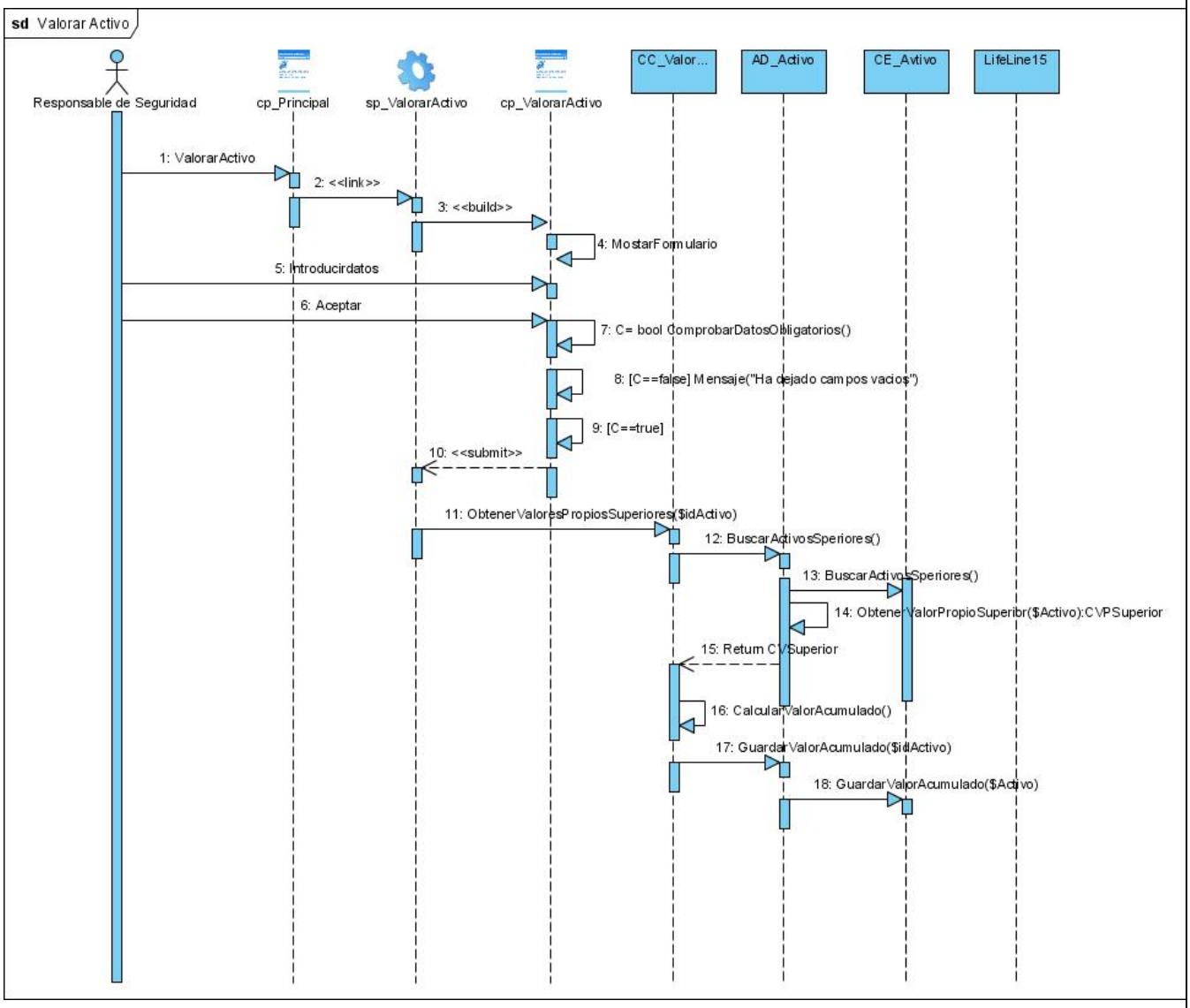


Diagrama de secuencia CU Establecer Dependencia

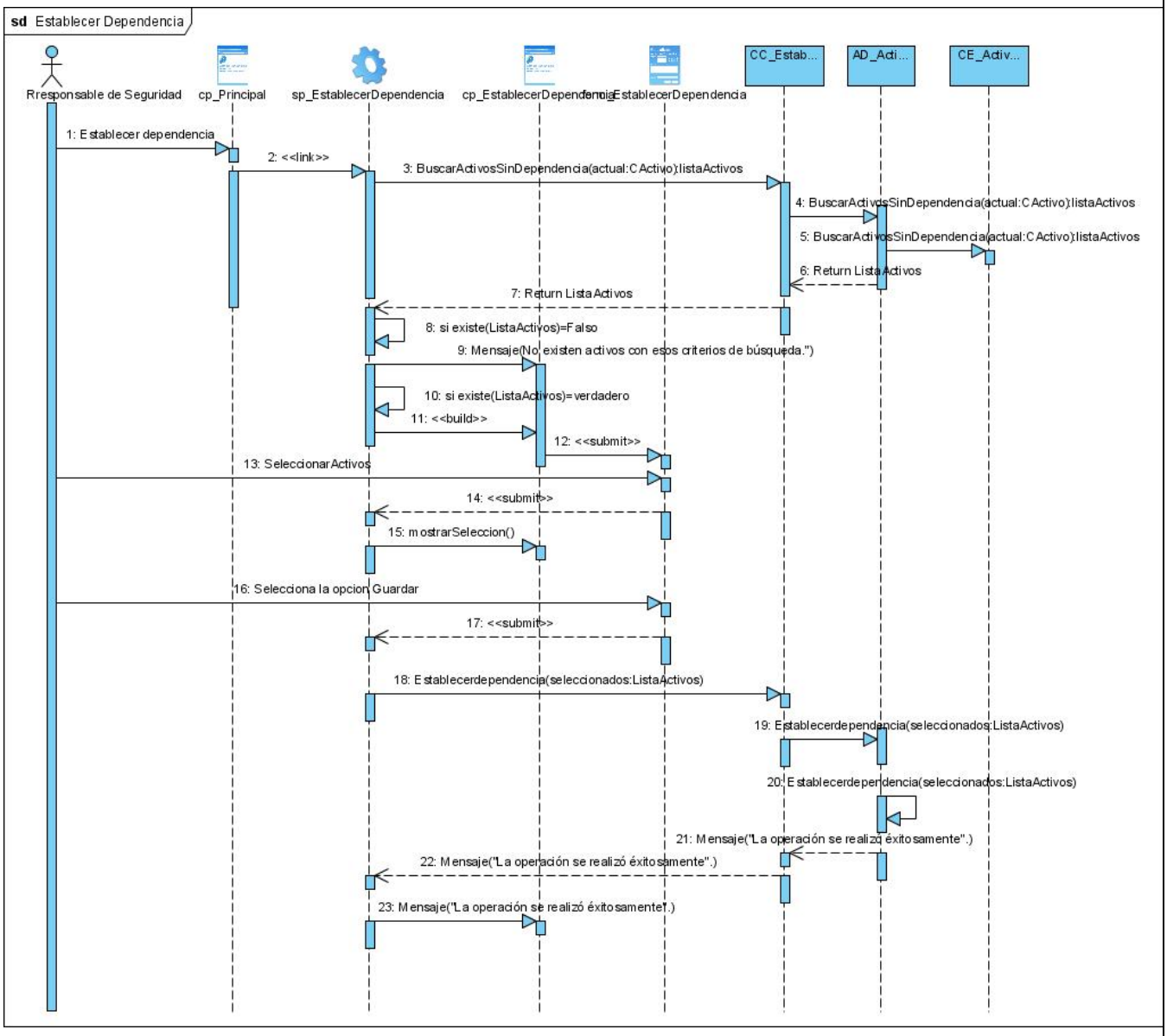


Diagrama de secuencia CU Identificar Amenaza

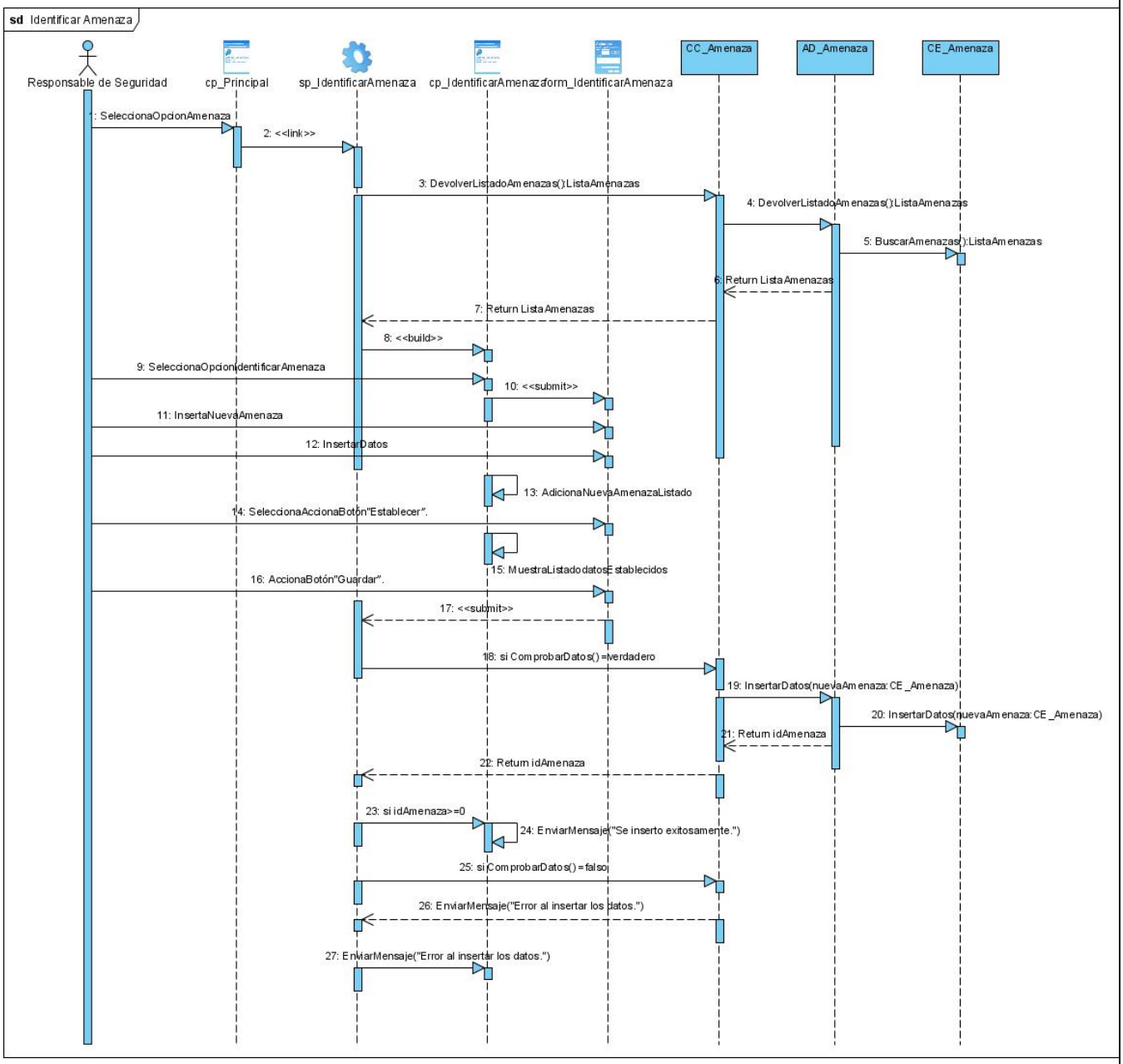


Diagrama de secuencia CU Valorar Amenaza

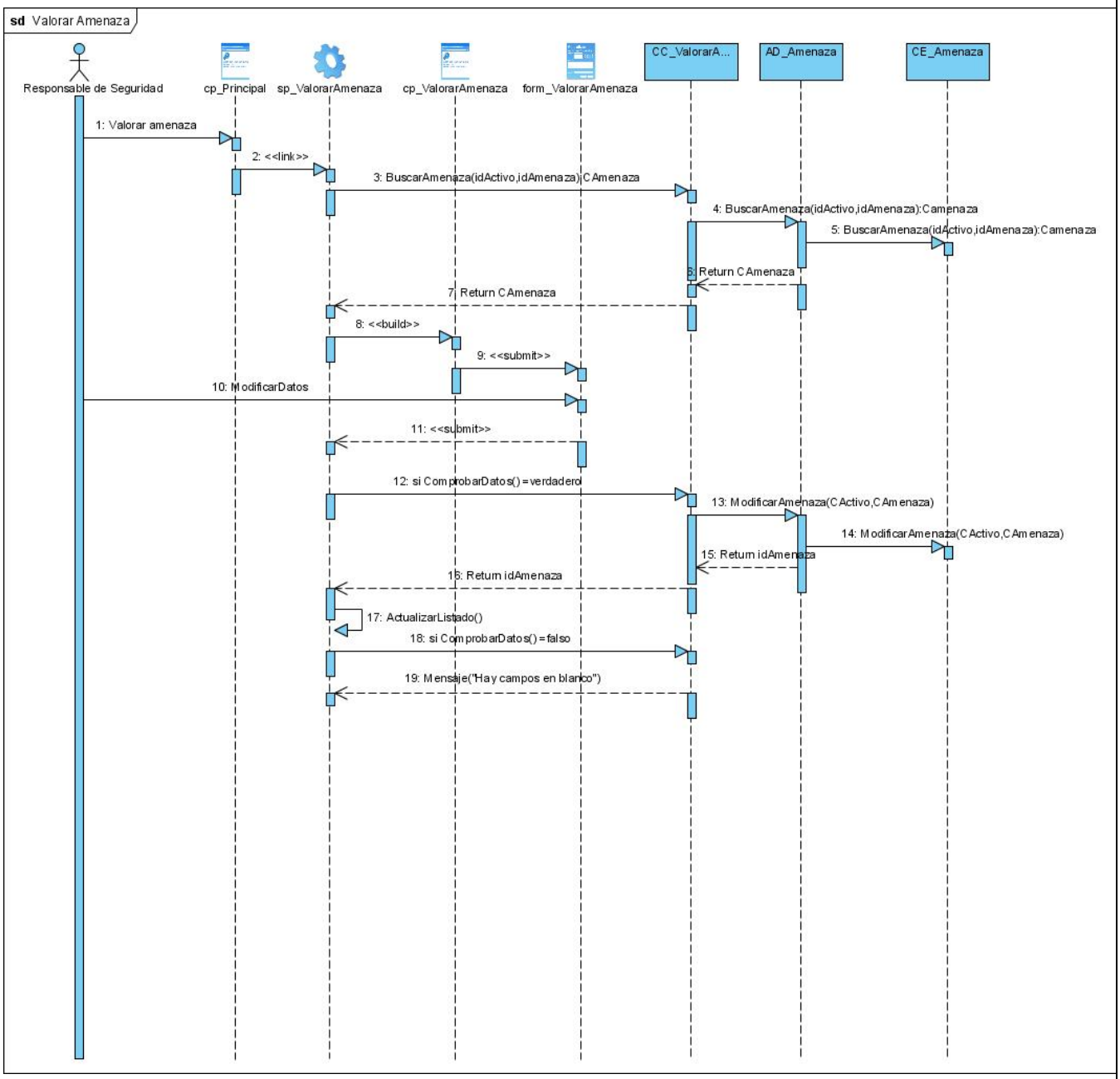


Diagrama de secuencia CU Identificar Salvaguarda

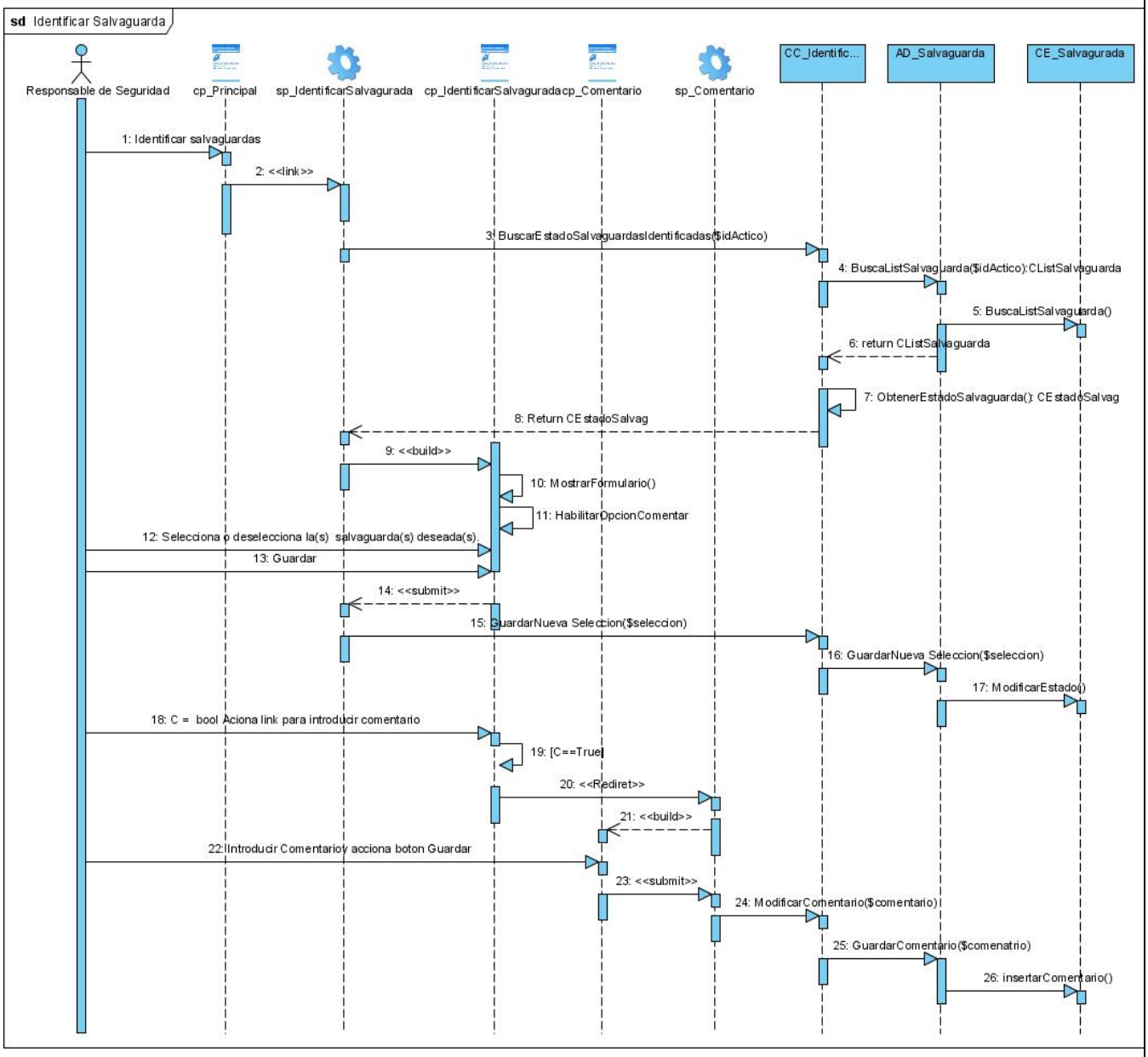
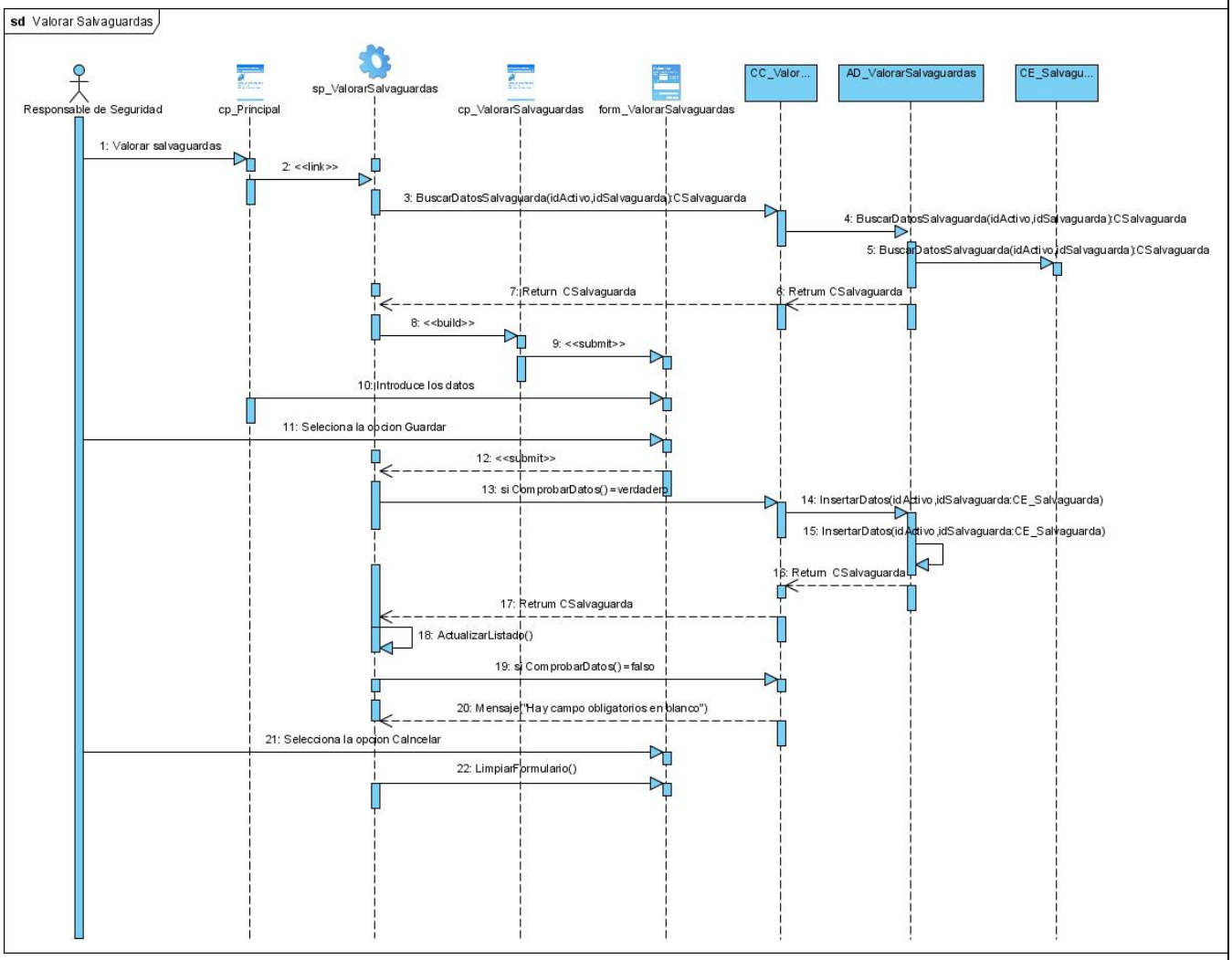


Diagrama de secuencia CU Valorar Salvagurada



GLOSARIO DE TERMINOS Y SIGLAS

CASE: *Computer Aided Software Engineering.*

Bienes Informáticos: Elementos componentes del sistema informático que deben ser protegidos en evitación de que como resultado de la materialización de una amenaza sufran algún tipo de daño.

Plan de Seguridad Informática: Documento básico que establece los principios organizativos y funcionales de la actividad de seguridad informática en una entidad.

Sistema de Seguridad Informática: Conjunto de medios humanos, técnicos y administrativos, que de manera interrelacionada garantizan diferentes grados de seguridad informática en correspondencia con la importancia de los bienes a proteger y los riesgos estimados.

Dominio: Un dominio de seguridad es una colección de activos uniformemente protegidos, típicamente bajo una única autoridad. Los dominios de seguridad se utilizan para diferenciar entre zonas en el sistema de información.

Por ejemplo:

- Instalaciones centrales, sucursales, comerciales trabajando con portátiles
- Central server, frontal unix, y PCs administrativos
- …

Cada activo pertenece a un dominio de seguridad.

PC: maquina u ordenador.

CGI: *Common Gateway Interface.*

MySQL: Es un sistema de gestión de bases de datos relacional que cuentan con todas las características de un motor de BD comercial: transacciones atómicas, triggers, replicación, llaves foráneas entre otras. Su ingeniosa arquitectura lo hace extremadamente rápido y fácil de personalizar.

PostgreSQL: es un Sistema de Gestión de Bases de Datos Objeto-Relacionales (ORDBMS) libre.

HTML: HyperText Markup Language. Lenguaje usado para escribir documentos para servidores World Wide Web. Es una aplicación de la ISO Standard 8879:1986. Es un lenguaje de marcas. Los lenguajes de marcas no son equivalentes a los lenguajes de programación aunque se definan igualmente como "lenguajes". Son sistemas complejos de descripción de información, normalmente documentos, que se pueden controlar desde cualquier editor ASCII.

Hypertext Transfer Protocol: Protocolo de Transferencia de Hipertextos. Modo de comunicación para solicitar páginas Web.