

UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS
FACULTAD 8

Análisis, diseño e implementación del submódulo
Solicitudes
perteneiente al módulo Investigación Criminalística
del Sistema de Investigación e Información Policial

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

Autores:

Luis Enrique Hernández Vega
Reinaldo Javier Rodríguez Font

Tutores:

Ing. Yadiel Ramos Rodríguez
Ing. Yudanis Gago Martínez

Ciudad de la Habana, 2009
“Año del 50 Aniversario del Triunfo de la Revolución”

Declaración de autoría

Declaramos que somos los autores de este trabajo y autorizamos a la Facultad 8 de la Universidad de las Ciencias Informáticas, así como a dicho centro para que hagan el uso que estimen pertinente con este trabajo.

Para que así conste firmamos la presente a los _____ días del mes de _____ del año 2009.

Autor

Reinaldo Javier Rodríguez Font

Autor

Luis Enrique Hernández Vega

Tutor

Ing. Yadiel Ramos Rodríguez

Tutor

Ing. Yudanis Gago Martínez

Agradecimientos

A la Revolución y a la Universidad de las Ciencias Informáticas por darnos la oportunidad de formarnos como profesionales.

A nuestros tutores, Yadiel y Yudanís, y a todo el equipo de trabajo, con los que compartimos numerosas horas de intenso trabajo y sin los cuales no habría sido posible llegar a donde estamos hoy.

- Reinaldo Javier:

A mis padres y abuela por sus sabias enseñanzas, su confianza, el cariño y la dedicación que me han brindado, por haber guiado mis pasos hasta convertirme en el hombre que soy.

A mi hermana, quien ha sido un ejemplo a seguir a lo largo de mis estudios.

A Ohannys, quien ha sabido comprenderme en estos años y a la que amo con toda mi vida.

A mis amigos de siempre Pablo, Arián y Shael, y a todos los que se han sumado; por haber vivido juntos tantos momentos alegres y difíciles para llegar hasta aquí.

- Luis Enrique:

A mis padres, por todo su amor y cariño, por sus enseñanzas, por ser tan comprensivos, por forjar el hombre que soy.

A mis abuelos; a mi abuelita del alma por criarme y quererme tanto; a mi abuelo del corazón, el tronco de los Hernández, por amar y querer tanto a sus nietos, por darme siempre tantos impulsitos y por haber logrado en vida guiarme en la realización de mis sueños.

A mi hermano mayor, por todo su cuidado, apoyo y cariño, siendo por siempre un guía mostrándome el camino a seguir.

A mis amigos de la UCI, Dayana, Carlos, Rosalba, Reinaldo, Eduardo, Lara, Raciél, por compartir tantos momentos juntos.

Dedicatoria

*A nuestros padres, porque hoy
también se hacen realidad sus sueños.*

Resumen

La Coordinación Nacional de Criminalística (CNC) es una institución adscrita al Cuerpo de Investigaciones Científicas, Penales y Criminalísticas de Venezuela, la cual tiene como función fundamental planificar, coordinar y dirigir todos los procesos técnico-científicos de las investigaciones. El objetivo de la investigación se centra en el análisis diseño e implementación de una porción de software que cumpla con los requisitos relacionados con las solicitudes criminalísticas atendidas por la CNC.

Para dar solución al problema planteado anteriormente se desarrolló una aplicación web, guiada en el proceso de desarrollo por las prácticas del Proceso Ágil Unificado y basada en patrones de diseño, haciendo uso de frameworks de desarrollo. La aplicación desarrollada cumple con los requisitos obtenidos del estudio del funcionamiento de la CNC y los estándares de calidad establecidos.

Abstract

The National Coordination of Criminalistics (CNC, by its acronym in Spanish) is a member of the Corps for Scientific, Penal, and Criminal Research, from Venezuela, which has as its central role to plan, coordinate and direct all the processes of technical-scientific research. The goal of the investigation is centered in the analysis, design and implementation of a software portion which complies with the requisites related to the criminalistic applications attended by the CNC.

To solve this problem, a Web application was developed, whose development process was guided by the Agile Unified Process practices and based in design patterns, using development frameworks. The developed application complies with the requisites obtained by studying how the CNC works, and the established software quality standards.

Índice

Introducción	1
Capítulo 1 Fundamentación teórica	4
1.1 Sistemas de Gestión de Información	4
1.1.1 Sistemas de Gestión de Información Policial.....	4
1.1.2 Sistema Integrado de Información Policial	5
1.1.3 Sistema de Investigación e Información Policial (SIIPOL)	5
1.2 Estudio de sistemas de interés	6
1.2.1 Sistema de Información Policial (SIP)	6
1.2.2 Sistema Territorial de Emergencias y Gestión Policial (STEGPOL)	7
1.2.3 Sistema de Gestión Penitenciaria (SIGEP)	7
1.2.4 Resultados obtenidos.....	8
1.3 Proceso de desarrollo de software	8
1.3.1 Proceso Unificado de Desarrollo (RUP)	9
1.3.2 Proceso Ágil Unificado (AUP)	11
1.3.3 Selección del proceso de desarrollo.....	13
1.4 Lenguaje de modelado	13
1.5 Lenguaje de programación	14
1.5.1 Java	14
1.6 Plataforma de desarrollo.....	15
1.7 Entorno de Desarrollo Integrado.....	16
1.7.1 Red Hat Developer Studio.....	16
1.8 Herramienta CASE	17
1.8.1 Visual Paradigm for UML	18

1.9 Frameworks utilizados en la solución	18
1.9.1 Presentación	19
1.9.2 Lógica	21
1.9.3 Acceso a datos	22
1.10 Sistema Gestor de Base de Datos.....	23
1.11 Arquitectura técnica.....	24
1.12 Conclusiones.....	26
Capítulo 2 Análisis, diseño e implementación de la propuesta de solución	27
2.1 Análisis del submódulo Solicitudes.....	27
2.1.1 Modelo de Casos de Uso.....	27
2.1.2 Diagrama de estado de la solicitud	28
2.1.3 Casos de Uso significativos	30
2.1.4 Requisitos no funcionales relevantes	31
2.2 Diseño del submódulo Solicitudes.....	32
2.2.1 Diagrama de paquetes.....	32
2.2.2 Patrones de diseño	34
2.2.3 Tarjetas CRC	35
2.2.4 Realización de los Casos de Uso.....	46
2.3 Modelo de Datos	48
2.3.1 Diagrama de clases persistentes	48
2.3.2 Diagrama de tablas del modelo relacional.....	50
2.4 Modelo de Implementación.....	51
2.4.1 Diagrama de subsistemas de implementación	51
2.4.2 Diagrama de componentes	52

2.4.3 Estándar de codificación	56
2.4.4 Interfaces de usuario.....	56
2.5 Conclusiones.....	60
Capítulo 3 Validación de la propuesta de solución	61
3.1 Métodos de prueba. Niveles de prueba.	61
3.1.1 Métodos de prueba	61
3.1.2 Niveles de prueba	61
3.2 Análisis de las pruebas realizadas.....	62
3.3 Conclusiones.....	70
Conclusiones	71
Recomendaciones	72
Bibliografía.....	73
Glosario de términos.....	75
Anexos.....	77
Anexo 1: Estructura organizacional del CICPC.....	77
Anexo 2: Tarjetas CRC para la capa de negocio del submódulo	78
Anexo 3: Resto de las interfaces de usuario del submódulo.....	79
Anexo 3.1: IU Incluir Solicitud de Experticia a Entidades Financieras	79
Anexo 3.2: IU Incluir Solicitud de Rollo Fotográfico.....	80
Anexo 3.3: IU Incluir Solicitud de Identificación de Individuo	81
Anexo 3.4: IU Tratar Reseñas para Solicitud – Editar Reseña	82
Anexo 3.5: IU Tratar Reseñas para Solicitud – Consultar Persona	83
Anexo 3.6: IU Completar Solicitud de Experticia de Avalúo	84
Anexo 4: Caso de Prueba Gestionar Solicitud de Experticia Criminalística	85

Índice de figuras

Figura 1-1: Fases y flujos de trabajo definidos por AUP.....	12
Figura 1-2: Flujo de petición utilizando Ajax4JSF.....	20
Figura 1-3: Composición de Spring.....	21
Figura 1-4: Arquitectura de Hibernate.....	22
Figura 1-5: Modelo Cliente-Servidor.....	24
Figura 1-6: Representación del patrón arquitectónico de 3 capas integrado a los frameworks.....	25
Figura 1-7: Manejo de la seguridad mediante Acegi.....	26
Figura 2-1: Diagrama de CU.....	28
Figura 2-2: Diagrama de estado de la Solicitud.....	29
Figura 2-3: Diagrama de paquetes.....	33
Figura 2-4: Tarjeta CRC para la jerarquía de Diligencia.....	35
Figura 2-5: Tarjeta CRC para las solicitudes de experticia criminalística.....	36
Figura 2-6: Tarjeta CRC para las solicitudes de revelado de rollo fotográfico.....	37
Figura 2-7: Diagrama de clases persistentes.....	49
Figura 2-8: Diagrama de tablas del modelo relacional.....	50
Figura 2-9: Diagrama de subsistemas de implementación.....	51
Figura 2-10: Diagrama de Componentes. Vista Acceso a Datos.....	52
Figura 2-11: Diagrama de Componentes. Vista Lógica de Negocio.....	53
Figura 2-12: Diagrama de Componentes. Vista Presentación.....	54
Figura 2-13: Diagrama de Componentes. Vista de las páginas.....	55
Figura 2-14: CU Tratar Sitios de Suceso para Solicitud.....	56
Figura 2-15: CU Tratar Evidencias para Solicitud.....	57
Figura 2-16: CU Tratar Personas para Solicitud.....	57
Figura 2-17: IU Incluir Solicitud de Experticia Criminalística.....	58
Figura 2-18: IU Incluir Solicitud de Experticia Criminalística - Elementos.....	59
Figura 2-19: IU Vista Previa Solicitud de Experticia Criminalística.....	60
Figura 3-1: No Conformidades detectadas durante las Pruebas Internas.....	65
Figura 3-2: No Conformidades detectadas durante las Pruebas Cruzadas.....	65
Figura 3-3: No Conformidades detectadas durante la 1ra iteración de las Pruebas de Liberación.....	66

Figura 3-4: No Conformidades detectadas durante la 2da iteración de las Pruebas de Liberación	66
Figura 3-5: No Conformidades detectadas durante las Pruebas de Aceptación	67
Figura 3-6: No Conformidades detectadas durante las Pruebas Piloto	68
Figura 3-7: Resumen general de las pruebas realizadas al submódulo	69

Introducción

El Cuerpo de Investigaciones Científicas, Penales y Criminalísticas (CICPC) de la República Bolivariana de Venezuela, es una institución que garantiza la eficiencia en la investigación del delito, mediante su determinación científica, asegurando el ejercicio de la acción penal que conduzca a una sana administración de justicia. Tiene la visión de ser la institución indispensable, por su reconocida capacidad científica y máxima excelencia de sus recursos, con la finalidad de alcanzar el más alto nivel de credibilidad nacional e internacional en la investigación del fenómeno delictivo organizado y criminalidad violenta. (1)

Dicha institución se compone en su estructura organizativa de varias instancias, como delegaciones, direcciones y coordinaciones (nivel de la estructura organizativa en el que se delegan procesos especializados), las que interactúan constantemente entre sí para dar cumplimiento a sus objetivos estratégicos (Ver Anexo 1). La Coordinación Nacional de Criminalística (CNC) es una institución que compone al CICPC, la cual tiene como función fundamental planificar, coordinar y dirigir todos los procesos técnico-científicos de las investigaciones. Esta coordinación se encarga de definir las políticas, normas, procedimientos y planes estratégicos que se implementan para el correcto trabajo investigativo, además de diseñar las estrategias que garanticen la actualización técnico-científica y definir métodos de estandarización en criminalística.

En la actualidad el CICPC cuenta con una aplicación informática conocida como Sistema Integrado de Información Policial, que posee una tecnología obsoleta y no brinda soporte a todos los procesos que se realizan a diario en la institución, factores que frenan su correcto funcionamiento y acarrearán ineficiencia en la primordial tarea de esclarecer los hechos delictivos.

La problemática existente en la Coordinación Nacional de Criminalística está determinada principalmente por la poca eficiencia que brinda el sistema gestor de información. Las entidades deben hacer las solicitudes de experticia por vía telefónica y muchas veces enviarse y esperar un documento físico, provocando lentitud en la fluidez de la información y pérdida de los datos en determinados casos. Gradualmente se van tornando más difíciles e imposibles los procesos de filtrado, procesamiento, consulta y recuperación de la información. (2)

Teniendo en cuenta los elevados índices de criminalidad mostrados en Venezuela, así como la insatisfacción en el cumplimiento de los objetivos y estrategias trazadas para revertir este indicador, se decide, en el marco de las relaciones entre Cuba y Venezuela, firmar un contrato con el fin de modernizar y automatizar el CICPC, en el cual una de las principales metas es el desarrollo de un nuevo Sistema Integrado de Información Policial, que perfeccione e incorpore nuevas funcionalidades al sistema existente, ofrezca una mejor gestión de la información, disminuya los tiempos de respuesta de las investigaciones, y cubra los procesos de la Coordinación Nacional de Criminalística.

Después de analizar el funcionamiento de la coordinación y sus procesos de negocio con el uso de la ingeniería de requerimientos, se obtuvieron los requisitos funcionales y no funcionales que se deben cumplir para lograr un correcto funcionamiento de la CNC, eliminando las deficiencias y no pocas limitaciones del antiguo Sistema Integrado de Información Policial.

Para desarrollar esta solución informática se ha modelado un sistema estructurado en módulos y submódulos con responsabilidades semejantes, agrupados según sus funcionalidades; entre ellos se encuentra el submódulo Solicitudes perteneciente al módulo Investigación Criminalística, el cual constituye la esencia de esta investigación.

Como centro de este trabajo se plantea el siguiente **problema científico**: ¿Cómo garantizar el cumplimiento de los requisitos funcionales y no funcionales asociados al submódulo Solicitudes del módulo Investigación Criminalística en el Sistema de Investigación e Información Policial?

El **objeto de estudio** establecido para la investigación se basa en el proceso de desarrollo de software en Sistemas de Gestión de Información Policial y el **campo de acción** se enmarca en el análisis, diseño e implementación del submódulo Solicitudes, perteneciente al módulo Investigación Criminalística.

El **objetivo general** trazado consiste en analizar, diseñar e implementar el submódulo Solicitudes perteneciente al módulo Investigación Criminalística del SIIPOL que satisfaga los requisitos funcionales y no funcionales obtenidos como resultado de la aplicación de la Ingeniería de Requerimientos a los procesos de la Coordinación Nacional de Criminalística.

Para darle cumplimiento al objetivo general se presentan los siguientes **objetivos específicos**:

1. Evaluar el uso de patrones de diseño e implementación en proyectos semejantes y de gran envergadura.

2. Valorar los resultados obtenidos por la Ingeniería de Requerimientos.
3. Implementar el submódulo Solicitudes perteneciente al módulo Investigación Criminalística del SIIPOL respetando la arquitectura definida.
4. Integrar el submódulo Solicitudes con el resto de los módulos del CICPC.

Por todo lo antes planteado **la idea a defender** es: con el análisis, diseño e implementación del submódulo Solicitudes se garantizará el cumplimiento de los requisitos funcionales y no funcionales obtenidos por la Ingeniería de Requerimientos.

Para alcanzar los objetivos propuestos las **tareas investigativas** a realizar son:

1. Investigar acerca de otras aplicaciones o soluciones similares.
2. Estudiar el proceso, las herramientas y lenguaje a utilizar en el análisis, diseño e implementación del submódulo Solicitudes perteneciente al módulo Investigación Criminalística del SIIPOL.
3. Estudiar la propuesta de Arquitectura para el SIIPOL y el modelo de Casos de Uso que da cumplimiento a los requisitos funcionales y no funcionales asociados al submódulo.
4. Investigar la aplicación de Patrones de Diseño.
5. Estudiar la especificación de los Casos de Uso del Sistema.
6. Elaborar los artefactos UML que sean necesarios.
7. Implementar los componentes necesarios que dan solución a los objetivos propuestos.
8. Probar los componentes desarrollados una vez que se integren al resto de los módulos del SIIPOL.
9. Validar la solución desarrollada.

El presente trabajo está estructurado en tres capítulos. El primero de ellos se ha denominado **Fundamentación Teórica** pues trata los elementos teóricos que soportan la investigación. El segundo capítulo, titulado **Análisis, diseño e implementación de la propuesta de solución**, constituye el centro de la investigación pues en él se describe la estrategia de solución. El tercer y último capítulo, **Validación de la propuesta de solución**, resumirá el conjunto de pruebas realizadas al software que demostrarán la calidad obtenida en su desarrollo.

Capítulo 1 Fundamentación teórica

En este capítulo se hace una revisión y análisis sobre los temas y conceptos más relevantes que conciernen al objeto de estudio del presente trabajo. Se definen y abordan valoraciones sobre las tendencias actuales en el mundo para darle solución a problemas como este, también se realiza el estudio del ambiente de desarrollo, brindando las principales características de la metodología, lenguaje y herramientas que se utilizarán en la implementación de la solución

1.1 Sistemas de Gestión de Información

Una definición muy acertada para un Sistema de Gestión de Información (SGI) podría consistir en un conjunto de elementos que interactúan entre sí con el fin de apoyar las actividades que se realizan en una organización o para automatizar los procesos de trabajo que se efectúan dentro de la misma. Un sistema de información se centra en estudiar las formas para mejorar el uso de la tecnología que soporta el flujo de información dentro de la organización.

El desarrollo de la tecnología y el aumento del cúmulo de información con que cuentan las empresas para tomar decisiones elevan el nivel de competitividad entre ellas y en este punto es donde los sistemas de gestión de información se hacen más útiles. La puesta en marcha de un sistema de este tipo proporciona cuantiosos beneficios, entre ellos la reducción del tiempo de búsqueda de documentos y la minimización de los riesgos de pérdida de archivos por estar en formato físico. Los SGI garantizan la integridad, seguridad e intercambio instantáneo de la información; disminuyen el costo de actividades elementales para las organizaciones como la elaboración de resúmenes y estadísticas al poderlas hacer de forma automática, con una interfaz agradable a los usuarios que interactúan con el sistema; aumentan la eficiencia de las empresas brindando más y mejores servicios a los clientes, lo que las hace más competitivas.

1.1.1 Sistemas de Gestión de Información Policial

En la actualidad, las instituciones más exitosas son las que han logrado adaptarse con una mayor rapidez al impetuoso desarrollo tecnológico y aprovechan las ventajas de la información. Agencias como la policía no pueden permanecer ajenas a estos cambios, máxime si su capacidad de procesar información determina en parte la calidad de los bienes públicos que ofrece. El aumento de los índices de criminalidad a nivel mundial hace que las instituciones policiales realicen planes estratégicos para su modernización a

partir del uso inteligente de la información. Por tanto se hace indispensable el uso de un sistema de gestión de información policial.

Un sistema de gestión de información policial se define como un sistema de gestión de información que se restringe a automatizar los procesos en entidades policiales, empleando un sistema de trabajo que facilita y soporta la generación de estrategias de prevención y combate contra el crimen, mediante el análisis sistemático de la información y la instrumentación de mecanismos efectivos de evaluación de resultados.

Los sistemas de gestión de información policial están diseñados para recolectar, almacenar, procesar y analizar toda la información proveniente de diferentes fuentes para permitir su procesamiento de manera eficaz. Su objetivo es contar con información útil, oportuna y eficiente para el desarrollo de estrategias, así como apoyar la toma de decisiones a todos los niveles, disminuyendo de esta forma el tiempo de respuesta para combatir los delitos.

Una correcta preparación del personal de las instituciones policiales y el uso de un sistema eficiente de gestión policial es la combinación perfecta para combatir los males que ponen en peligro la seguridad social y el orden interior de cada país.

1.1.2 Sistema Integrado de Información Policial

El Sistema Integrado de Información Policial es una aplicación informática que el Cuerpo de Investigaciones Científicas, Penales y Criminalísticas de la República Bolivariana de Venezuela utiliza para llevar a cabo las tareas fundamentales que realizan. Este sistema está desarrollado sobre una tecnología obsoleta: utilizando el lenguaje de programación Natural, Adabas como gestor de bases de datos y servidores SUN 6500.

Dicho software presenta una serie de limitaciones entre las que se destacan el lento procesamiento de la información, tiempo de respuesta demorado y no abarca todos los procesos que se realizan en la institución; problemas estos que afectan seriamente el trabajo de los funcionarios del CICPC que en demasiadas ocasiones no pueden llegar al fondo de las investigaciones. (2)

1.1.3 Sistema de Investigación e Información Policial (SIIPOL)

En el marco de cooperación del ALBA, Cuba y Venezuela firmaron un contrato con el objetivo de automatizar el Cuerpo de Investigaciones Científicas Penales y Criminalísticas. Como parte de la planificación general, se pretende desarrollar un nuevo sistema, tomando como referencia el sistema

existente, que incremente las posibilidades de uso, gestión de la información, agilice los tiempos de respuesta y abarque en una mayor cantidad y con mejor calidad los procesos ejecutados en cada una de las instituciones que componen a dicha entidad.

1.2 Estudio de sistemas de interés

Con el fin de fundamentar un estudio del tema tratado se realizó una búsqueda de aplicaciones informáticas que traten la gestión de información policial para tener un punto de partida al comenzar a desarrollar la propuesta. La principal característica de estos sistemas es su confidencialidad por lo que la información al respecto es escasa. A continuación se hace referencia a algunos de los sistemas relacionados a la gestión policial y criminalística.

1.2.1 Sistema de Información Policial (SIP)

La Secretaría de Seguridad Pública del Distrito Federal de México emprendió a partir del 2002 la Reforma de la Policía de la Ciudad, cuyo objetivo principal es la modernización de la operación policial a partir del uso inteligente de la información. Bajo esta premisa, la Dirección General de Estadística e Información Policial comenzó a desarrollar el Sistema de Información Policial. Este sistema es una nueva forma de trabajar de toda la policía con base en los siguientes principios:

- Uso sistemático de la información para diseñar estrategias y medir resultados.
- Seguimiento permanente del trabajo de cada unidad policial.

Entre las novedades que propone este sistema se encuentra la construcción de un sistema de indicadores de desempeño que permitirá monitorear por medio de un tablero de control la información crítica tanto operativa como administrativa y toda la relacionada con los delitos que se cometen diariamente en la ciudad. Un sistema para la captura de delitos, incidentes y detenidos que permite la generación de mapas de criminalidad y reportes estadísticos básicos. Implementar una red de voz, datos y video con tecnología IP que permitirá la trasmisión de información administrativa y delictiva en línea para la toma de decisiones. La red comunicará a 96 puntos en la ciudad que incluye la totalidad de los sectores y otros edificios policiales. El uso del mapeo computarizado como una herramienta de apoyo para el análisis criminal con vistas a determinar fácilmente patrones, tendencias delictivas y puntos críticos. Para mejorar los flujos de información y disminuir los errores de captura se diseñó un parte de novedades electrónico que permite sistematizar diariamente la información de los 70 sectores que componen la policía mexicana. (3)

1.2.2 Sistema Territorial de Emergencias y Gestión Policial (STEGPOL)

Es un Sistema de Información Geográfica con tecnología de punta sobre una Plataforma Nacional Común de Información aplicada al "Sistema de Emergencias Nacionales" y al "Sistema Territorial de Gestión Policial" que actualmente operan en Chile. Se caracteriza por ser un sistema con tecnología de punta aplicado al "control territorial de la gestión policial", el cual identifica de forma veraz y efectiva dónde geográficamente se están cometiendo o se han cometido actos delictivos a nivel territorial, apoyado con sistemas de información en línea desde el lugar de los hechos, que permitan la acción rápida y coordinada entre los diferentes actores encargados de la seguridad ciudadana a nivel de país.

Integra a las diferentes entidades como Carabineros, Investigaciones, Ministerio del Interior y Municipios, entre otros, en una Plataforma Nacional Común de Información que permita el intercambio de datos y que sirva de apoyo a la gestión operacional regional o comunal, donde dichas Instituciones estén interconectadas entre sí, en especial con diferentes Divisiones o Departamentos de Carabineros, tales como Jefaturas de Zona, Prefecturas, Comisarías, Sub-Comisarías, Tenencias y por último los Retenes, más el apoyo directo y coordinado entre Carabineros y el área de Seguridad de los Municipios.

El sistema interconecta un gran número de Comisarías dentro del país, para que todos tengan acceso a una misma Plataforma de Mapas Digitalizados, además de las otras Instituciones relacionadas al tema de Seguridad y sus respectivas Unidades Especializadas. Vía la Intranet del estado o vía comunicación en Banda Ancha - Internet se puede llegar a acceder al Sistema Territorial de Emergencias y Gestión Policial montado en la web con diferentes contraseñas de acceso restringido, tanto para los operadores encargados de ingresar o modificar datos en línea, como para aquellos que solo tengan acceso a consultar.

El sistema trata de darle un enfoque específico a la seguridad ciudadana integrando y mejorando el manejo de las emergencias y la gestión policial a lo largo del país, haciendo operar entre sí a todas las instituciones de gobierno que tienen relación con la seguridad de la población, teniendo al todo el territorio de la nación en línea sobre una misma Plataforma de Gestión Operacional. (4)

1.2.3 Sistema de Gestión Penitenciaria (SIGEP)

El Sistema de Gestión Penitenciaria es uno de los proyectos firmados en el marco de las relaciones Cuba-Venezuela que está actualmente en etapa de desarrollo. El mismo pretende agilizar la gestión de

información y organizar los procesos de la Dirección General de Custodia y Rehabilitación del Recluso (DGCRR).

El sistema forma parte del Proyecto de Humanización Penitenciaria impulsado por Hugo Chávez Frías, el cual recoge un plan de acción diseñado en función de tres componentes esenciales: Infraestructura y Tecnología, Transformación Institucional y Atención Integral al Interno. La solución tecnológica apunta hacia la modernización de los controles de acceso y la creación de una base de datos integral sobre los internos y sus familias, en el que están incluyendo indicadores como historial médico, antecedentes penales, cuándo y dónde le corresponden las audiencias a cada recluso, observaciones de conducta, además de las referencias básicas. (5)

1.2.4 Resultados obtenidos

A pesar de la poca información disponible acerca del funcionamiento de estos sistemas se pudo constatar que existen ciertos parámetros y funciones que todos cumplen por igual para lograr un servicio de calidad y constituir una verdadera carta de triunfo para las instituciones policiales en el combate contra el crimen. Entre estos elementos, con los que la aplicación a desarrollar deberá constar, se encuentra la generación de reportes y estadísticas para proveer una vista de los sectores más afectados según determinados criterios y poder dirigir los esfuerzos hacia las áreas que más lo necesiten; también facilita la toma de decisiones por parte de los directivos al tener elementos claves y suficientes datos reales con los que contar. Otra de las características que comparten los sistemas estudiados es el uso eficiente de la información a través de búsquedas ágiles y eficaces que reduzcan considerablemente el tiempo que demoran las investigaciones.

El software SIGEP se pudo estudiar más a fondo por desarrollarse aquí en la universidad y ser otro de los proyectos firmados con la República Bolivariana de Venezuela. Los principales elementos aportados por el estudio estuvieron determinados por el uso de Java como lenguaje de programación, este en calidad de lenguaje de programación libre y de código abierto; entre ellos, el desarrollo de una arquitectura basada en frameworks que se especializan en capas específicas de la aplicación.

1.3 Proceso de desarrollo de software

Un proceso de desarrollo de software puede definirse como un conjunto de herramientas, métodos y prácticas que se emplean para producir software con el objetivo de elevar la calidad de este, a través de la

producción eficaz de un producto software que reúna los requisitos del cliente. La elección del proceso de desarrollo para el SIIPOL no es del alcance de este trabajo, aún así se brindan elementos que justifican el estilo de trabajo seguido en el sistema.

1.3.1 Proceso Unificado de Desarrollo (RUP)

El Proceso Unificado de Desarrollo (Rational Unified Process en inglés) es uno de los procesos más generales que existen actualmente, su finalidad no está restringida a guiar el desarrollo de *software*, sino a cualquier tipo de proyecto. Es ampliamente usado por el personal de administración en procesos productivos pues intenta conseguir el objetivo común por medio del orden y la documentación. RUP está estructurado en varios flujos de trabajo que se evidencian a lo largo de cuatro fases (inicio, elaboración, construcción y transición).

Flujos de trabajo:

- Modelación del negocio: Describe los procesos de negocio, identificando quiénes participan y las actividades que requieren automatización.
- Requerimientos: Define qué es lo que el sistema debe hacer, para lo cual se identifican las funcionalidades requeridas y las restricciones que se imponen.
- Análisis y diseño: Describe cómo el sistema será realizado a partir de la funcionalidad prevista y las restricciones impuestas (requisitos), por lo que indica con precisión lo que se debe programar.
- Implementación: Define cómo se organizan las clases y objetos en componentes, cuáles nodos se utilizarán y la ubicación en ellos de los componentes y la estructura de capas de la aplicación.
- Prueba: Busca los defectos a lo largo del ciclo de vida.
- Instalación: Produce un entregable del producto y realiza actividades como empaque, instalación, asistencia a usuarios, etc. para entregar el *software* a los usuarios finales.
- Administración del proyecto: Involucra actividades con las que se busca producir un producto que satisfaga las necesidades de los clientes.
- Administración de configuración y cambios: Describe cómo controlar los elementos producidos por todos los integrantes del equipo de proyecto en cuanto a: utilización/actualización concurrente de elementos, control de versiones, etc.

- Ambiente: Contiene actividades que describen los procesos y herramientas que soportarán el equipo de trabajo del proyecto; así como el procedimiento para implementar el proceso en una organización.

Las principales bibliografías que abordan esta metodología coinciden en definir tres características fundamentales al hablar de RUP:

1. *Dirigido por casos de uso*: Los casos de uso reflejan lo que los usuarios futuros necesitan y desean, lo cual se capta cuando se modela el negocio y se representa a través de los requisitos. A partir de aquí los casos de uso guían el proceso de desarrollo, ya que los modelos que se obtienen, como resultado de los diferentes flujos de trabajo, representan la realización de los casos de uso.
2. *Centrado en la arquitectura*: La arquitectura muestra la visión común del sistema completo en la que el equipo de proyecto y los usuarios deben estar de acuerdo, por lo que describe los elementos del modelo que son más importantes para su construcción, los cimientos del sistema que son necesarios como base para comprenderlo, desarrollarlo y producirlo económicamente. RUP se desarrolla mediante iteraciones, comenzando por los CU relevantes desde el punto de vista de la arquitectura. El modelo de arquitectura se representa a través de vistas(4+1), o sea perspectivas del sistema, que logran una abstracción particular en cada uno de los casos, en otras palabras, se trata de que en cada una de las llamadas vistas se represente el sistema en su totalidad teniendo en cuenta solo determinados aspectos.
3. *Iterativo e incremental*: RUP propone que cada fase se desarrolle en iteraciones. Una iteración involucra actividades de todos los flujos de trabajo, aunque desarrolla fundamentalmente algunos más que otros. Las iteraciones hacen referencia a pasos en los flujos de trabajo, y los incrementos, al crecimiento del producto.

La aplicación de RUP ha resultado exitosa en cientos de proyectos pero como metodología pesada concentra sus mayores esfuerzos en cuestiones administrativas de planificación, control y elaboración de documentación facilitando así, mayormente, el trabajo de líderes y gerentes, pero sacrificando un poco las tareas de diseño y codificación. Estas cuestiones han hecho que se tomen en cuenta nuevos métodos de trabajo que apuesten por el desarrollo ágil, como vía para lograr un software de calidad, donde clientes y equipo de desarrollo queden satisfechos, a través de metodologías que gestionen una cantidad apropiada de procesos y más orientadas a los desarrolladores.

1.3.2 Proceso Ágil Unificado (AUP)

El Proceso Ágil Unificado (Agil Unified Process AUP), es una versión simplificada de RUP que describe de forma simple y fácil de comprender el uso de técnicas ágiles para el desarrollo de aplicaciones que permanezcan dentro de los conceptos de RUP.

AUP mantiene las cuatro fases propuestas por RUP (inicio, elaboración, construcción y transición), no así para los flujos de trabajo donde el Modelo, primer flujo establecido por AUP, abarca las disciplinas Modelo de Negocio, Requerimientos y Análisis y Diseño. (6)

Fases:

- Inicio: Se determina el alcance inicial del proyecto, se identifica una arquitectura candidata para el sistema y se obtiene la financiación inicial del proyecto así como la aceptación de los interesados.
- Elaboración: Se prueba la arquitectura del sistema.
- Construcción: Se construye un software de manera regular y gradual partiendo por aquellas funcionalidades que poseen la prioridad más alta en términos de necesidad del cliente.
- Transición: Se valida y despliega el sistema en el entorno de producción.

Flujos de Trabajo:

- Modelo: Este flujo de trabajo centra su atención en entender el negocio de la organización, el problema por el cual surgió el proyecto, y determinar una solución viable para resolver el problema.
- Implementación: El objetivo de este flujo de trabajo es transformar los modelos en código ejecutable y desarrollar un nivel básico de pruebas, particularmente pruebas de unidad.
- Prueba: Se persigue realizar una evaluación objetiva para garantizar la calidad del producto. Esto incluye la búsqueda de defectos, validar que el sistema funcione tal como está establecido, y verificar que se cumplan los requisitos.
- Despliegue: Esta disciplina establece un plan para la entrega del sistema de manera que los usuarios finales reciban un software completamente funcional.
- Gestión de Configuración: Se administra el acceso a los artefactos producidos como resultado del proceso de desarrollo del software, así como la actualización y el control de versiones.
- Gestión de Proyecto: La finalidad de este flujo consiste en dirigir las actividades que tienen lugar dentro del proyecto, esto incluye administración de riesgos y dirección del personal, así como la

coordinación entre personas y sistemas fuera del marco del proyecto con el objetivo de garantizar la entrega del sistema en tiempo y sin exceder el presupuesto.

- Ambiente: Dispone actividades que describen los procesos y herramientas (hardware, software, etc.) que soportarán el equipo de trabajo del proyecto.

El Proceso Ágil Unificado está basado en los siguientes principios:

- Simplicidad: Se genera toda la documentación necesaria usando solo unas cuantas páginas, no miles de ellas.
- Agilidad: AUP se ajusta a los valores y principios del desarrollo ágil de software y de la Alianza Ágil
 - Individuos e interacciones más que procesos y herramientas.
 - Software operante más que documentaciones completas.
 - Colaboración con el cliente más que negociaciones contractuales.
 - Respuesta al cambio más que apegarse a una rigurosa planificación.
- Centrado en actividades de gran valor: La atención se centra en las actividades que en realidad cuentan, no en cada posible situación que podría suceder en el proyecto.

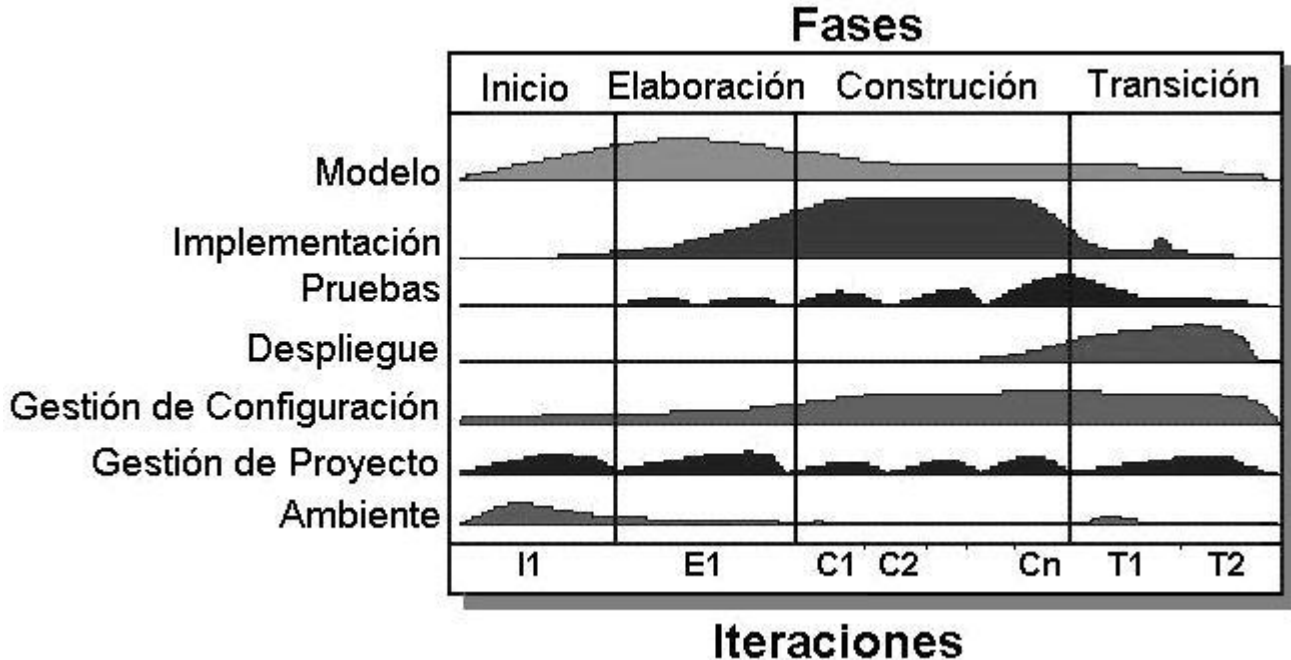


Figura 1-1: Fases y flujos de trabajo definidos por AUP

1.3.3 Selección del proceso de desarrollo

Cada uno de los procesos puede ser adaptado a cada proyecto, pero teniendo en cuenta las características del Proceso Ágil Unificado, que como se mencionó anteriormente, incorpora elementos ágiles para el desarrollo a los ya establecidos por RUP, este se convierte en la mejor opción para llevar a cabo el desarrollo del submódulo Solicitudes. Esto se debe a que en un equipo de trabajo pequeño formado en su mayoría por desarrolladores, el uso una metodología ágil ha probado a nivel mundial ser la mejor vía para el desarrollo de un software que satisfaga los requisitos del usuario y reúna estándares aceptables de calidad.

Además AUP, como está basado en los principios del desarrollo ágil, garantizará una rápida reacción ante los cambios ocasionados en los requisitos del cliente, algo que suele ocurrir si se tiene en cuenta que se están automatizando procesos policiales; centrará los esfuerzos del equipo de desarrollo en las tareas de mayor valor, dedicando especial atención a la excelencia técnica y al buen diseño. El uso del Proceso Ágil Unificado hará más eficiente al personal a través de mejoras frecuentes de los procesos adoptados por el equipo.

1.4 Lenguaje de modelado

El Lenguaje Unificado de Modelado (UML, por sus siglas en inglés, Unified Modeling Language) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad; es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema. Ofrece un estándar para describir un modelo del sistema, incluyendo aspectos conceptuales tales como procesos de negocio y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes reutilizables.

UML está compuesto por diversos elementos gráficos que se combinan para conformar diagramas; es un lenguaje por lo que cuenta con reglas para combinar tales elementos. Es importante recalcar que UML no es una guía para realizar el análisis y diseño orientado a objetos, es decir, no es un proceso. UML es un lenguaje que permite la modelación de sistemas con tecnología orientada a objetos. (7)

UML describe lo que supuestamente hará un sistema, pero no dice cómo implementar dicho sistema; está formado por distintos diagramas con la finalidad de presentar diversas perspectivas de un sistema, a las

cuales se les conoce como modelo. Estos diagramas se clasifican jerárquicamente como se muestra a continuación:

- **Diagramas de Estructura:** Enfatizan en los elementos que deben existir en el sistema modelado; incluye, entre otros, Diagrama de clases, Diagrama de componentes, Diagrama de objetos y Diagrama de paquetes.
- **Diagramas de Comportamiento:** Enfatizan en lo que debe suceder en el sistema modelado; ejemplo son el Diagrama de actividades y Diagrama de casos de uso.
- **Diagramas de Interacción:** Son un subtipo de diagramas de comportamiento, que enfatiza sobre el flujo de control y de datos entre los elementos del sistema modelado; aquí se puede ver el Diagrama de secuencia y de Diagrama de colaboración.

1.5 Lenguaje de programación

Un lenguaje de programación es un conjunto de símbolos, reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos y expresiones. Es utilizado para controlar el comportamiento físico y lógico de una máquina.

1.5.1 Java

Java es un lenguaje de programación desarrollado por Sun Microsystems Inc, con el objetivo de implementar una máquina virtual y un lenguaje con una estructura y sintaxis similar a C++. Proporcionando un lenguaje independiente de la plataforma y un entorno de ejecución ligero y gratuito, de forma que las aplicaciones Java pudiesen ejecutarse en cualquier plataforma. (8) Las características principales de Java respecto a cualquier otro lenguaje de programación, son: (9)

- Simple: Ofrece toda la funcionalidad de un lenguaje potente, pero sin las características menos usadas y más confusas de éstos. Java elimina muchas de las características de otros lenguajes como C++, para mantener reducidas las especificaciones del lenguaje.
- Orientado a objetos: Java trabaja con sus datos como objetos y con interfaces a esos objetos. Soporta las tres características propias del paradigma de la orientación a objetos: encapsulación, herencia y polimorfismo.

- Robusto: Realiza verificaciones en busca de problemas tanto en tiempo de compilación como en tiempo de ejecución. Maneja la memoria para eliminar las preocupaciones por parte del programador de la liberación o corrupción de memoria.
- Seguro: Las aplicaciones de Java no acceden a zonas delicadas de memoria o de sistema, con lo cual evitan la interacción virus. Java no posee una semántica específica para modificar la pila de programa, la memoria libre o utilizar objetos y métodos de un programa sin los privilegios del sistema operativo.
- Portable: Como el código compilado de Java es interpretado, un programa compilado de Java puede ser utilizado por cualquier computadora que tenga implementado el interprete de Java.
- Arquitectura neutral: El compilador Java compila su código a un fichero objeto de formato independiente a la arquitectura de la máquina en que se ejecutará. Al compilar un programa en Java, el código resultante es un tipo de código binario conocido como byte code. Este código es interpretado por diferentes computadoras de igual manera, solamente hay que implementar un intérprete para cada plataforma. De esa manera Java logra ser un lenguaje que no depende de una arquitectura computacional definida.
- Dinámico: No requiere que se compilen todas las clases de un programa para que este funcione. Al realizar una modificación a una clase, Java se encarga de realizar un Dynamic Binding o un Dynamic Loading para encontrar las clases.

Java se distingue como un lenguaje de programación adecuado para aplicaciones empresariales y de gran envergadura. Con fines de garantizar un sistema portable, robusto y seguro es seleccionado para el desarrollo del SIIPOL.

1.6 Plataforma de desarrollo

Java 2 Plataforma de Edición Empresarial (J2EE por sus siglas en inglés), es una plataforma de programación para desarrollar y ejecutar software de aplicaciones en lenguaje de programación Java. Basándose ampliamente en los componentes de software y ejecutándose sobre un servidor de aplicaciones. J2EE configura algunas especificaciones únicas para componentes de Java y varias tecnologías de servicios web, permitiendo que el servidor de aplicaciones pueda manejar transacciones, la seguridad, escalabilidad, concurrencia y gestión de los componentes desplegados. La idea principal de

J2EE es la de proveer un estándar simple y unificado para aplicaciones distribuidas a través de un modelo de aplicaciones basado en componentes. (10) El modelo de J2EE es el siguiente:

- La aplicación es asociada a un conjunto de clases con la cual se puede definir objetos. Este conjunto de clases se obtiene como paquete complementario al conjunto de clases que en un principio ofrece Java para implementar y consumir servicios web.
- La funcionalidad de la aplicación en sí, o también definida como la lógica de la aplicación, es construida utilizando componentes EJB (Enterprise JavaBeans). Estos componentes proporcionan acceso a base de datos mediante JDBC (Java DataBase Connectivity).
- Cualquier aplicación puede conectarse a una aplicación J2EE utilizando la tecnología de servicios web.

J2EE define un estándar para el desarrollo de aplicaciones empresariales multicapa y simplifica las mismas basándolas en componentes modulares y estandarizados, provee un completo conjunto de servicios a estos componentes, y manejan muchas de las funciones de la aplicación de forma automática, sin necesidad de una programación compleja. Por estas características es seleccionado para el desarrollo del SIIPOL.

1.7 Entorno de Desarrollo Integrado

Una vez seleccionada la plataforma de desarrollo J2EE y el lenguaje de programación Java, solo resta seleccionar un Entorno de Desarrollo Integrado (IDE, por sus siglas en inglés) compatible con la plataforma y lenguaje. Un IDE es un programa compuesto por un conjunto de herramientas que han sido empaquetadas, es decir, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica.

1.7.1 Red Hat Developer Studio

Red Hat Developer Studio combina productos aportados a Red Hat por Exadel en marzo de 2007 - Exadel Studio Pro, RichFaces y Ajax4jsf - con software de middleware Jboss como JBoss Seam e Hibernate dentro de un potente entorno de desarrollo para aplicaciones SOA, Ajax y Java empresariales. (11) Entre sus características más relevantes se encuentran:

- Un modelo de programación unificado: Developer Studio aumenta y proporciona nuevas herramientas alrededor de JBoss Seam para construir aplicaciones de manera sencilla y

consistente. Hoy, el tipo de aplicación típicamente dicta el modelo de programación que el desarrollador utiliza, lo que significa que deben aprender a utilizar diferentes modelos. JBoss, que actualmente está siendo estandarizado como Web Beans en el Java Community Process, ofrece un modelo unificado y sencillo para desarrollar cualquier clase de aplicación, eliminando la necesidad de múltiples modelos de programación.

- Potentes capacidades Ajax: Proporciona un entorno de desarrollo Ajax integrado y potente con JBoss Seam y JBoss Ajax4jsf, componentes Web JBoss RichFaces, y herramientas What You See Is What You Get (WYSIWYG) para crear interfaces y páginas web que soportan Ajax.
- Utilidades Java Platform Enterprise Edition (EE): Developer Studio hace más sencilla la construcción de aplicaciones Java EE, con capacidades como WYSIWYG y edición de JavaServer Faces y páginas Facelets, asistencia de código dinámico y una paleta de componentes. Además, al incluir e integrar JBoss Application Server, Developer Studio simplifica el despliegue, la ejecución y la depuración de las aplicaciones Java EE.
- Un tiempo de ejecución integrado con las herramientas de desarrollo: Red Hat Developer Studio es el primer entorno de desarrollo open source basado en Eclipse que une el runtime con las herramientas. Esto elimina la necesidad de que el desarrollador improvise estructuras y componentes open source antes de que empiece a desarrollar.

Con vistas a maximizar la productividad del equipo de trabajo se decidió escoger Red Hat Developer Studio como IDE de desarrollo por las funcionalidades y mejoras para el desarrollo que ofrece para el programador.

1.8 Herramienta CASE

CASE son las siglas correspondientes a Computer Aided Software Engineering, que en su traducción al español significa Ingeniería de Software Asistida por Computadoras. Las herramientas CASE son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el costo de las mismas en términos de tiempo y dinero. (12)

Están destinadas a:

- Mejorar la productividad en el mantenimiento y desarrollo del software.

- Automatizar, desarrollo del software, documentación, generación de código, pruebas de errores y gestión del proyecto.
- Mejorar el tiempo y costo de desarrollo y mantenimiento de los sistemas informáticos.
- Aumentar la calidad del software.
- Facilitar el uso de las distintas metodologías propias de la ingeniería del software.

1.8.1 Visual Paradigm for UML

Visual Paradigm es una herramienta CASE que usa UML como lenguaje de modelado, permite la generación automática de reportes en formato pdf y html, el reconocimiento de artefactos de ingeniería a partir de reconocimiento de textos formales o informales, implementa una actualización automática del modelo de diseño y código permitiendo mantener la documentación de ambos modelos actualizadas con los cambios que ocurran en ambos sentidos, optimizando la descripción textual de elementos de código a partir de la descripción visual. Es capaz de importar y exportar elementos de otras herramientas CASE como Rational Rose y presenta una interfaz de uso intuitiva y con increíbles facilidades a la hora de modelar los diagramas que soportan la Ingeniería de Requerimientos. (13)

Visual Paradigm constituye un candidato más fuerte para la elección de una herramienta CASE para el proyecto CICPC, debido principalmente a que iguala a Rational Rose en cuanto a capacidades y es mucho más integrable con el resto de las herramientas y estándares escogidos como Eclipse, Hibernate, Subversion, todas aplicables para el desarrollo.

1.9 Frameworks utilizados en la solución

En general, el término framework, se refiere a una estructura de software compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación. En otras palabras, un framework se puede considerar como una aplicación genérica incompleta y configurable a la que pueden añadirse las últimas piezas para construir una aplicación concreta.

Java soporta gran número de frameworks especializándose cada uno de ellos en una parte específica del sistema, ya sea la capa de presentación, la capa de la lógica de negocio o la capa de acceso a datos. A continuación se presentan los principales frameworks por cada una de las capas lógicas de la aplicación que se seleccionaron para integrarlos y dar soporte a la arquitectura del proyecto.

1.9.1 Presentación

JavaServer Faces (JSF) es un framework que facilita la construcción de aplicaciones siguiendo el patrón Modelo Vista Controlador (MVC), modelo de componentes orientado a objetos, conversión de tipos, separación de responsabilidades, desarrolladores de componentes, desarrolladores de lógica de aplicación, montadores de páginas, poderoso sistema de navegación declarativa, uso de simples clases java como controladores, fácil incorporación de potencialidades Ajax, posee un conjunto prefabricado de componentes de interfaz de usuario, modelo de programación orientado a eventos. (14)

Las siguientes características diferencian a JSF de otros marcos de trabajo:

- Control de Beans de Respaldo: Los beans de respaldo son componentes JavaBeans asociados con componentes de interfaz de usuario (UI) utilizados en la página.
- Modelo de componentes UI extensible: Los componentes UI de JSF son elementos configurables, reutilizables que componen las interfaces de usuario de aplicaciones JSF.
- Modelo de Renderizado Flexible: Un renderizador (sistema que se encarga de convertir los componentes propios de JSF a componentes HTML clásicos) separa la vista y la funcionalidad de los componentes UI. Se pueden crear y utilizar varios renderizadores para definir diferentes apariencias del mismo componente para el mismo o diferentes clientes.
- Modelo de Conversión y Validación Extensible: Basados en los convertidores y validadores estándar, se pueden desarrollar convertidores y validadores personalizados, que proporcionan un mejor modelo de protección.

Ajax4JSF es una extensión Open Source del estándar JSF que se integra a este con gran facilidad, evita escribir código Java Scripts al brindar una amplia gama de componentes, permite agregar capacidades AJAX (del término en inglés Asynchronous Java Script and XML) incluso a aplicaciones JSF ya creadas. En fin, las prestaciones que brinda la poderosa combinación de JSF con AJAX, hacen que sea más atractiva por la comodidad que supone para el desarrollador y además por integrarse eficientemente con otros frameworks.

RichFaces es una librería de componentes para JSF y un avanzado framework para la integración de AJAX con facilidad en la capacidad de desarrollo de aplicaciones empresariales. RichFaces facilita el desarrollo y permite la reusabilidad de componentes de una manera estándar y bastante bien documentada, pues los desarrolladores podrían crear sus propios componentes JSF artesanales con

funcionalidad Ajax implementada con scripts propios, sin embargo RichFaces posibilita la programación con un extenso número de componentes listos para utilizarse. RichFaces también aprovecha al máximo los beneficios de JSF framework incluyendo, la validación y conversión de instalaciones, junto con la gestión de estática y dinámica los recursos.

Ajax4jsf y RichFaces se integran totalmente en la arquitectura de JSF y hereda las funcionalidades de sus etiquetas dotándolas con tecnología Ajax de forma limpia y sin añadir código Javascript. Mediante los mismos podemos variar el ciclo de vida de una petición JSF, recargar determinados componentes de la página sin necesidad de recargarla por completo, realizar peticiones al servidor automáticas y control de cualquier evento de usuario.

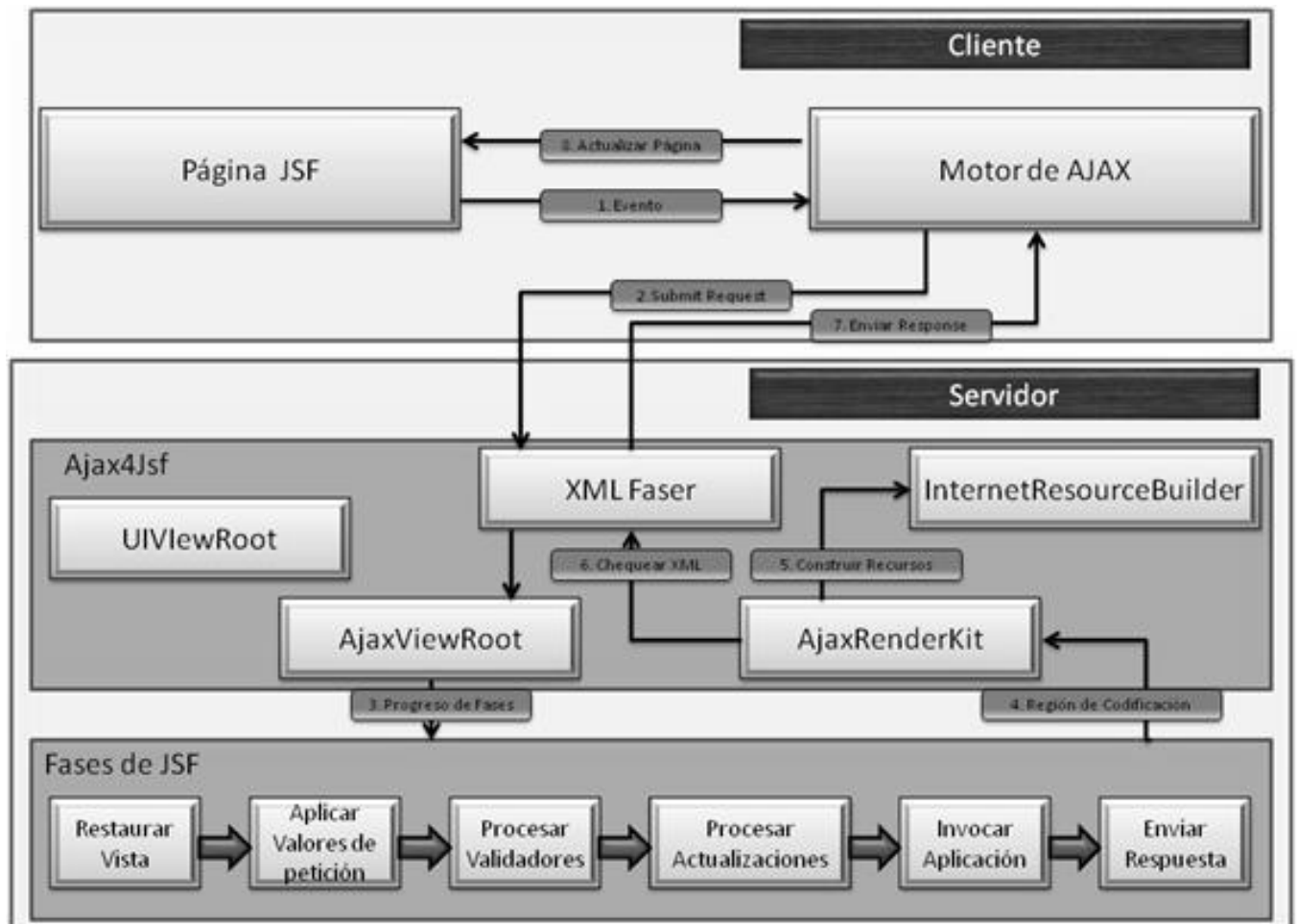


Figura 1-2: Flujo de petición utilizando Ajax4JSF

1.9.2 Lógica

Spring es un framework de código abierto para desarrollo de aplicaciones en plataforma J2EE. Spring interviene en todas las capas arquitectónicas de una aplicación J2EE, brinda soporte a varios frameworks de presentación, entre ellos Java Server Faces (JSF), y se integra con varios frameworks de acceso a datos. (15)

Spring promueve el bajo acoplamiento de las clases conformantes de la solución por medio de la técnica conocida como inversión del control (IoC). Soporta la Programación Orientada a Aspectos (AOP) que complementa la Programación Orientada a Objetos (POO), proponiendo otra manera de pensar sobre la estructura de un programa. Mientras que la POO descompone las aplicaciones en una jerarquía de objetos, la AOP descompone los programas en aspectos o preocupaciones. Dichas preocupaciones se convierten en servicios del sistema, separados de la lógica de negocio y que se ejecutan de manera transversal a la funcionalidad base, lo que proporciona una definición de responsabilidades superior. Spring básicamente es un contenedor, que se encarga de gestionar y administrar el ciclo de vida y configuración de las clases de la aplicación.

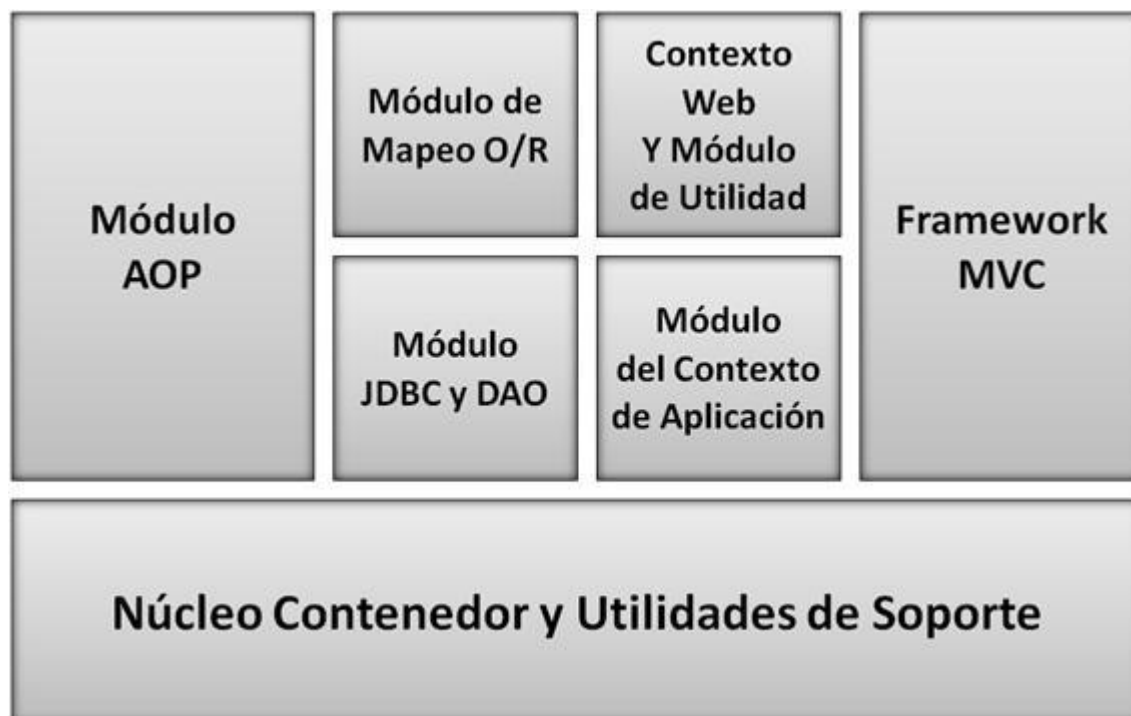


Figura 1-3: Composición de Spring

1.9.3 Acceso a datos

Hibernate es una solución ORM (Mapeo Objeto-Relacional) para Java, como todas las herramientas de su tipo. Hibernate busca solucionar el problema de la diferencia entre el modelo orientado a objetos y el usado en las bases de datos modelo relacional mediante archivos declarativos. Es una capa de persistencia objeto-relacional y un generador de sentencias SQL (Structured Query Language). Permite diseñar objetos persistentes que podrán incluir polimorfismos, relaciones, colecciones, y un gran número de tipos de datos. Hibernate convierte los datos entre los tipos utilizados por Java y los definidos por SQL.

Hibernate libera al desarrollador del manejo manual de los datos que resultan de la ejecución de sentencias SQL, manteniendo la portabilidad entre todas las bases de datos con un ligero incremento en el tiempo de ejecución. Hibernate permite expresar consultas en su propio lenguaje HQL (Hibernate Query Language), así como en SQL nativo, o con uno orientado a objetos. (16)

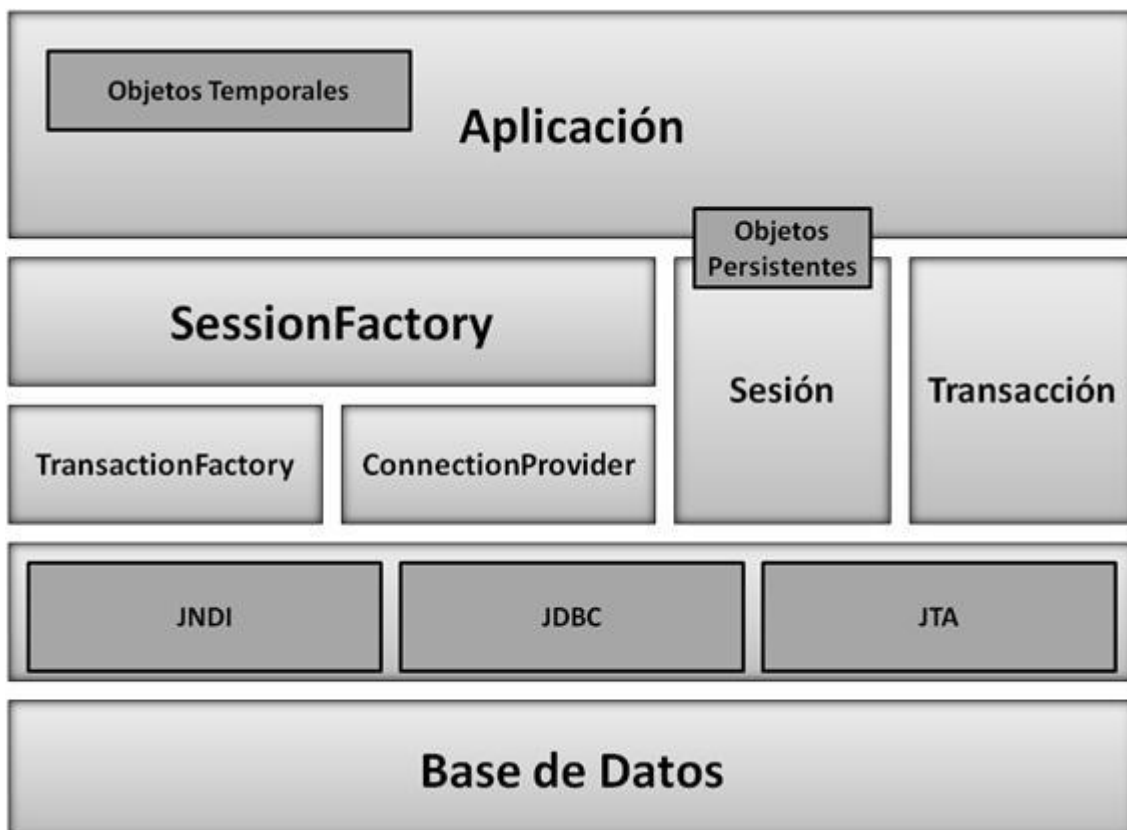


Figura 1-4: Arquitectura de Hibernate

1.10 Sistema Gestor de Base de Datos

Un Sistema Gestor de Bases de Datos (SGBD) es un conjunto de programas que permite a los usuarios crear y mantener una base de datos, por lo tanto, el SGBD es un software de propósito general que facilita el proceso de definir, construir y manipular la base de datos para diversas aplicaciones.

Oracle es un sistema gestor de base de datos relacional fabricado por Oracle Corporation. Se considera uno de los sistemas de bases de datos más completos, destacando su soporte de transacciones, estabilidad y escalabilidad. (17)

Es un producto vendido a nivel mundial, aunque la gran potencia que tiene y su elevado precio hacen que sólo se vea en empresas muy grandes y multinacionales, por norma general. En el desarrollo de aplicaciones web pasa lo mismo: como es un sistema muy caro no está tan extendido como otros, por ejemplo, Access, MySQL y SQL Server. Aún así fue seleccionado para la de modernización del CICPC y pagada su licencia dentro del presupuesto del proyecto, basando esta decisión en la seguridad que aporta y las muchas ventajas que posee, entre ellas:

- Puede ejecutarse en todas las plataformas, desde una simple computadora personal hasta un supercomputador utilizado como servidor. El software del servidor puede ejecutarse en multitud de sistemas operativos.
- Oracle soporta todas las funciones que se esperan de un buen servidor de bases de datos: un lenguaje de diseño de bases de datos muy completo (PL/SQL, lenguaje de quinta generación, muy potente para tratar y gestionar la base de datos) que permite implementar diseños con disparadores y procedimientos almacenados, con una integridad referencial declarativa muy potente.
- Permite el uso de particiones para la mejora de la eficiencia, de replicación e incluso ciertas versiones admiten la administración de bases de datos distribuidas.
- Este sistema ha comenzado a evolucionar en cuanto al mundo objetual, añadiendo tipos de clases, referencias, tablas anidadas, matrices y otras estructuras de datos complejas.

Oracle ha sido diseñado para que las organizaciones puedan controlar y gestionar grandes volúmenes de contenidos no estructurados en un único repositorio con el objetivo de reducir los costes y los riesgos asociados a la pérdida de información.

1.11 Arquitectura técnica

Como propuesta de arquitectura para el Sistema de Investigación e Información Policial se define una solución informática cliente-servidor de alto nivel que especifica una aplicación que hace uso del protocolo HTTP, con cuatro servidores HP Integrity. En los dos primeros estará instalado el servidor de aplicaciones, utilizando un failover y en los restantes un servidor de base de datos Oracle conectado en clúster los que estarán corriendo de manera continua; no obstante, de existir problemas en los mismos, existe un centro de contingencia el cual tiene dos servidores HP Integrity, uno para la aplicación web y otro para la base de datos.

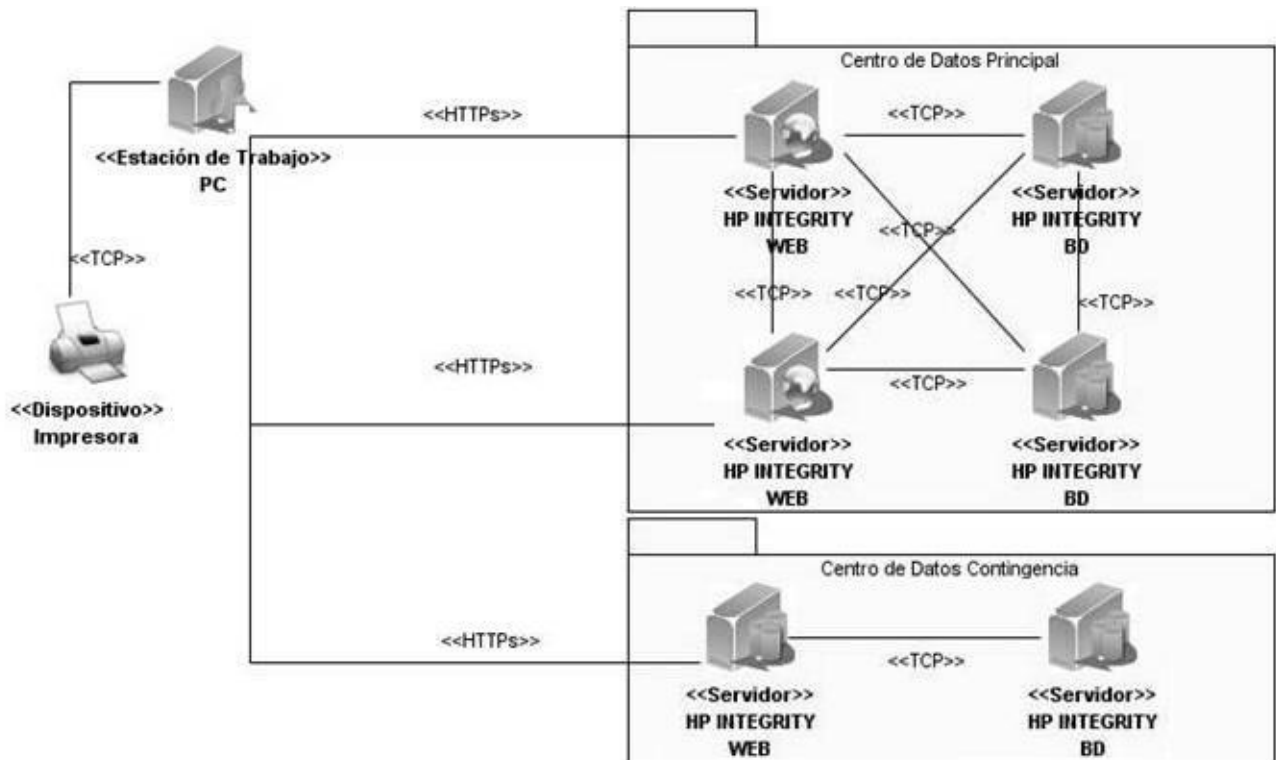


Figura 1-5: Modelo Cliente-Servidor

El uso de una arquitectura en capas permitirá desarrollar un sistema modular, escalable y de fácil desarrollo y mantenimiento, debido a las ventajas que este brinda, entre las que se encuentran una reutilización real del código, facilita la estandarización, las dependencias se limitan a las capas adyacentes lo que garantiza una contención de cambios a una o pocas capas, además centraliza los aspectos de seguridad.

Como patrón arquitectónico se elige MVC (Modelo Vista Controlador), el que permitirá una serie de ventajas a partir de su propia estructura: el **modelo**, que es la representación de la información que maneja la aplicación; la **vista**, que representa el modelo de forma gráfica disponible para la interacción con el usuario y el **controlador** que es el encargado de manejar y responder las solicitudes del usuario, procesando la información necesaria y modificando el modelo en caso de ser necesario. Entre las utilidades que brinda este patrón se encuentran la separación del modelo de la vista, es decir, separar los datos de la representación visual de los mismos, haciendo mucho más sencillo agregar múltiples representaciones de los mismos datos o información. Además facilita el mantenimiento de la aplicación, ofrece maneras más sencillas para probar el correcto funcionamiento del sistema, haciéndolo a la vez más escalable.

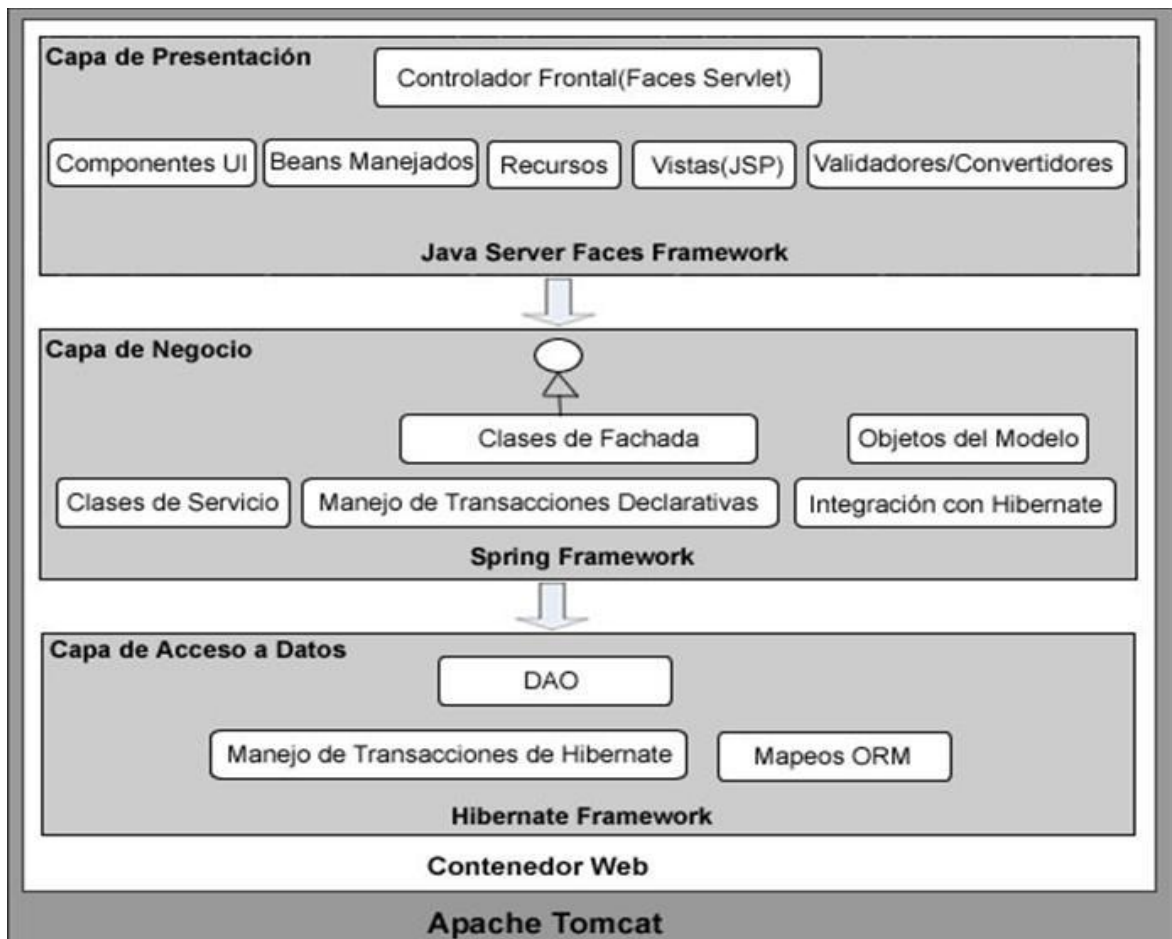


Figura 1-6: Representación del patrón arquitectónico de 3 capas integrado a los frameworks

La seguridad de la aplicación se garantiza con Acegi Security, un framework no intrusivo que abstrae al desarrollador de realizar tareas para asegurar el sistema, el acceso a la funcionalidad se maneja paralelamente a la misma a través de filtros que interceptan las peticiones web y determinan si se puede realizar. El aseguramiento de la aplicación no es del alcance de este trabajo, por lo que no se explica detalladamente.

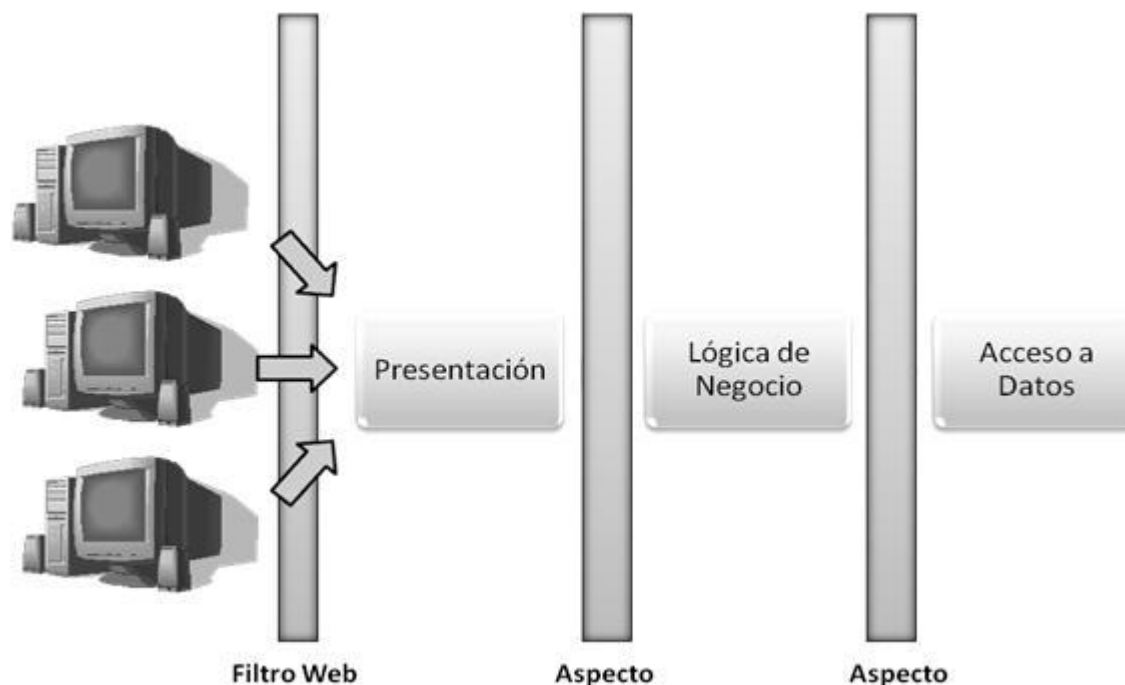


Figura 1-7: Manejo de la seguridad mediante Acegi

1.12 Conclusiones

En este capítulo se brindaron los conceptos necesarios para centrar la investigación en el marco de los sistemas gestores de información policial. Se hizo referencia al análisis de sistemas similares a nivel mundial con vistas a poder desarrollar un software basado en estándares internacionales y con la calidad requerida. Se realizó un estudio de las tecnologías, tendencias y procesos de desarrollo de software actuales, enunciando sus principales características y ventajas, lo que permitió seleccionar las herramientas para la construcción de la aplicación. Finalmente quedó expuesta la propuesta de solución para el Sistema de Investigación e Información Policial, siendo esta el punto inicial para el desarrollo del submódulo Solicitudes, centro de esta investigación.

Capítulo 2 Análisis, diseño e implementación de la propuesta de solución

En el presente capítulo se desarrollará la solución del submódulo Solicitudes propuesta anteriormente, a través de la realización de los artefactos UML necesarios para su comprensión y definidos por cada flujo de trabajo de la metodología seleccionada.

Primeramente se realiza un análisis del submódulo Solicitudes a partir de los resultados obtenidos por la ingeniería de requerimientos realizada por el equipo de analistas. Se mostrará el modelado e implementación del sistema a través del seguimiento de varios casos de uso considerados significativos y la generación de artefactos como modelos de datos, modelo de implementación, entre otros.

2.1 Análisis del submódulo Solicitudes

El módulo Investigación Criminalística brinda la posibilidad de acceder a las funcionalidades que permiten llevar a cabo las actividades que tienen que ver con la investigación criminalística. En él se podrá tener el control de las solicitudes de experticia criminalística que se pueden realizar, los resultados correspondientes a dichas solicitudes y los controles internos de cada despacho de la Coordinación Nacional Criminalística, así como el acceso a las funcionalidades de otros módulos, que permiten complementar las labores de carácter criminalístico.

En la solución, el módulo Investigación Criminalística se dividió en tres submódulos, llamados Experticias, Control de Investigaciones Criminalísticas y Solicitudes. El submódulo Solicitudes permite llevar a cabo el manejo de las solicitudes de experticia criminalística, dichas solicitudes dependen del tipo de experticia a realizar y de los elementos que se le puedan asociar. Cada Solicitud deberá ser respondida con una experticia, la cual mostrará todos los datos relevantes a lo que se solicitó.

2.1.1 Modelo de Casos de Uso

El diagrama de casos de usos mostrado a continuación es el resultado del análisis realizado a la documentación entregada por los analistas según define la metodología rectora a nivel central del proyecto (RUP). No obstante, debido a la extensión y complejidad de los casos de uso especificados (Gestionar Solicitud de Identificación de Individuo con 140 páginas), y a la necesidad de ir obteniendo partes funcionales para medir el avance de la implementación, se utilizaron los enfoques ágiles que define la metodología AUP, para estructurar los casos de uso de forma tal que se fueran generando resultados

de valor que garanticen la supervisión por parte de los expertos en el negocio. Los Casos de Uso se agruparon según sus funcionalidades y características.

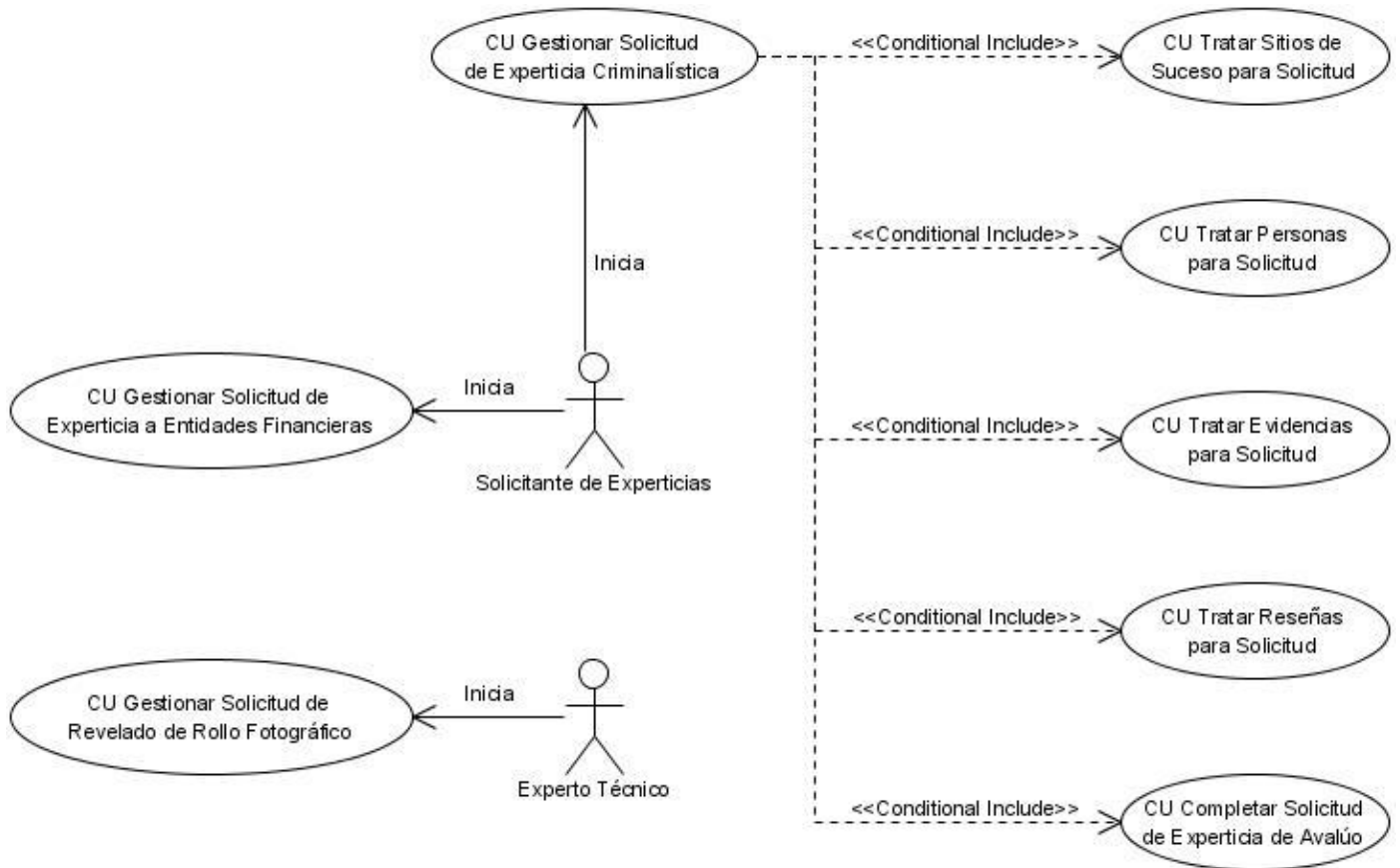


Figura 2-1: Diagrama de CU

2.1.2 Diagrama de estado de la solicitud

Para un mayor entendimiento del submódulo se hace necesario describir el flujo por el que pasan las solicitudes desde que se crean hasta que son respondidas por un experto del despacho al que la solicitud fue enviada. El proceso comienza cuando el solicitante de experticias crea una nueva solicitud, que siempre va dirigida a una dependencia específica del CICPC; la solicitud pasa por procesos de revisión y aprobación ejecutados por varios funcionarios dentro del despacho hasta que se determina que está correctamente elaborada y se envía al despacho destino. Una vez en su destino la solicitud es recibida y asignada a un especialista para que realice la experticia correspondiente al tipo de solicitud.

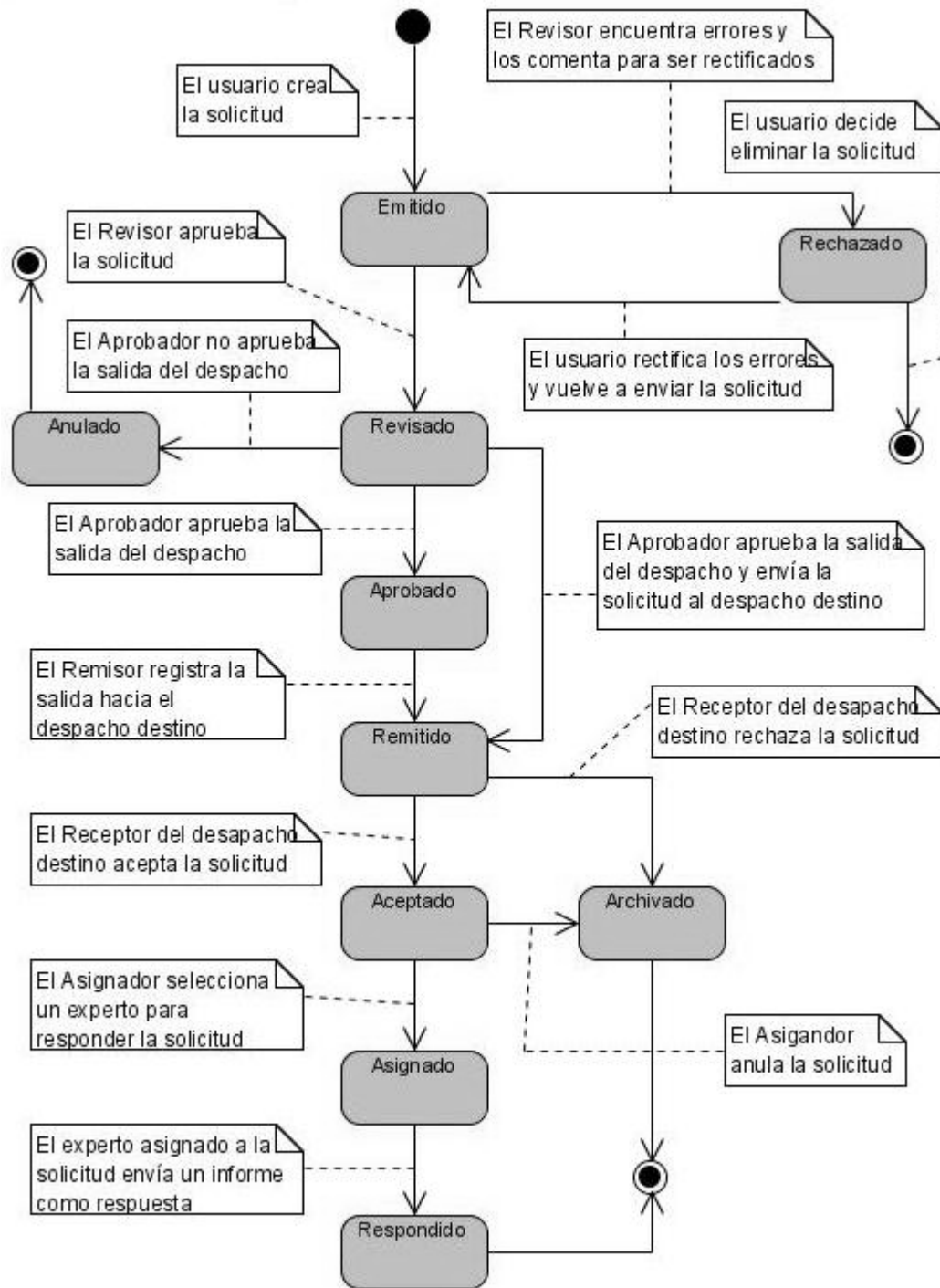


Figura 2-2: Diagrama de estado de la Solicitud

2.1.3 Casos de Uso significativos

Se han seleccionado los siguientes casos de uso como significativos pues ellos contienen la lógica necesaria para modelar e implementar el resto de los casos de uso pertenecientes al submódulo Solicitudes.

Nombre del CU	Gestionar Solicitud de Experticia Criminalística
Actor	Solicitante de Experticias
Descripción	<p>El caso de uso se inicia cuando el actor selecciona la opción que le permite realizar una acción sobre una Solicitud de Experticia Criminalística. El actor puede incluir, ver y modificar una Solicitud de Experticia Criminalística. En caso de que seleccione la opción de incluir una Solicitud de Experticia Criminalística, el sistema dará la posibilidad de insertar los datos que se necesitan para registrar una solicitud. Si el actor elige la opción de ver una Solicitud de Experticia Criminalística, el sistema mostrará el contenido de la solicitud en cuestión. Si el actor elige la opción de modificar una Solicitud de Experticia Criminalística, el sistema mostrará los datos que pueden ser editables dentro de la misma, y una vez realizados los cambios, guardará las modificaciones. El sistema permite ver una vista previa de una Solicitud de Experticia Criminalística, con posibilidad de imprimir y exportar a PDF.</p>
Referencia	<p>RF. Gestionar Solicitud de Experticia Criminalística</p> <ul style="list-style-type: none"> • Incluir solicitud • Ver datos de solicitud • Modificar datos de solicitud <p>RF. Asignar automáticamente el número único para cada solicitud.</p> <p>RF. Imprimir o exportar a PDF una Solicitud.</p> <p>RF. Validar la integridad de los datos introducidos por el usuario.</p>

Nombre del CU	Tratar Evidencias para Solicitud
Actor	Solicitante de Experticias
Descripción	El caso de uso se inicia cuando el actor accede a la opción que le permiten asociar y/o disociar evidencias a la solicitud que está creando. Si el origen de la solicitud es un Acta Procesal o un Informe el sistema brindará la posibilidad de asociar a la solicitud las evidencias pertenecientes a estos. Si la solicitud proviene de una Comunicación se permitirá incluir tantas evidencias como desee.
Referencia	RF. Asociar evidencias a la solicitud. RF. Disociar evidencias a la solicitud. RF. Incluir nuevas evidencias a la solicitud.

2.1.4 Requisitos no funcionales relevantes

Los requisitos no funcionales son propiedades o cualidades que el producto debe tener. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable (18). Los requisitos no funcionales asociados al submódulo Solicitudes son:

Seguridad

- La información en el sistema no puede ser eliminada.
- Un elemento es disociado solamente por el mismo actor que lo asoció o incluyó a la solicitud.
- Un elemento es eliminado si se disocia por el mismo actor que lo incluyó y no está relacionado a ninguna otra Acta Procesal, Informe o Solicitud.
- El sistema implementará el uso de campos obligatorios y validaciones para garantizar la integridad de la información que se introduce por el usuario.

Funcionalidad

- Los elementos asociados a una Solicitud de Experticia Criminalística tienen que pertenecer a la misma Acta Procesal.

Interfaz de Usuario

- Los errores serán visibles al usuario, marcando de rojo los campos con problemas.
- El sistema validará la información introducida por el usuario antes de almacenarla, ante la ocurrencia de un error, el sistema señalará los campos que generan el mismo, ya sea porque contienen información incompleta, o porque se encuentran vacíos.

2.2 Diseño del submódulo Solicitudes

Dentro del Modelo, primer flujo de trabajo definido por el Proceso Ágil Unificado, se realiza el diseño del software que se quiere implementar teniendo en cuenta aspectos generales de cada proceso de desarrollo de software como el uso de patrones de diseño, la satisfacción de los requisitos funcionales y no funcionales y las restricciones del entorno de implementación; agregando a estos, principios del desarrollo ágil como evitar el uso de documentación que genera trabajo y no aporta un valor directo al producto, y a su vez garantice la interacción entre los miembros del equipo de desarrollo.

Para lograr lo dicho anteriormente se optó por la realización de un diseño basado en tarjetas CRC, una técnica para la representación de sistemas orientados a objetos que brinda la posibilidad de modelar las responsabilidades de las clases y su interacción. El nombre CRC viene dado por clase, responsabilidad y colaboración, elementos que sus creadores vieron como las dimensiones esenciales del modelo orientado a objetos.

El uso de tarjetas CRC asegura la comunicación dentro del equipo de desarrollo posibilitando modelar un sistema donde todos los desarrolladores mantengan el mismo estilo de trabajo y establezcan la prioridad de las funcionalidades a implementar. La técnica CRC permite construir un sistema desde sus partes más simples hasta sus interacciones más complejas a partir de la unión de varias tarjetas; se pueden imprimir, lo que garantiza que puedan ser utilizadas en cualquier espacio de trabajo, no solo frente a una computadora.

2.2.1 Diagrama de paquetes

Los paquetes reflejan la arquitectura de alto nivel de un sistema: su descomposición en subsistemas y sus dependencias. Están organizados de manera funcional, siguiendo un cierto principio racional, tal como funcionalidad común, implementación estrechamente relacionada, y un punto de vista común. Una dependencia entre paquetes resume las dependencias entre los contenidos del paquete. (18)

A continuación se muestra el diagrama de paquetes del submódulo Solicitudes, que por el alcance de sus procesos guarda relación con la mayoría de los módulos del sistema.

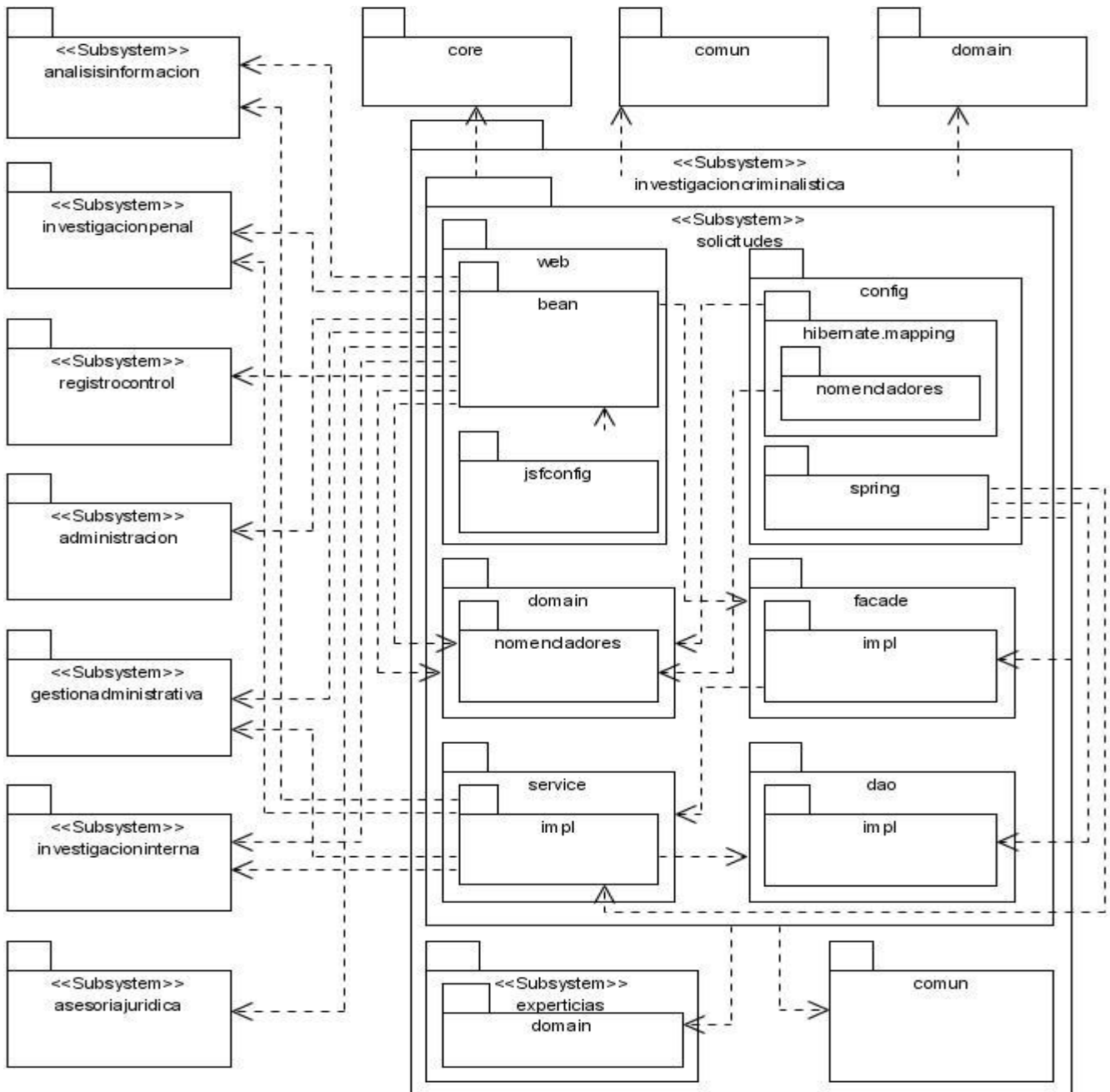


Figura 2-3: Diagrama de paquetes

2.2.2 Patrones de diseño

Los patrones de diseño son descripciones de clases cuyas instancias colaboran entre sí. Cada patrón es adecuado para ser adaptado para resolver un determinado tipo de problema en un contexto particular; permiten que algunos aspectos de la estructura de un sistema puedan cambiar independientemente de otros aspectos.

Con el fin de facilitar reusabilidad, extensibilidad y mantenimiento, se emplearon patrones de diseño en la solución propuesta. A continuación se exponen los patrones de diseño que se utilizaron y sus principales características.

El patrón de diseño **Front-Controller** (Controlador Frontal), proporciona un punto de entrada único que controla y gestiona las peticiones web realizadas por los clientes, incluyendo la invocación de servicios de seguridad como la autenticación y la autorización, delegando el procesamiento de negocios a los objetos responsabilizados. En la solución propuesta el mismo está implementado por el framework JSF, en este caso el controlador es la clase de respaldo de la página que recibe la petición, la procesa invocando al modelo, y actualiza la página o redirecciona hacia otra vista; se utiliza la variante de utilizar un controlador por cada página.

El patrón **Composite View** (Vista Compuesta) promueve la creación de una vista compuesta basada en la inclusión y sustitución de fragmentos de plantilla modulares tanto estáticos como dinámicos. También promueve la reutilización de porciones atómicas de la vista asegurando un diseño modular. Es apropiado utilizar este patrón para generar páginas que muestran componentes que podrían combinarse en una gran variedad de formas.

El patrón **Facade** (Fachada) se utiliza para proveer de una interfaz unificada sencilla que haga de intermediaria entre un cliente y una interfaz o grupo de interfaces más complejas. El objetivo principal de una fachada se basa en simplificar o hacer más amigable la complejidad asociada a un sistema, el conjunto de clases ó componentes que lo forman, centralizando las dependencias externas hacia el sistema en uno o pocos puntos de entrada. El resultado de la aplicación de dicho patrón en el submódulo Solicitudes es la fachada `investigacionCriminalisticaSolicitudesFacade`, la cual será el punto de entrada a cada una de las funcionalidades del subsistema.

El patrón **Service Locator** propone la creación de un punto de entrada común para la instanciación de los servicios que se desean invocar. Permite que varios clientes puedan reutilizar el objeto Service Locator para reducir la complejidad del código, proporcionando un punto de control.

El patrón **DAO** (Data Access Object u objetos de acceso a datos en español), consiste en utilizar un objeto de acceso a datos para abstraer y encapsular todos los accesos a la fuente de datos. El DAO maneja la conexión con la fuente de datos para obtener y almacenar datos; de manera que se pueda acceder a los datos que maneja la aplicación sin necesidad de conocer la fuente que suministra esa información.

2.2.3 Tarjetas CRC

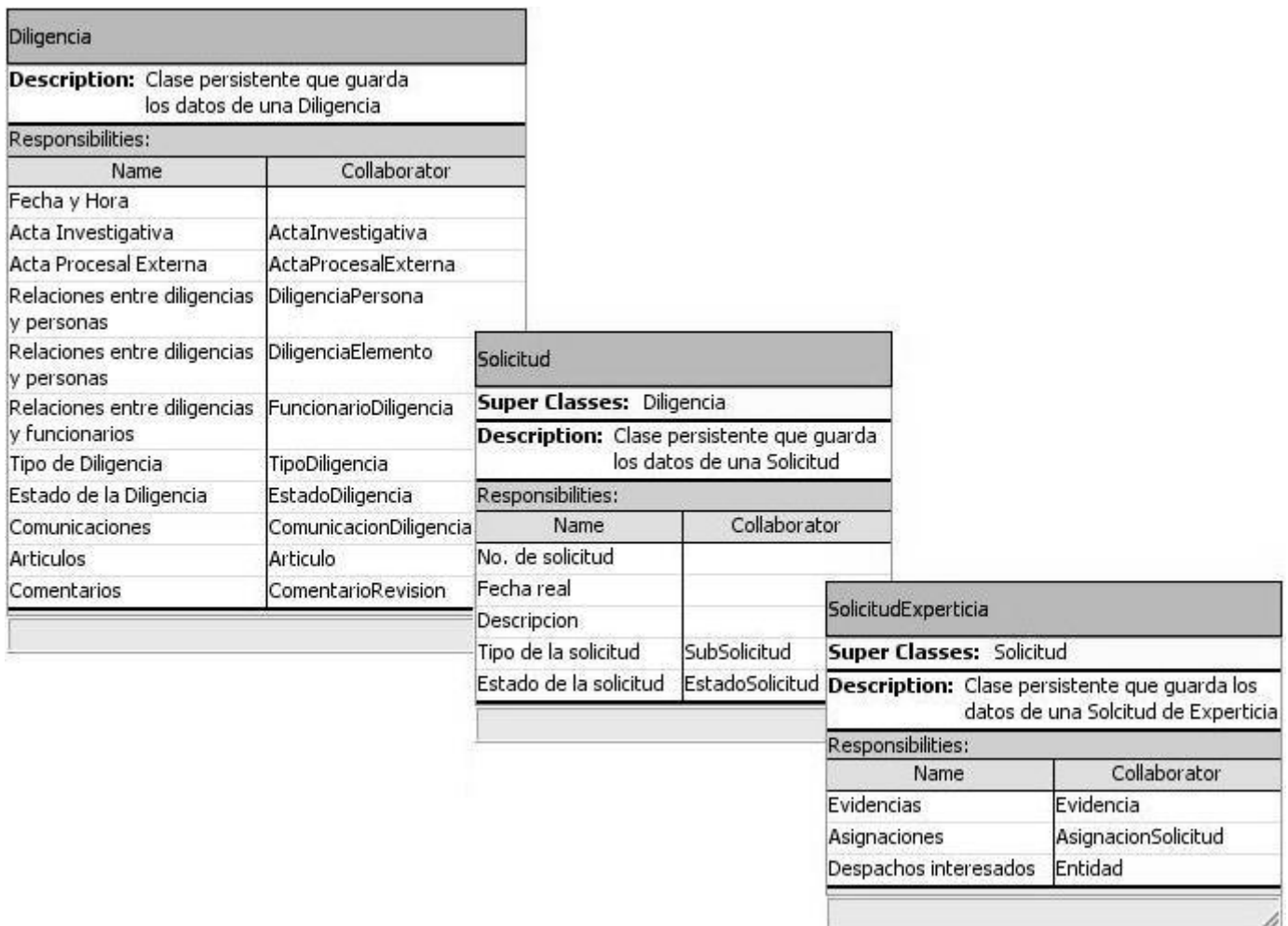


Figura 2-4: Tarjeta CRC para la jerarquía de Diligencia

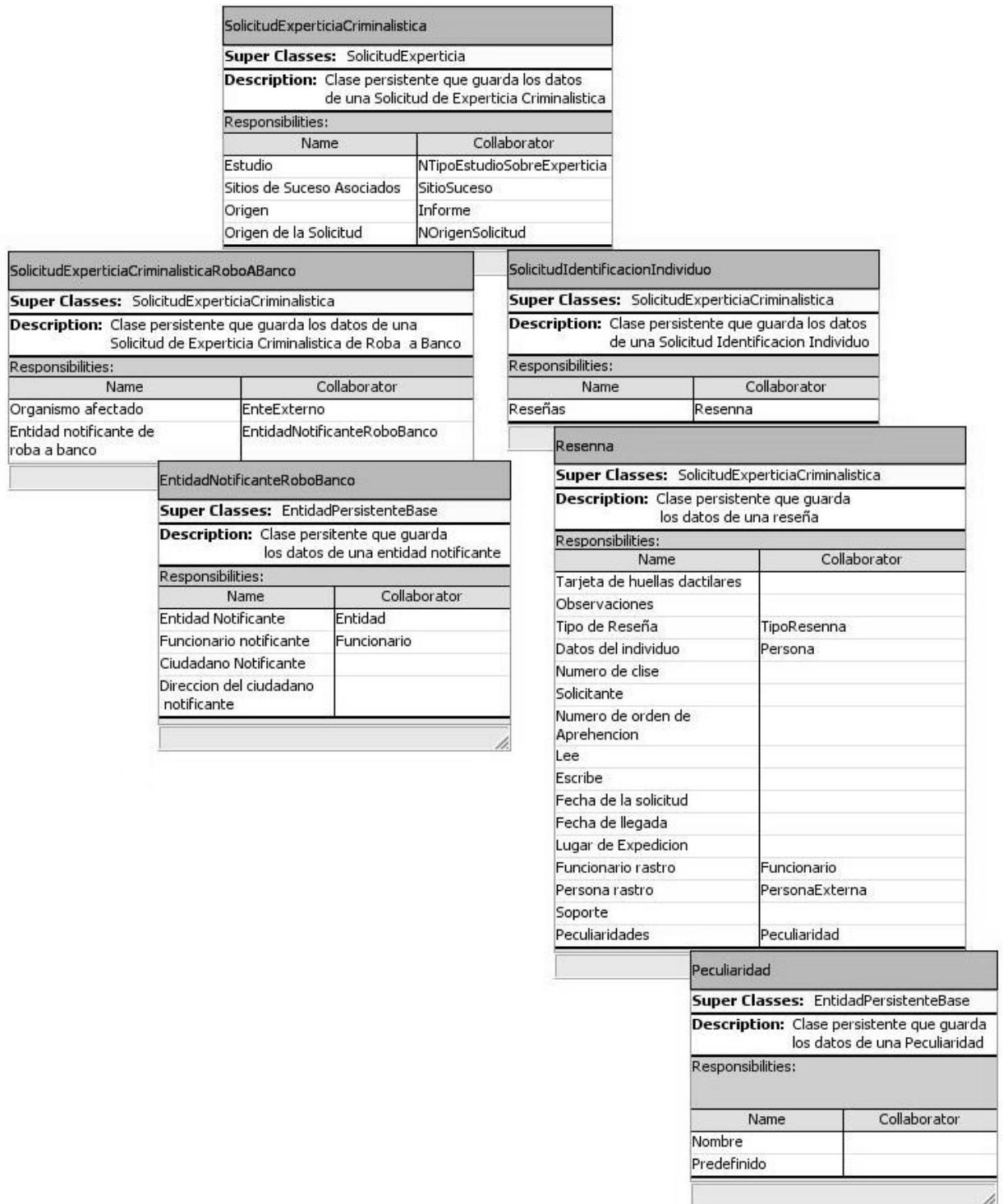


Figura 2-5: Tarjeta CRC para las solicitudes de experticia criminalística

SolicitudExperticiaReveladoRolloFotografico	
Super Classes: SolicitudServicio	
Description: Clase persistente que guarda los datos de una Solicitud Revelado Rollo Fotografico	
Responsibilities:	
Name	Collaborator
Despachos interesados	Entidad
Rollos fotograficos	RolloFotografico

RolloFotografico	
Super Classes: EntidadPersistenteBase	
Description: Clase persistente que guarda los datos de un Rollo Fotografico	
Responsibilities:	
Name	Collaborator
Numero de rollo	
Cantidad de fotos	
Observaciones	
Contener Actas Investigativas	ActaInvestigativa

Figura 2-6: Tarjeta CRC para las solicitudes de revelado de rollo fotográfico

El resto de las tarjetas CRC desarrolladas se encuentran el Anexo 2.

A continuación se describen un conjunto de clases usadas en la solución del submódulo Solicitudes que agrupan funcionalidades comunes, con el fin de eliminar la repetición innecesaria de código y lograr una mayor flexibilidad en el sistema a la hora de realizar futuras refactorizaciones, pedidos de cambio solicitados por el cliente o la incorporación de nuevos componentes desarrollados por el equipo de Arquitectura.

Nombre: SolicitudUtil	
Descripción: Clase utilitaria que agrupa funcionalidades comunes en consecuencia a la lógica de negocio de Solicitudes Criminalísticas.	
Atributo	Tipo
<i>Esta clase no presenta atributos.</i>	
Responsabilidades:	
Nombre:	asociarElementosACasoViaDiligencia(Diligencia informe, TipoRelElemento tipoRelacionElemento, NTRelAveriguacionPreliminarElemento tipoRelacionElementoInterna)
Descripción:	Funcionalidad encargada de asociar Elementos (arma, vehículo, objeto) a un Acta Investigativa especificando el tipo de relación que se establece de acuerdo al tipo acta

	asociada a la diligencia: Acta Procesal ó Acta Disciplinaria.
Nombre:	disociarElementosACasoViaDiligencia(Diligencia informe)
Descripción:	Funcionalidad encargada de disociar Elementos (arma, vehículo, objeto) del Acta Investigativa asociada a la diligencia.
Nombre:	poblarDiligencia(Diligencia diligencia, boolean emitido)
Descripción:	Funcionalidad encargada de asociar a la diligencia los datos que se necesitan para insertar la entidad en la base de datos.

Nombre: SolicitudExperticia	
Descripción: Clase genérica que agrupa elementos comunes de una solicitud criminalística y de la cual heredan las solicitudes del submódulo Solicitudes.	
Atributo	Tipo
evidencias	Set<Evidencia>
asignaciones	Set<AsignacionSolicitud>
despachosInteresados	Set<Entidad>
Responsabilidades:	
Nombre:	getEvidencias()
Descripción:	Funcionalidad encargada de obtener las evidencias asociadas a la solicitud de experticia.
Nombre:	getAsignaciones()
Descripción:	Funcionalidad encargada de obtener las asignaciones que representan la relación de asignación de una solicitud a un funcionario.
Nombre:	getDespachosInteresados()
Descripción:	Funcionalidad que se encarga de obtener los despachos interesados de la solicitud y a los cuales será enviado el informe.

Nombre: SolicitudCriminalisticaServiceImpl	
Descripción: Clase encargada de implementar la interfaz que presenta métodos de igual firma y tiene el mismo nombre sin el sufijo Impl. Constituye la súper clase inmediata de todos los servicios de solicitudes criminalísticas y utiliza la clase SolicitudUtil que engloba funcionalidades comunes de dichas solicitudes.	
Atributo	Tipo
investigacionPenalFacade	InvestigacionPenalFacade
nomencladorInvestigacionPenalFacade	NomencladorInvestigacionPenalFacade
informeUtil	InformeUtil
experticiaCriminalisticaDao	ExperticiaCriminalisticaDao
Responsabilidades:	
Nombre:	guardarEnviarSolicitudExperticia(SolicitudExperticia solicitud, boolean accion)
Descripción:	Funcionalidad encargada de realizar las acciones de incluir/actualizar una solicitud

	determinada de acuerdo a la acción especificada por parámetro invocando la clase SolicitudUtil para proveer el objeto de los datos necesarios para ejecutarlas.
--	---

Nombre: InvestigacionCriminalisticaSolicitudesFacadelImpl	
Descripción: Clase encargada de implementar la interfaz que presenta métodos de igual firma y tiene el mismo nombre sin el sufijo Impl. Constituye la implementación de la fachada del submódulo Solicitudes donde tienen lugar las llamadas a los diferentes servicios para la ejecución de las funcionalidades requeridas.	
Atributo	Tipo
gestionSolicitudService	GestionSolicitudService
solicitudExperticiaRoboABancoService	SolicitudCriminalisticaService
solicitudIdentificacionIndividuoService	SolicitudIdentificacionIndividuoService
solicitudExperticiaReveladoRolloFotograficoService	SolicitudExperticiaReveladoRolloFotograficoService
investigacionCriminalisticaSolicitudNomencladoresService	InvestigacionCriminalisticaSolicitudNomencladoresService
Responsabilidades:	
Nombre:	guardarEnviarSolicitudExperticia(SolicitudExperticiaCriminalistica solicitud, boolean accion)
Descripción:	Funcionalidad encargada de realizar las acciones de incluir/actualizar una solicitud de experticia criminalística de acuerdo a la acción especificada por parámetro.
Nombre:	obtenerSolicitudExperticia(int idSolicitud)
Descripción:	Funcionalidad encargada de obtener una solicitud de experticia criminalística dado su identificador.
Nombre:	guardarEnviarSolicitudExperticia(SolicitudExperticiaCriminalisticaRoboABanco solicitudExperticiaCriminalisticaRoboABanco, boolean accion)
Descripción:	Funcionalidad encargada de realizar las acciones de incluir/actualizar una solicitud de experticia de robo a banco de acuerdo a la acción especificada por parámetro.
Nombre:	obtenerSolicitudExperticiaRoboABanco(int idSolicitud)
Descripción:	Funcionalidad encargada de obtener una solicitud de experticia de robo a banco conociendo su identificador de base de datos.
Nombre:	guardarEnviarSolicitudIdentificacionIndividuo(SolicitudIdentificacionIndividuo solicitud, boolean emitido)
Descripción:	Funcionalidad encargada de realizar las acciones de incluir/actualizar una solicitud de identificación de individuo de acuerdo a la acción especificada por parámetro.
Nombre:	verSolicitudIdentificacionIndividuo(Integer idSolicitud)
Descripción:	Funcionalidad encargada de obtener una solicitud de identificación de individuo conociendo su identificador de base de datos.

Nombre:	buscarPersonalIdentificada(String prNombre, String sgNombre, String prApellido, String sgApellido, String ICedula, String noCedula, String noPasaporte)
Descripción:	Funcionalidad encargada de obtener un listado de personas identificadas atendiendo a criterios (letra de cedula, número de cedula, número de pasaporte).
Nombre:	guardarEnviarSolicitudExperticia(SolicitudExperticiaReveladoRolloFotografico sol, boolean accion)
Descripción:	Funcionalidad encargada de realizar las acciones de incluir/actualizar una solicitud de revelado rollo fotográfico de acuerdo a la acción especificada por parámetro.
Nombre:	obtenerPorIdSolRevelado(int id)
Descripción:	Funcionalidad encargada de obtener una solicitud de revelado de rollo fotográfico conociendo su identificador de base de datos.
Nombre:	eliminarSolicitudExpRevRolloFotografico(SolicitudExperticiaReveladoRolloFotografico solicitud, String justificacion)
Descripción:	Funcionalidad encargada de eliminar una solicitud de revelado de rollo fotográfico permitiendo incluir la justificación de su eliminación.

Nombre: CriminalisticaBaseBean	
Descripción: Clase que representa el bean manejado padre de la herencia de todos los beans manejados del submódulo Experticias Criminalísticas.	
Atributo	Tipo
acción	String
nomencladorComunFacade	NomencladorComunFacade
gestionarBaseLegalSubviewManejado	GestionarBaseLegalSubviewManejado
Responsabilidades:	
<i>Esta clase no presenta responsabilidades, solo métodos de acceso a sus atributos.</i>	

Para llevar a cabo la implementación del submódulo también se modelaron clases específicas cuyas funcionalidades responden a necesidades específicas de cada caso de uso. Seguidamente se describen las clases relevantes concebidas para dar solución a los casos de uso seleccionados.

- CU Gestionar Solicitud de Experticia Criminalística

Nombre: SolicitudExperticiaCriminalisticaDaoImpl	
Descripción: Clase encargada de implementar la interfaz que presenta métodos de igual firma y tiene el mismo nombre sin el sufijo Impl. Posee las funcionalidades necesarias para realizar las acciones de acceso a dato de la entidad SolicitudExperticiaCriminalistica.	
Atributo	Tipo

<i>Esta clase no presenta atributos.</i>	
Responsabilidades:	
Nombre:	obtenerInformeSegunSolicitud(Integer solicitud)
Descripción:	Funcionalidad encargada de obtener el informe criminalística asociado a una determinada solicitud.
Nombre:	obtenerInformeDadoSolicitudSitioSuceso(Integer sol, Integer sitio)
Descripción:	Funcionalidad encargada de obtener los informes asociados a una solicitud y un sitio de suceso delictivo determinado.
Nombre:	obtenerTiposDeExperticiaCriminalisticaSegunDespacho(int idDespacho)
Descripción:	Funcionalidad encargada de obtener los tipos de experticias criminalísticas realizadas por una determinada entidad conociendo su identificador de base de datos.

Nombre: GestionSolicitudServiceImpl	
Descripción: Clase encargada de implementar la interfaz que presenta métodos de igual firma y tiene el mismo nombre sin el sufijo Impl. Posee las funcionalidades necesarias para realizar la lógica de negocio correspondiente a la entidad SolicitudExperticiaCriminalistica.	
Atributo	Tipo
dolicitudExperticiaCriminalisticaDao	SolicitudExperticiaCriminalisticaDao
analisisInformacionFacade	AnalisisInformacionFacade
nomencladorComunFacade	NomencladorComunFacade
gestionAdministrativaFacade	GestionAdministrativaFacade
nomencladoresGestionAdministrativaFacade	NomencladoresGestionAdministrativaFacade
Responsabilidades:	
Nombre:	informeAsociadoSitioSuceso(Integer sitio, Integer solicitud)
Descripción:	Funcionalidad encargada de obtener los informes asociados a una solicitud y sitio de suceso delictivo determinado.
Nombre:	obtenerInformeSegunSolicitud(Integer solicitud)
Descripción:	Funcionalidad encargada de obtener el informe criminalística asociado a una determinada solicitud.
Nombre:	obtenerPorID(int id)
Descripción:	Funcionalidad encargada de obtener una solicitud de experticia criminalística atendiendo al criterio de su identificador de base de datos.
Nombre:	obtenerTiposDeExperticiaCriminalisticaSegunDespacho(int idDespacho)
Descripción:	Funcionalidad encargada de obtener los tipos de experticias criminalísticas realizadas por una determinada entidad conociendo el identificador de base de datos de la misma.

Nombre: SolicitudExperticiaCriminalisticaIncluirManejado	
Descripción: Clase que representa el bean de respaldo a la página encargada de mostrar una interfaz que permita al usuario incluir una nueva Solicitud de Experticia Criminalística.	
Atributo	Tipo
solicitud	SolicitudExperticiaCriminalistica
informeOrigen	Informe
actaProcesalOrigen	ActaInvestigativa
comunicacionOrigen	ComunicacionDiligencia
actaProcesalComunicacion	ActaInvestigativa
actaProcesalExternaComunicacion	ActaProcesalExterna
expedienteTanatologico	ExpedienteTanatologico
dependenciaAResponder	DependenciaInterna
enteExternoAResponder	EnteExterno
sitioSucesoNuevo	SitioSuceso
entidadesAResponder	List<Entidad>
listaArticulos	List<Articulo>
evidenciasAsociadas	List<Evidencia>
personasAsociadas	List<Persona>
sitiosSucesoAsociados	List<SitioSuceso>
listaDependencias	List<SelectItem>
listaEntesExternos	List<SelectItem>
listaEstudios	List<SelectItem>
listaImportancias	List<SelectItem>
listaSubtiposSolicitud	List<SelectItem>
listaTodasDependencias	List<SelectItem>
listaTodosEntesExternos	List<SelectItem>
listaEstados	List<SelectItem>
listaMunicipios	List<SelectItem>
listaParroquias	List<SelectItem>
listaExpedientesTanat	List<SelectItem>
asociarEvidenciasSubViewManejado	AsociarEvidenciasSubViewManejado
completarExperticiaAvaluoEspecialSubviewManejado	CompletarExperticiaAvaluoEspecialSubviewManejado
solicitudExperticiaCriminalisticaVistaPreviaManejado	SolicitudExperticiaCriminalisticaVistaPreviaManejado
administracionFacade	AdministracionFacade
AnalisisInformacionFacade	AnalisisInformacionFacade
investigacionPenalFacade	InvestigacionPenalFacade
nomencladorComunFacade	NomencladorComunFacade
nomencladorCriminalisticaFacade	NomencladorCriminalisticaFacade
nomencladoresGestionAdministrativaFacade	NomencladoresGestionAdministrativaFacade
nomencladorInvestigacionPenalFacade	NomencladorInvestigacionPenalFacade
registroControlFacade	RegistroControlFacade
investigacionForenseFacade	InvestigacionForenseFacade
Responsabilidades:	

Nombre:	guardarEnviar()
Descripción:	Funcionalidad encargada de incluir una solicitud en el sistema, la solicitud incluida toma estado <i>Emitida</i> .
Nombre:	vistaPrevia()
Descripción:	Funcionalidad encargada de navegar hacia la UI Vista Previa.
Nombre:	limpiarCampos(ActionEvent e)
Descripción:	Funcionalidad encargada de retornar al estado inicial los campos editables de la página.
Nombre:	cancelar(ActionEvent e)
Descripción:	Funcionalidad encargada de regresar a la página que invocó el escenario actual del caso de uso.
Nombre:	reset()
Descripción:	Funcionalidad auxiliar encargada de eliminar todas a las referencias utilizadas en la realización del informe, dejando listas las instancias para la creación de uno nuevo.
Nombre:	personalizarComunicacion(ActionEvent e)
Descripción:	Funcionalidad encargada de navegar hacia el caso de uso extendido Configurar Encabezado
Nombre:	isInicializarIncluidos()
Descripción:	Funcionalidad encargada de inicializar las subvistas incluidas en el escenario actual.
Nombre:	mostrarEvidenciaAsociada(ActionEvent e)
Descripción:	Funcionalidad encargada de navegar hacia la interfaz que muestra los datos de las evidencias asociadas a la solicitud.
Nombre:	validarDatos()
Descripción:	Funcionalidad encargada de validar los datos incluidos en la página.
Nombre:	adicionarDependenciaAResponder()
Descripción:	Funcionalidad encargada de validar de incluir en la solicitud el listado de entidades a responder.
Nombre:	asociarEvidencia(ActionEvent e)
Descripción:	Funcionalidad encargada de asociar evidencias a la para su estudio.
Nombre:	asociarPersona()
Descripción:	Funcionalidad encargada de asociar personas a la solicitud.
Nombre:	asociarSitioSuceso()
Descripción:	Funcionalidad encargada de asociar sitios de sucesos a la solicitud.

Nombre:	inicializarListaEstudiosYDependencias(ActionEvent e)
Descripción:	Funcionalidad encargada de listar los tipos estudios criminalísticas realizados por cada Dependencia, dicha funcionalidad utiliza la fachada

Nombre: SolicitudExperticiaCriminalisticaVerManejado	
Descripción: Clase que representa el bean de respaldo a la página encargada de mostrar una interfaz que permita al usuario ver los datos de una Solicitud de Experticia Criminalística.	
Atributo	Tipo
solicitud	SolicitudExperticiaCriminalistica
evidenciasAsociadas	List<Evidencia>
sitiosSucesoAsociados	List<SitioSuceso>
personasAsociadas	List<Persona>
entidadesAResponder	List<Entidad>
articulosAsociados	List<Articulo>
completarExperticiaAvaluoEspecialVerSubviewwManejado	CompletarExperticiaAvaluoEspecialVerSubviewManejado
solicitudExperticiaCriminalisticaModificarManejado	SolicitudExperticiaCriminalisticaModificarManejado
solicitudCriminalisticaEliminarManejado	SolicitudCriminalisticaEliminarManejado
verDetallesEvidenciaManejado	VerDetallesEvidenciaManejado
notificarCambiosEntidad	NotificarCambiosEntidad
analisisInformacionFacade	AnalisisInformacionFacade
Responsabilidades:	
Nombre:	cerrar(ActionEvent e)
Descripción:	Funcionalidad encargada de regresar a la página que invocó el escenario actual del caso de uso.
Nombre:	eliminar()
Descripción:	Funcionalidad encargada de navegar hacia la UI Eliminar Solicitud.
Nombre:	getParametrosReporte()
Descripción:	Funcionalidad encargada de obtener los recursos necesarios para mostrar el reporte correspondiente a la solicitud.
Nombre:	getEntidadesAResponder()
Descripción:	Funcionalidad encargada de obtener las Entidades a responder asociadas a la solicitud.
Nombre:	getEvidenciasAsociadas()
Descripción:	Funcionalidad encargada de obtener las evidencias asociadas a la solicitud.
Nombre:	modificar()
Descripción:	Funcionalidad encargada de navegar hacia la UI Modificar Solicitud.

Nombre:	getPersonasAsociadas()
Descripción:	Funcionalidad encargada de obtener las personas asociadas a la solicitud.
Nombre:	getSitiosSucesoAsociados()
Descripción:	Funcionalidad encargada de obtener los sitios de sucesos asociados a la solicitud.

Nombre: SolicitudCriminalisticaEliminarManejado	
Descripción: Clase que representa el bean de respaldo a la página encargada de mostrar una interfaz que permita al usuario eliminar una Solicitud de Experticia Criminalística.	
Atributo	Tipo
solicitud	Solicitud
justificacion	String
solicitudComunFacade	SolicitudFacade
notificarCambiosEntidad	NotificarCambiosEntidad
Responsabilidades:	
Nombre:	eliminar(ActionEvent e)
Descripción:	Funcionalidad encargada de ejecutar el método que elimina la solicitud.
Nombre:	cancelar(ActionEvent e)
Descripción:	Funcionalidad encargada de regresar a la página que invocó el escenario actual del caso de uso.
Nombre:	aceptar(ActionEvent e)
Descripción:	Funcionalidad encargada de regresar a la página que invocó el escenario actual del caso de uso después de haber sido eliminada la solicitud.

- CU Tratar Evidencias para Solicitud

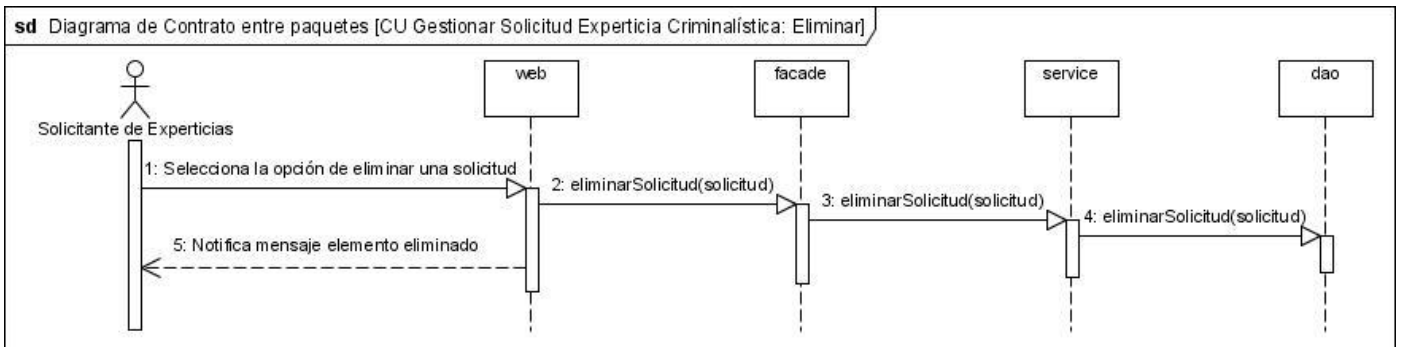
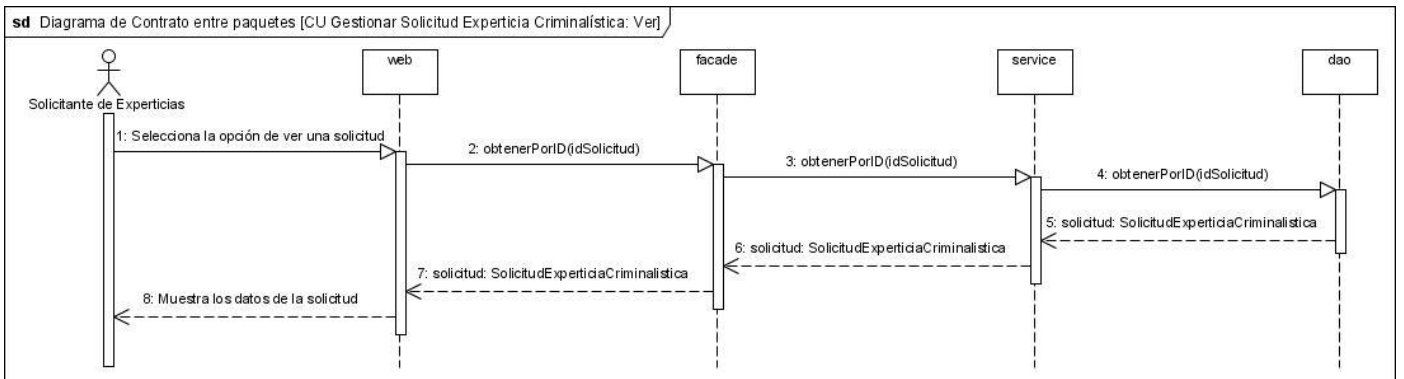
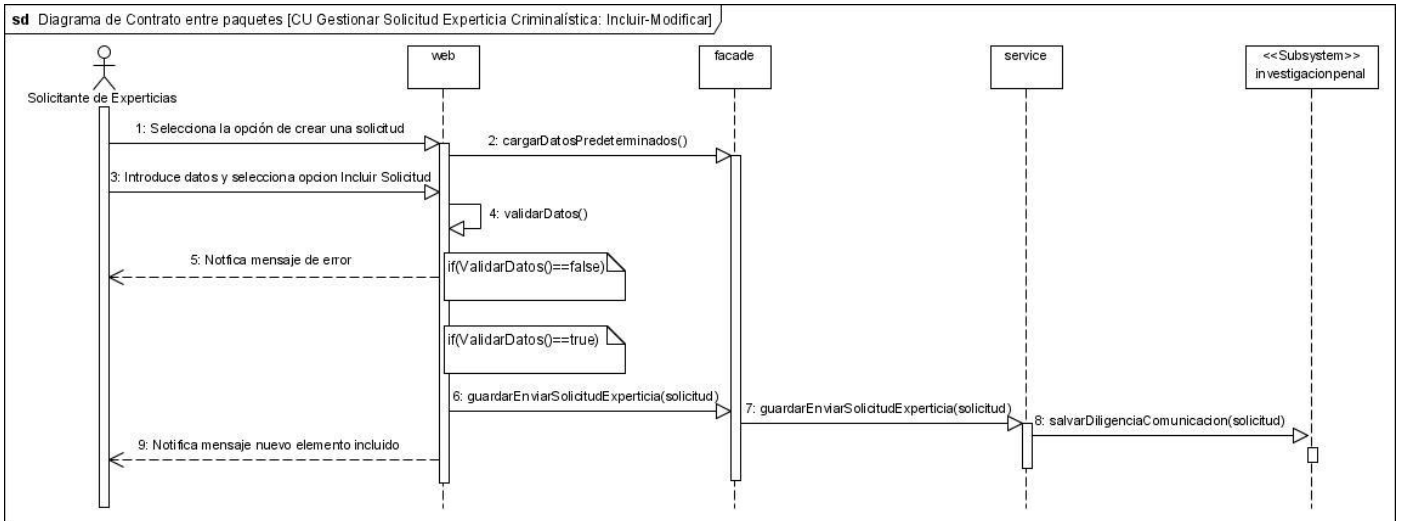
Nombre: AsociarEvidenciasSubViewManejado	
Descripción: Clase que representa el bean de respaldo a la página encargada de mostrar una interfaz que permita al usuario asociar evidencias a una solicitud.	
Atributo	Tipo
solicitud	SolicitudExperticiaCriminalistica
interfaz	AsociarEvidenciasManejadoInterface
ingresarEvidenciaNoActaProcesalManejado	IngresarEvidenciaNoActaProcesalManejado
Responsabilidades:	
Nombre:	disociarEvidencia(ActionEvent e)

Descripción:	Funcionalidad encargada de disociar una evidencia de la solicitud.
Nombre:	getEvidenciasAsociadas()
Descripción:	Funcionalidad encargada de obtener las evidencias asociadas a la solicitud.
Nombre:	getArmasAsociadas()
Descripción:	Funcionalidad encargada de obtener las evidencias de tipo <i>Arma</i> asociadas a la solicitud.
Nombre:	getObjetosAsociados()
Descripción:	Funcionalidad encargada de obtener las evidencias de tipo <i>Objeto</i> asociadas a la solicitud.
Nombre:	getVehiculosAsociados()
Descripción:	Funcionalidad encargada de obtener las evidencias de tipo <i>Vehículo</i> asociadas a la solicitud.
Nombre:	incluirArma()
Descripción:	Funcionalidad encargada de asociar una evidencia de tipo <i>Arma</i> a la solicitud.
Nombre:	incluirObjeto()
Descripción:	Funcionalidad encargada de asociar una evidencia de tipo <i>Objeto</i> a la solicitud.
Nombre:	incluirVehiculo()
Descripción:	Funcionalidad encargada de asociar una evidencia de tipo <i>Vehículo</i> a la solicitud.
Nombre:	solicitarExperticias(ActionEvent e)
Descripción:	Funcionalidad encargada de navegar hacia la UI Incluir Solicitud Experticia.
Nombre:	verEvidencia(ActionEvent e)
Descripción:	Funcionalidad encargada de navegar hacia la UI Ver Evidencia.

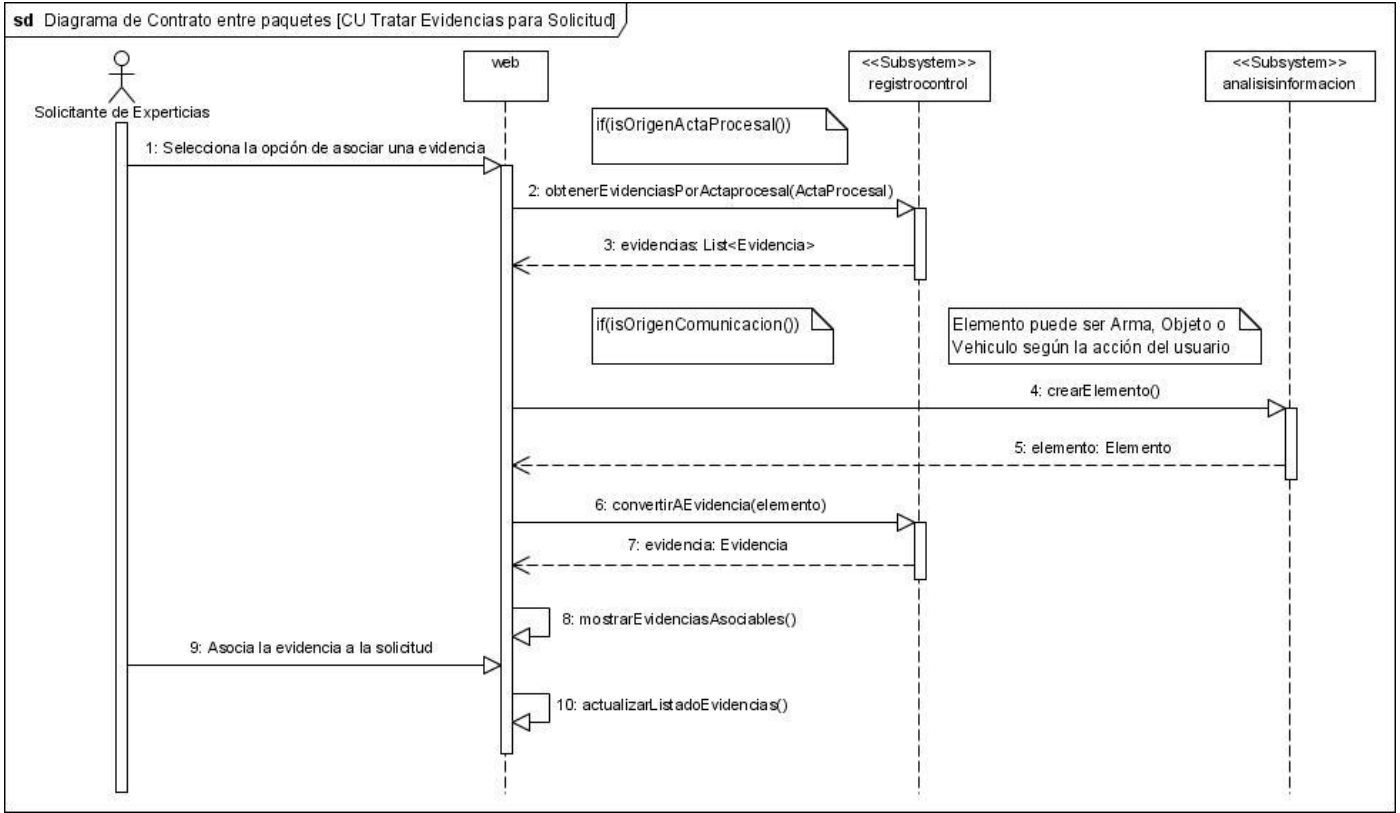
2.2.4 Realización de los Casos de Uso

Los diagramas de contrato entre paquetes constituyen un artefacto generado, a partir de necesidades propias del proyecto, como documentación al cliente que facilita un mayor entendimiento de los procesos una vez que se haga la transferencia tecnológica. Los diagramas de contrato entre paquetes representan la interacción entre subsistemas, constituyen diagramas de secuencia que ilustran las interacciones entre paquetes, abstrayéndose de los detalles internos de cada capa.

- CU Gestionar Solicitud de Experticia Criminalística



- CU Tratar Evidencias para Solicitud



2.3 Modelo de Datos

El modelo de datos describe la representación lógica y física de la información persistente manejada por el sistema. Estudia los datos independientemente del procesamiento que los transforma. Se compone de tres piezas de información interrelacionadas: el objeto de datos, los atributos que describen el objeto de datos y la relación que conecta objetos de datos entre sí. (18)

2.3.1 Diagrama de clases persistentes

El siguiente diagrama de clases persistentes representa la interacción entre las clases de dominio pertenecientes al submódulo Solicitudes y su vinculación con clases de otros módulos con las que se establecen relaciones de herencia y asociación significativas.

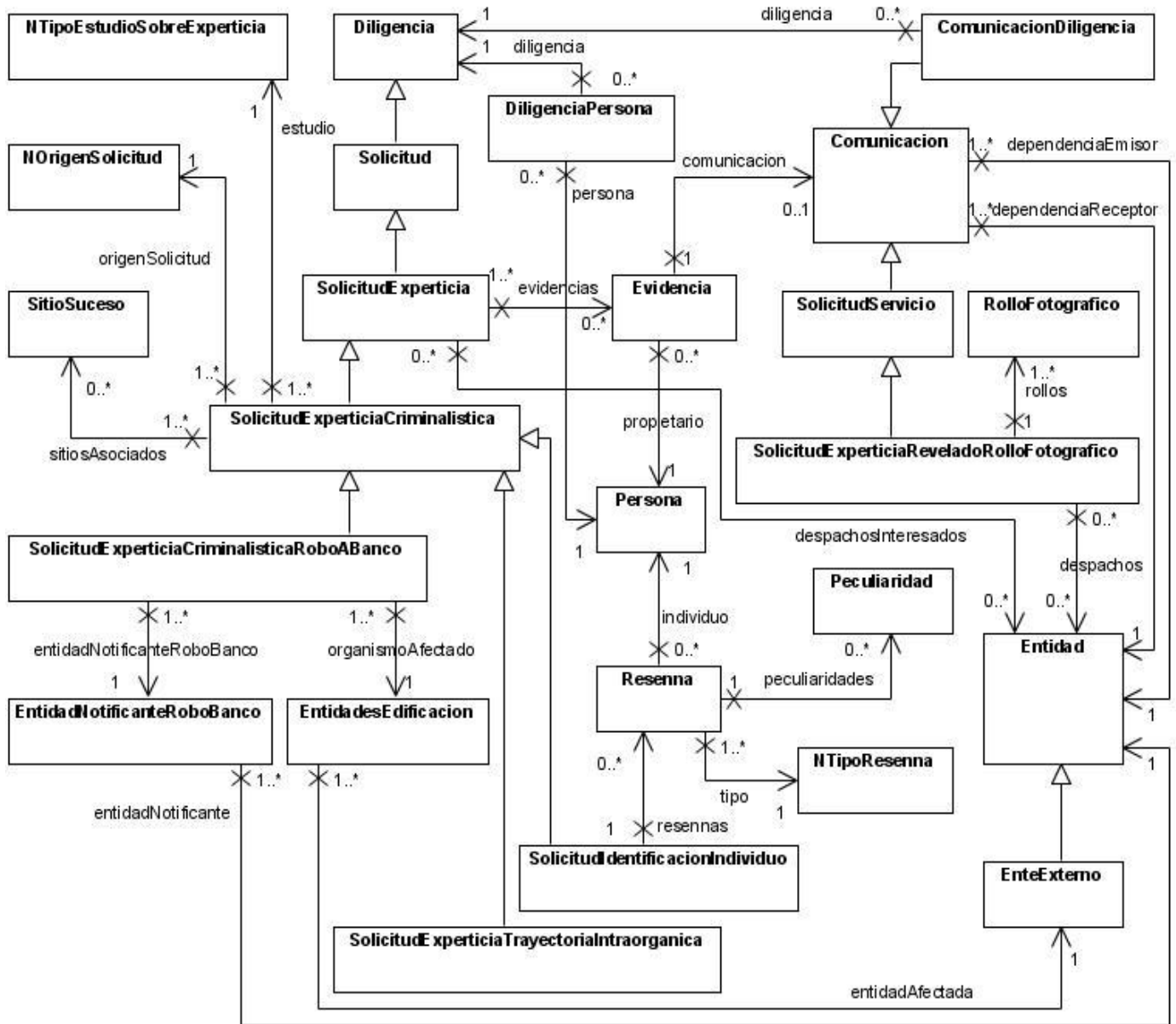


Figura 2-7: Diagrama de clases persistentes

2.3.2 Diagrama de tablas del modelo relacional

Estructura física de las clases de dominio del submódulo en la base de datos Oracle.

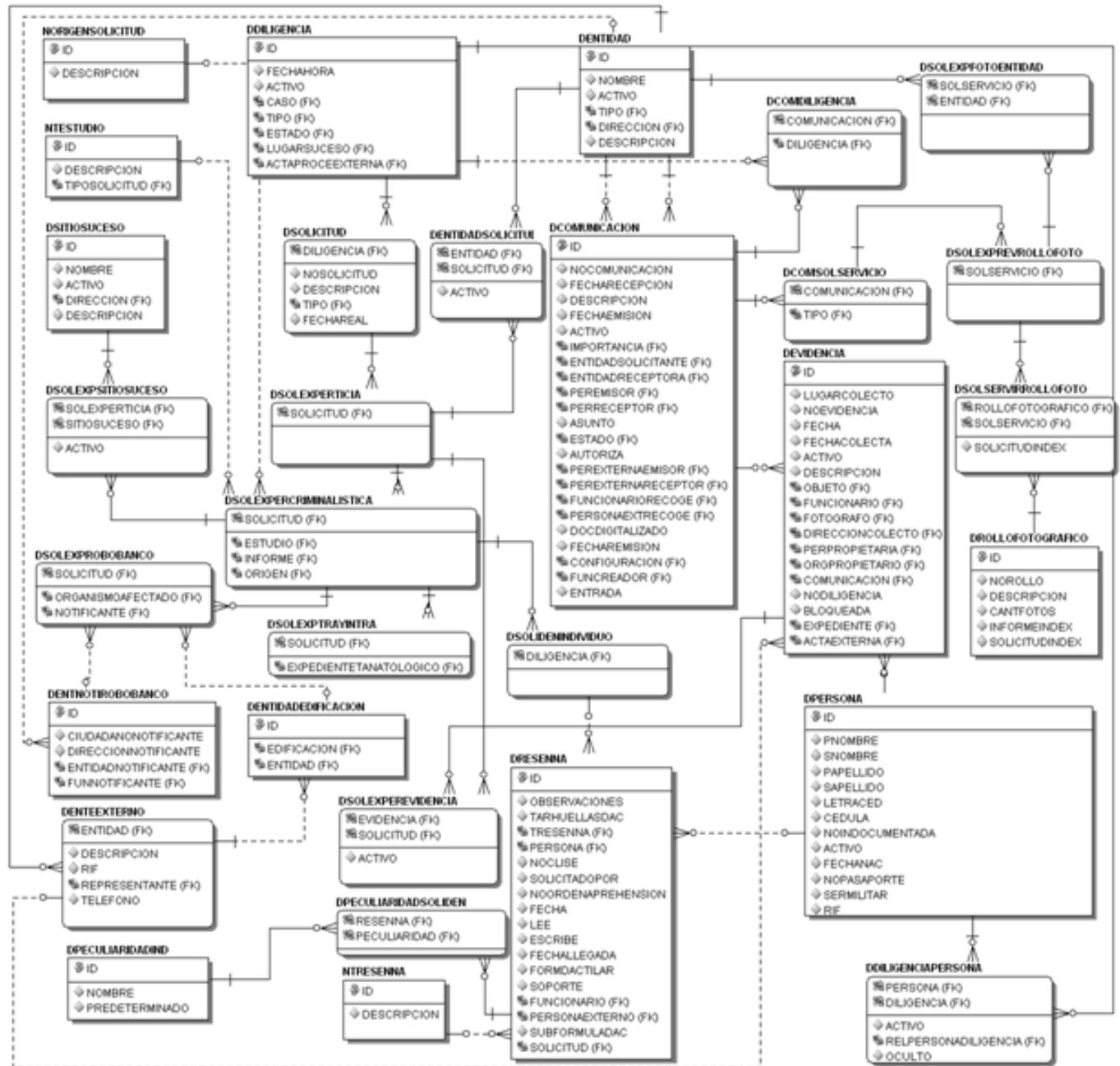


Figura 2-8: Diagrama de tablas del modelo relacional

2.4 Modelo de Implementación

El modelo de implementación es comprendido por un conjunto de componentes y subsistemas que constituyen la composición física de la implementación del sistema. Entre los componentes podemos encontrar archivos, ejecutables, ficheros de código fuente. Fundamentalmente, se describe la relación que existe desde los paquetes y clases del modelo de diseño a subsistemas y componentes físicos. (19)

2.4.1 Diagrama de subsistemas de implementación

Los subsistemas de implementación están muy relacionados con los subsistemas de diseño en el modelo de diseño, razón por la cual los subsistemas de implementación deberían seguir la traza uno a uno de sus subsistemas de diseño correspondientes. Proporcionan una forma de organizar los artefactos del modelo de implementación en fragmentos más manejables. Un subsistema puede estar formado por componentes, interfaces y otros subsistemas. Además un subsistema puede implementar las interfaces que representan la funcionalidad que exportan en forma de operaciones. (19)

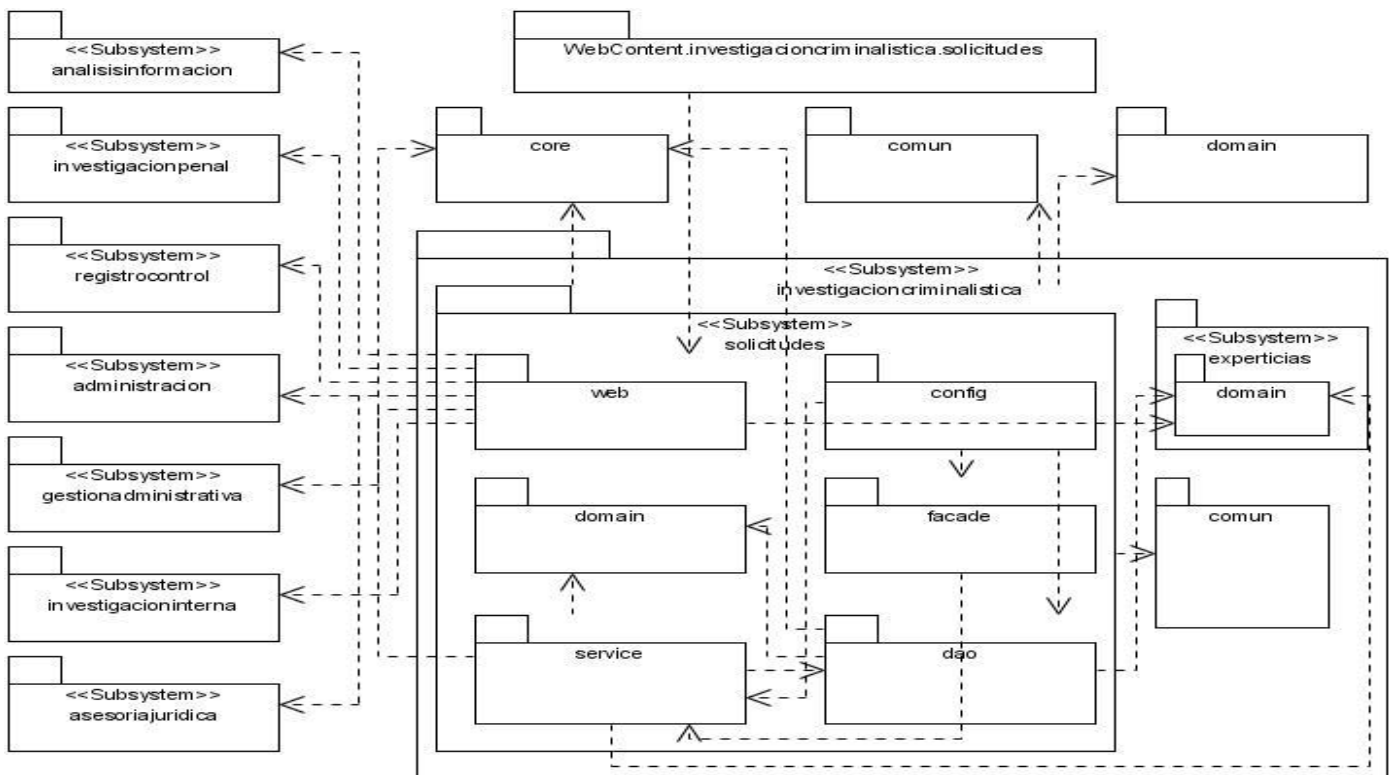


Figura 2-9: Diagrama de subsistemas de implementación

2.4.2 Diagrama de componentes

Los diagramas de componentes son usados para estructurar el modelo de implementación en términos de subsistemas de implementación y mostrar las relaciones entre los elementos de implementación. También se utilizan para mostrar las dependencias de compilación de los ficheros de código, relaciones de derivación entre ficheros de código fuente y ficheros que son resultados de la compilación, dependencias entre elementos de implementación y los correspondientes elementos de diseños que son implementados.

Debido a su extensión el diagrama de componentes para el submódulo fue dividido en varias partes para su mejor representación. (19)

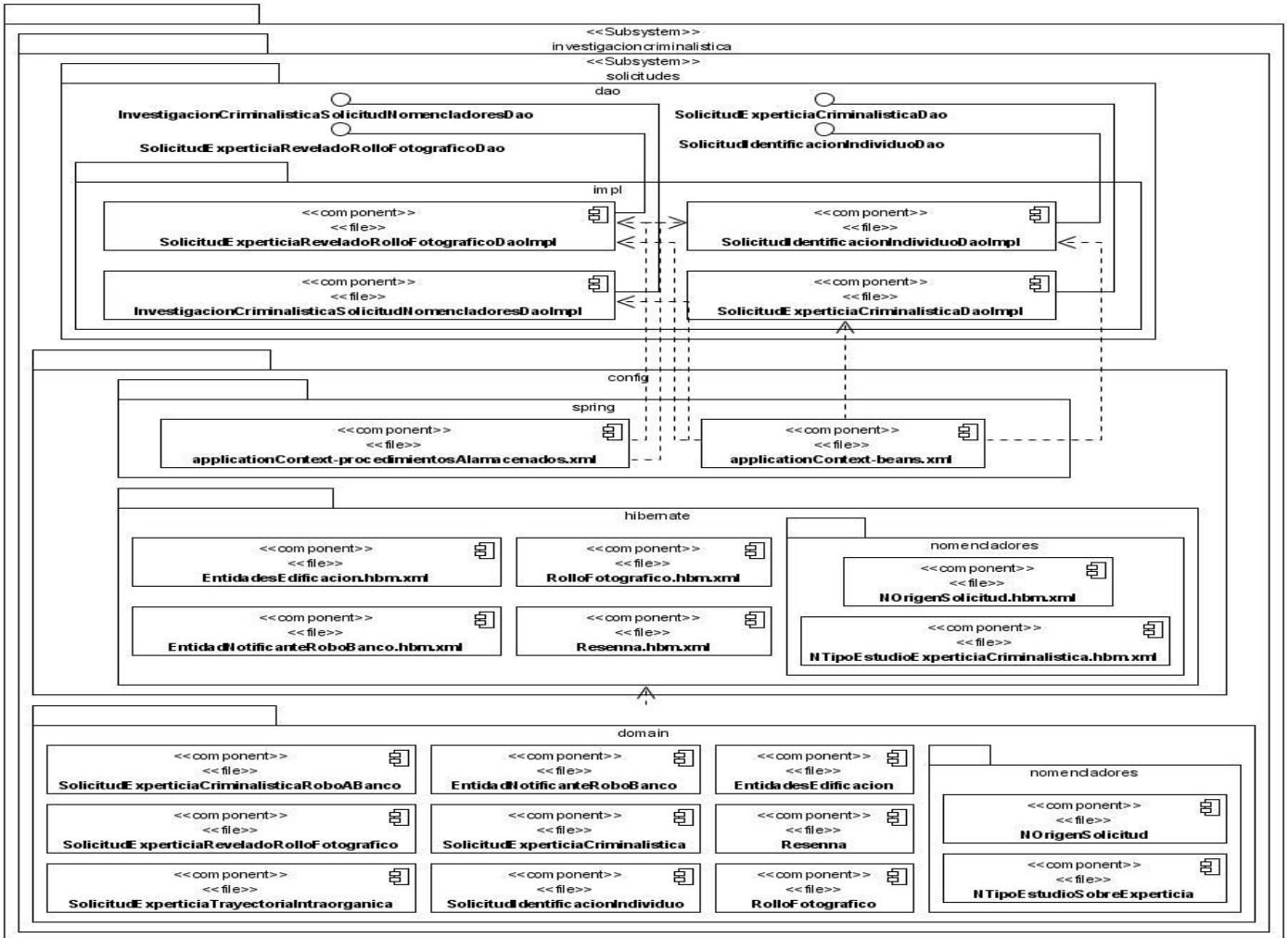


Figura 2-10: Diagrama de Componentes. Vista Acceso a Datos

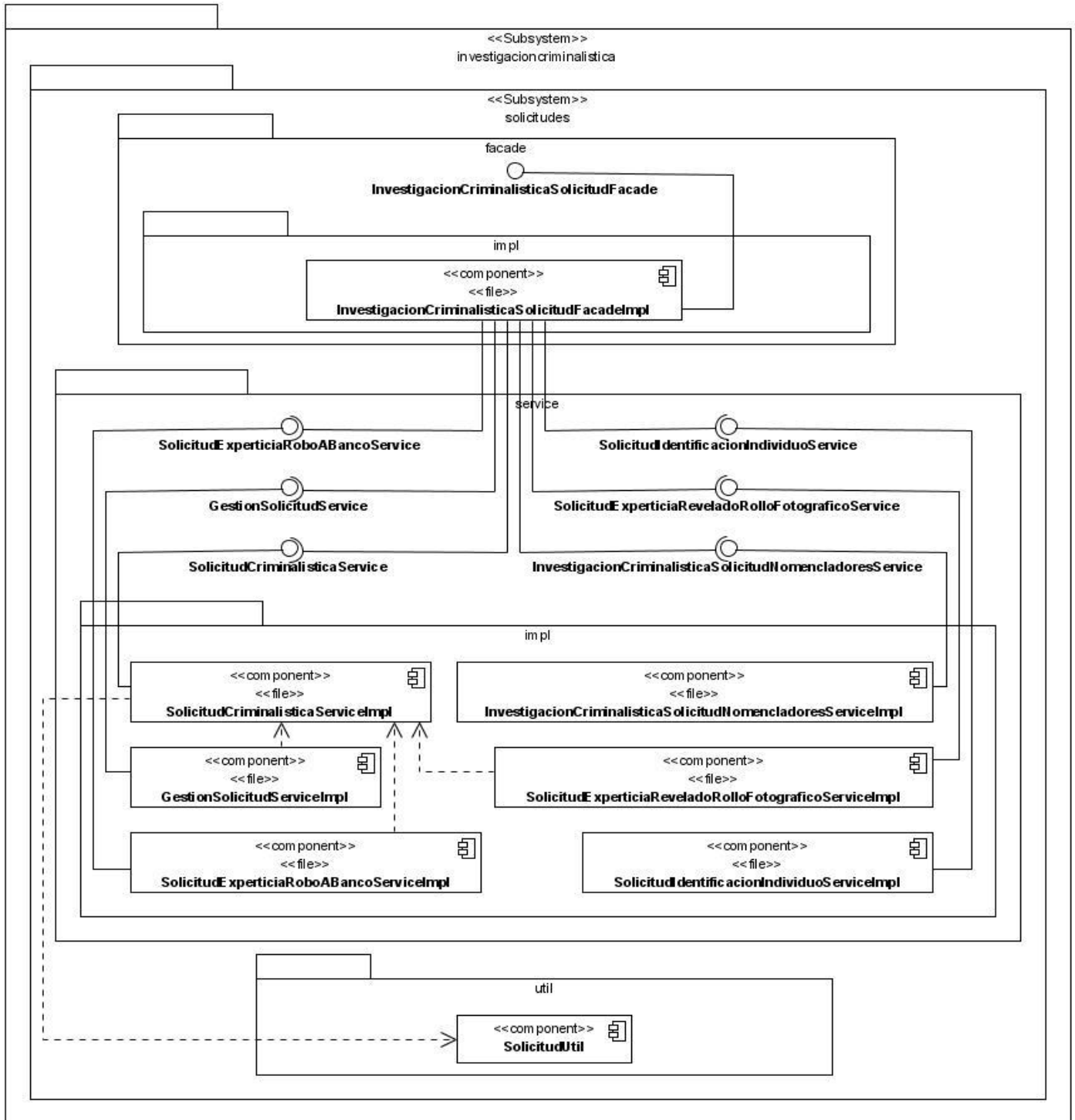


Figura 2-11: Diagrama de Componentes. Vista Lógica de Negocio

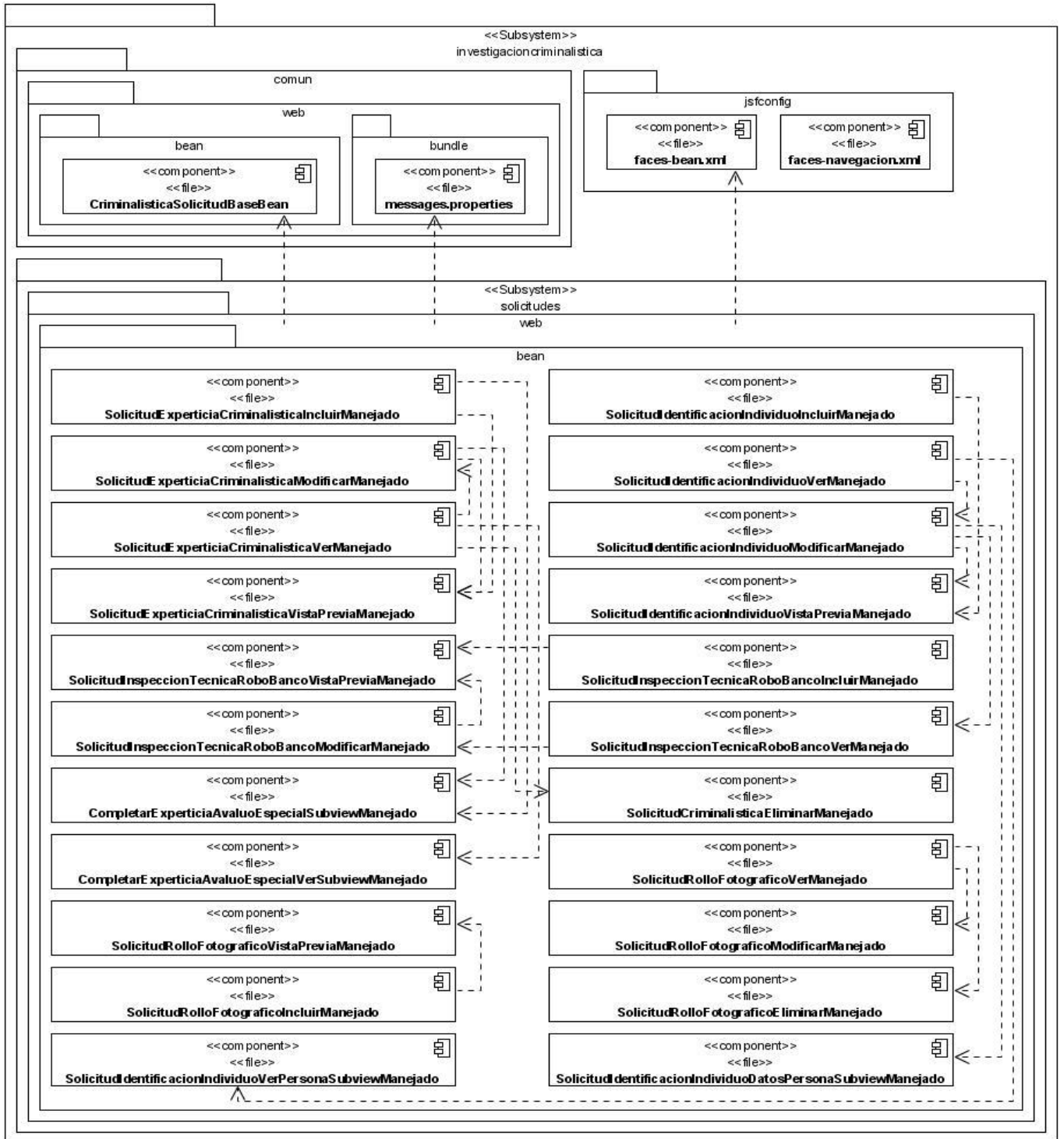


Figura 2-12: Diagrama de Componentes. Vista Presentación

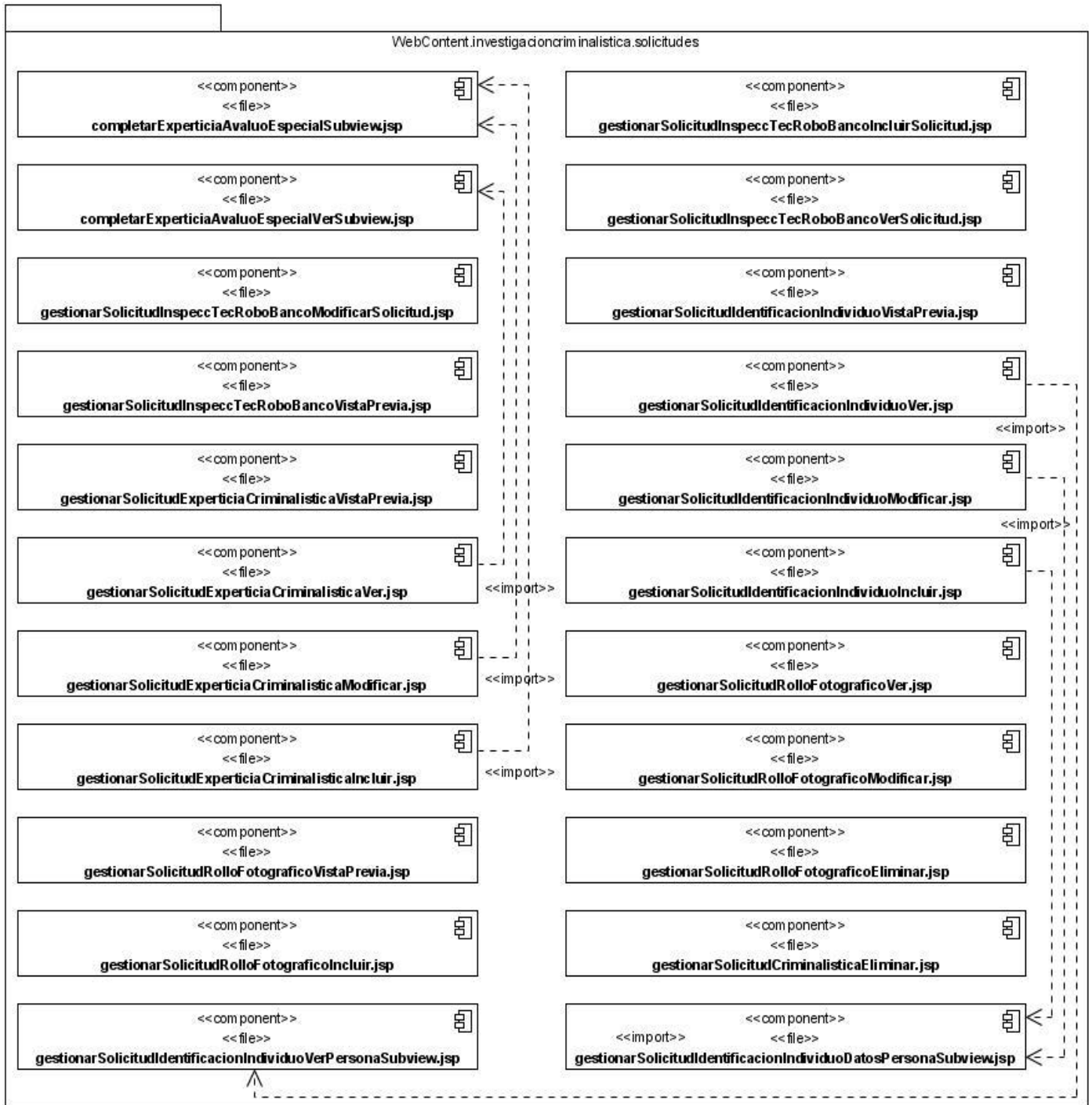


Figura 2-13: Diagrama de Componentes. Vista de las páginas

2.4.3 Estándar de codificación

Un estándar de codificación es un término que describe convenciones para escribir código fuente en lenguajes de programación y es frecuentemente dependiente del lenguaje que se haya elegido para codificar. Para el desarrollo del sistema, el equipo de Arquitectura elaboró el documento Guía de Estilo de Código para el proyecto CICPC que resume las características fundamentales del estilo de código puesto en práctica, siguiendo las Convenciones de Código Java (Java Code Conventions).

2.4.4 Interfaces de usuario

La interfaz de usuario (IU) es la porción de un software que permite comunicar al usuario con este, es la parte que ve y con la cual interactúa. La interfaz de usuario es uno de los aspectos más importantes de cualquier aplicación ya que garantiza que quienes interactúen con el sistema obtengan el máximo rendimiento del programa; deben estar elaboradas para que brinden tanto comodidad, como eficiencia.

Las interfaces de usuario del submódulo Solicitudes así como las de todo el SIIPOL fueron diseñadas por un equipo especializado según el Manual de Pautas de Diseño definido para el sistema. A continuación se muestran algunas de las interfaces de los casos de uso más significativos. El resto de las interfaces de usuario del submódulo se pueden consultar en el Anexo 3.



Figura 2-14: CU Tratar Sitios de Suceso para Solicitud

SIIPOL SISTEMA DE INVESTIGACIÓN E INFORMACIÓN POLICIAL

Bienvenido(a): AAILYADIEL RAMOS RODRIGUEZ. 19/04/2009 ... 16:04 Inicio Ayuda Salir

AGENDA DE TRABAJO

- Notificaciones
- Asignaciones
- Recepciones
- Aprobaciones
- Revisiones
- Remisiones
- Borradores
- Archivos

MENÚ PRINCIPAL

- Acceso
- Gestión de Despacho
- Atención Telefónica
- Investigación Penal
- Aprehensión
- Archivos Históricos

INCLUIR SOLICITUD DE EXPERTICIA CRIMINALÍSTICA

Solicitud Elementos

Evidencias Personas Sitios de Suceso

Evidencias Relacionadas

Armas Objetos Vehículos

No. Evidencia	Nombre Evidencia	Descripción	Disociar
EK-08-0001-00006-4	Casquillo	Evidencia recién encontrada en pésimo estado	Disociar
EK-08-0001-00006-3	Guante	Este es el guante referente a al caso K-08-0001-00006	Disociar
Total: 2			

[Incluir Objeto](#)

Figura 2-15: CU Tratar Evidencias para Solicitud

SIIPOL SISTEMA DE INVESTIGACIÓN E INFORMACIÓN POLICIAL

Bienvenido(a): AAILYADIEL RAMOS RODRIGUEZ. 19/04/2009 ... 16:08 Inicio Ayuda Salir

AGENDA DE TRABAJO

- Notificaciones
- Asignaciones
- Recepciones
- Aprobaciones
- Revisiones
- Remisiones
- Borradores
- Archivos

MENÚ PRINCIPAL

- Acceso
- Gestión de Despacho
- Atención Telefónica

INCLUIR SOLICITUD DE EXPERTICIA CRIMINALÍSTICA

Solicitud Elementos

Evidencias **Personas** Sitios de Suceso

Personas Asociadas

Identificación	Nombres y Apellidos	Disociar
V-12502985	ETI Luis Alejandro Moreno Osuna	Disociar
E-83073110	AS Miguel Angel Monagas Reyes	Disociar
Total: 2		

[Incluir Persona](#)

Figura 2-16: CU Tratar Personas para Solicitud

SIIPOL

BIENVENIDO(A): AAILYADIEL RAMOS RODRIGUEZ

19/04/2009 ... 15:02

Inicio Ayuda Salir

AGENDA DE TRABAJO

- Notificaciones
- Asignaciones
- Recepciones
- Aprobaciones
- Revisiones
- Remisiones
- Borradores
- Archivos

MENÚ PRINCIPAL

- Acceso
- Gestión de Despacho
- Atención Telefónica
- Investigación Penal
- Aprehensión
- Archivos Históricos
- Investigación Criminalística
- Asesoría Jurídica
- Investigación en Ciencias ...
- Análisis de Información
- Estadísticas
- Dotación de Equipos Polic...
- Evidencia
- Auditoría
- Administración

INCLUIR SOLICITUD DE EXPERTICIA CRIMINALÍSTICA

Solicitud
Elementos

Datos de la Solicitud

No. Expediente	K-08-0001-00006
Dependencia Emisora	Dirección General

Datos de la Solicitud

Tipo de Solicitud	Estudio Criminalístico	Importancia
Solicitud de Experticia de Microscopía Electrónica	Estudio de Análisis de Trazas de Disparo	Normal

Descripción

El material suministrado para ser sometido al análisis de trazas de disparo consiste en muestras tomadas por adherencias, en las regiones dorsales y palmares de ambas manos del ciudadano: Juan Hernández García de cédula V-11034091, colectadas por el funcionario José Y. Urbina de credencial 2345231.

Enviar a Entidad

Dependencia Interna Dependencia: División de Física Comparativa

Entidad Externa

Dirigido a	C HECTOR JULIO JIMENEZ MENDOZA
Cargo	Jefe de la Div. Física Comparativa
Credencial	0015753

Responder a Entidades

Dependencia Interna Dependencia: Apoyo Administrativo

Entidad Externa

Dirigido a	CJ Benito Artiagas
Cargo	Administrativo
Credencial	0013133

Adicionar

Entidad	Dirigido a	Credencial / Identificación	Eliminar
DISIP	AS Lesky Alfonso De Leon	0000003	Eliminar
Total: 1			

Base Legal

Ley: LCICPC Artículo: Adicionar

De Conformidad con lo establecido en la Ley	Artículo	Eliminar
COPPC	Artículo 1	Eliminar
LCICPC	Artículo 4	Eliminar
Total: 2		

Personalizar Comunicación
Vista Previa
Incluir
Limpiar
Cancelar

Figura 2-17: IU Incluir Solicitud de Experticia Criminalística

SIIPOL SISTEMA DE INVESTIGACIÓN E INFORMACIÓN POLICIAL

Bienvenido(a): AAIL YADIEL RAMOS RODRIGUEZ 19/04/2009 ... 15:45 Inicio Ayuda Salir

AGENDA DE TRABAJO

- Notificaciones
- Asignaciones
- Recepciones
- Aprobaciones
- Revisiones
- Remisiones
- Borradores
- Archivos

MENÚ PRINCIPAL

- Acceso
- Gestión de Despacho
- Atención Telefónica
- Investigación Penal
- Aprehensión
- Archivos Históricos
- Investigación Criminalística
- Asesoría Jurídica
- Investigación en Ciencias ...
- Análisis de Información
- Estadísticas
- Dotación de Equipos Polic...
- Evidencia
- Auditoría
- Administración

INCLUIR SOLICITUD DE EXPERTICIA CRIMINALÍSTICA

Solicitud **Elementos**

Evidencias Correspondientes al Acta Procesal

No. Evidencia	Elemento	Descripción	Asociar
EK-08-0001-00006-2	Vehículo	El vehículo se encuentra en perfecto estado	Asociar
Total: 1			

Evidencias Asociadas

No. Evidencia	Elemento	Descripción	Disociar
EK-08-0001-00006-1	Objeto	Prenda de gran valor encontrada ...	Disociar
Total: 1			

Personas Correspondientes al Acta Procesal

Identificación	Nombres y Apellidos	Asociar
V-85121201	AAIL YANDER CACERES CRUZ	Asociar
Total: 1		

Personas Asociadas

Identificación	Nombres y Apellidos	Disociar
T-348	Andres A Perez Perez	Disociar
Total: 1		

Sitios de Suceso Correspondientes al Acta Procesal

Nombre	Dirección	Asociar
SitioA	Barinas, Cruz Paredes, Barrancas, 112, d32, 32121	Asociar
Total: 1		

Sitios de Suceso Asociados

Nombre	Dirección	Disociar
Café de Pedro	Falcón, Democracia, Piedra Grande	Disociar
Total: 1		

[Personalizar Comunicación](#)

Figura 2-18: IU Incluir Solicitud de Experticia Criminalística - Elementos

SIIPOL SISTEMA DE INVESTIGACIÓN E INFORMACIÓN POLICIAL

Bienvenido(a): AAIL YADIEL RAMOS RODRIGUEZ 19/04/2009 15:27 Inicio Ayuda Salir

AGENDA DE TRABAJO

- Notificaciones
- Asignaciones
- Recepciones
- Aprobaciones
- Revisiones
- Remisiones
- Borradores
- Archivos

MENÚ PRINCIPAL

- Acceso
- Gestión de Despacho
- Atención Telefónica
- Investigación Penal
- Aprehensión
- Archivos Históricos
- Investigación Criminalística
- Asesoría Jurídica
- Investigación en Ciencias ...
- Análisis de Información
- Estadísticas
- Dotación de Equipos Polic...
- Evidencia
- Auditoría

VISTA PREVIA DE SOLICITUD DE EXPERTICIA CRIMINALÍSTICA

Datos de la Solicitud

Responsable	AAIL YADIEL RAMOS RODRIGUEZ	Tipo de Solicitud	Solicitud de Experticia de Microscopía Electrónica
Credencial	0000011	Estudio Criminalístico	Estudio de Análisis de Trazas de Disparo
Dependencia Responsable	Dirección General	Importancia	Normal
No. Expediente	K-08-0001-00006		

Enviar a

Entidad	División de Física Comparativa
Funcionario	C HECTOR JULIO JIMENEZ MENDOZA

Responder a

Entidad	Enviar a	Credencial / Identificación
DISIP	AS Lesky Alfonso De Leon	0000003
Total: 1		

Descripción

El material suministrado para ser sometido al análisis de trazas de disparo consiste en muestras tomadas por adherencias, en las regiones dorsales y palmares de ambas manos del ciudadano: Juan Hernández García de cédula V-11034091, colectadas por el funcionario José Y. Urbina de credencial 2345231.

Base Legal

De Conformidad con lo establecido en la Ley	Artículo
LCICPC	Artículo 4
COPPC	Artículo 1
Total: 2	

Imprimir/Exportar Cerrar

Figura 2-19: IU Vista Previa Solicitud de Experticia Criminalística

2.5 Conclusiones

En el presente capítulo se llevó a cabo el desarrollo de la solución cumpliendo con los requisitos funcionales y no funcionales asociados al submódulo, a través el Modelado e Implementación de la propuesta, según establece la metodología seleccionada y basado siempre en la arquitectura definida para el sistema.

El correcto análisis de la documentación entregada por el equipo de analistas unido al apropiado empleo de patrones de diseño así como el uso de las tarjetas CRC permitió realizar un diseño adecuado para el submódulo Solicitudes, agilizando el proceso de desarrollo mediante la generación de documentación útil de un verdadero valor para el equipo de trabajo. También se elaboraron el modelo de datos y el modelo de implementación quedando una aplicación lista para el flujo de trabajo de prueba.

Capítulo 3 Validación de la propuesta de solución

Este capítulo resumirá el conjunto de pruebas realizadas al submódulo Solicitudes para comprobar la satisfacción de los requisitos funcionales y no funcionales asociados al mismo. Se abordarán los tipos de pruebas realizadas, resultados obtenidos en las mismas y la evaluación de esos resultados.

3.1 Métodos de prueba. Niveles de prueba.

Las pruebas de software son los procesos que permiten verificar y revelar la calidad de un producto. El objetivo de las pruebas no es asegurar la ausencia de defectos en un software, únicamente puede demostrar que existen defectos en el software. Una prueba tiene éxito si descubre un error no detectado hasta entonces. (20)

3.1.1 Métodos de prueba

Pruebas de Caja Negra: Pruebas que se llevan a cabo sobre la interfaz del software. El objetivo es demostrar que las funciones del software son operativas, que las entradas se aceptan de forma adecuada y se produce un resultado correcto, y que la integridad de la información externa se mantiene, sin tener en cuenta su funcionamiento interno. Se encargan de saber qué es lo que hace el software, pero sin dar importancia a cómo lo hace.

Pruebas de Caja Blanca: Se denomina cajas blancas a un tipo de prueba de software que se realiza sobre las funciones internas de un módulo. Así como las pruebas de caja negra ejercitan los requisitos funcionales desde el exterior del módulo, las de caja blanca están dirigidas a las funciones internas. Las pruebas de caja blanca se llevan a cabo en primer lugar, sobre un módulo concreto, para luego realizar las de caja negra sobre varios subsistemas (integración).

3.1.2 Niveles de prueba

Pruebas Unitarias: Comienzan con la prueba de cada módulo. Una prueba unitaria es una forma de probar el correcto funcionamiento de un módulo de código. Esto sirve para asegurar que cada uno de los estos funcione correctamente por separado. El objetivo de las pruebas unitarias es aislar cada parte del programa y mostrar que las partes individuales son correctas. Proporcionan un contrato escrito que el fragmento de código debe satisfacer. Estas pruebas aisladas proporcionan cinco ventajas básicas,

fomentan el cambio, simplifica la integración, documenta el código, separa la interfaz del código y hace que los errores estén más acotados y sean fáciles de localizar.

Pruebas de Integración: A partir del esquema del diseño, los módulos probados se vuelven a probar combinados para probar sus interfaces. Pruebas integrales o pruebas de integración son aquellas que se realizan en el ámbito del desarrollo de software una vez que se han aprobado las pruebas unitarias. Únicamente se refieren a la prueba o pruebas de todos los elementos unitarios que componen un proceso, hecha en conjunto, de una sola vez. Consiste en realizar pruebas para verificar que un gran conjunto de partes de software funcionan juntos.

Prueba del Sistema: El software ensamblado totalmente con cualquier componente hardware que requiera, se prueba para comprobar que se cumplen los requisitos funcionales. Cualquier pieza de software completo, desarrollado o adquirido, puede verse como un sistema que debe probarse, ya sea para decidir acerca de su aceptación, para analizar defectos globales o para estudiar aspectos específicos de su comportamiento, tales como seguridad o rendimiento. A este tipo de pruebas donde se estudia el producto completo se les llama Pruebas de Sistema.

Pruebas de Aceptación: Estas pruebas las realiza el cliente. Son básicamente pruebas funcionales, sobre el sistema completo, y buscan una cobertura de la especificación de requisitos y del manual del usuario. Estas pruebas no se realizan durante el desarrollo, pues sería impresentable al cliente; sino que se realizan sobre el producto terminado e integrado.

3.2 Análisis de las pruebas realizadas

Para evaluar la solución desarrollada se planificaron varias iteraciones donde se estuvo probando el software exhaustivamente. Se abarcaron los métodos y niveles de prueba expuestos anteriormente en cada una de las iteraciones, arrojando estas últimos resultados visibles.

Durante el proceso de desarrollo del módulo se llevaron a cabo pruebas unitarias para ir comprobando el correcto funcionamiento del avance de la solución. Estas pruebas fueron dirigidas por los mismos desarrolladores y se usaron para ensayar diferentes funcionalidades cruciales. La automatización de las pruebas de unidad es fundamental para agilizar el proceso, por tal motivo fue utilizado el framework JUnit. Estas pruebas no se planificaron ni se registraron sus resultados, se fueron haciendo a medida que la

solución se desarrollaba; como constancia se muestra a continuación un resumen de la clase creada para probar todos los métodos de la fachada del submódulo.

Nombre: InvestigacionCriminalisticaSolicitudFacadeJUnit	
Descripción: Clase encargada de realizar pruebas unitarias a las funcionalidades de la clase InvestigacionCriminalisticaSolicitudFacade.	
Atributo	Tipo
investigacionCriminalisticaSolicitudFacade	InvestigacionCriminalisticaSolicitudFacade
app	ApplicationContext
Responsabilidades:	
Nombre:	setUp()
Descripción:	Funcionalidad encargada de cargar los contextos.
Nombre:	testGuardarEnviarSolicitudExperticia()
Descripción:	Método encargado de realizar una prueba unitaria a las acciones de incluir/actualizar una solicitud de experticia.
Nombre:	testObtenerSolicitudExperticia()
Descripción:	Método encargado de realizar una prueba unitaria a la funcionalidad que permite obtener una solicitud de experticia criminalística.
Nombre:	testGuardarEnviarSolicitudExperticiaRoboBanco()
Descripción:	Método encargado de realizar una prueba unitaria a las acciones de incluir/actualizar una solicitud de experticia de robo a banco.
Nombre:	testObtenerSolicitudExperticiaRoboABanco()
Descripción:	Método encargado de realizar una prueba unitaria a la funcionalidad que permite obtener una solicitud de experticia de robo a banco.
Nombre:	testRegistrarSolicitudIdentificacionIndividuo()
Descripción:	Método encargado de realizar una prueba unitaria a las acciones de incluir/actualizar una solicitud de identificación de individuo.
Nombre:	testVerSolicitudIdentificacionIndividuo()
Descripción:	Método encargado de realizar una prueba unitaria a la funcionalidad que permite

	obtener una solicitud de identificación de individuo.
Nombre:	testBuscarPersonalIdentificada()
Descripción:	Método encargado de realizar una prueba unitaria a la funcionalidad que permite obtener un listado de personas para ser identificadas.
Nombre:	testGuardarEnviarSolicitudExperticiaRolloFotog()
Descripción:	Método encargado de realizar una prueba unitaria a las acciones de incluir/actualizar una solicitud de revelado rollo fotográfico.
Nombre:	testObtenerPorIdSolRevelado()
Descripción:	Método encargado de realizar una prueba unitaria a la funcionalidad encargada de obtener una solicitud de revelado de rollo fotográfico.
Nombre:	testEliminarSolicitudReveladoRolloFotografico()
Descripción:	Método encargado de realizar una prueba unitaria a la funcionalidad encargada de eliminar una solicitud de revelado de rollo fotográfico.

Una vez concluida la primera etapa de implementación del sistema y obtenido un producto, se procede a realizar distintos tipos de pruebas que van desde el equipo de calidad del propio proyecto hasta el cliente final. Como material base para el inicio de la validación de la solución propuesta, se estructuraron los Casos de Prueba a partir de los cuales se chequeó el cumplimiento de los requisitos funcionales del software. Un ejemplo de los casos de prueba elaborados se encuentra en el Anexo 4.

Durante las pruebas se estuvieron documentando todas las No Conformidades surgidas dándole la categoría de Alta, Media o Baja según su impacto en el negocio y el sistema. A continuación se muestra el comportamiento del submódulo Solicitudes a lo largo de cada ciclo de pruebas, comparando siempre los resultados con el submódulo Experticias dentro del total de no conformidades encontradas en el módulo Investigación Criminalística. Las pruebas realizadas se organizaron de la siguiente manera:

Pruebas Internas: Pruebas realizadas al sistema por el equipo de calidad interna del proyecto; se realizaron con el fin de entregar un producto lo más libre de errores posible al tercero encargado de validar la factibilidad del sistema. Se centraron en el cumplimiento de las funcionalidades descritas en el listado de requisitos y de casos de uso.

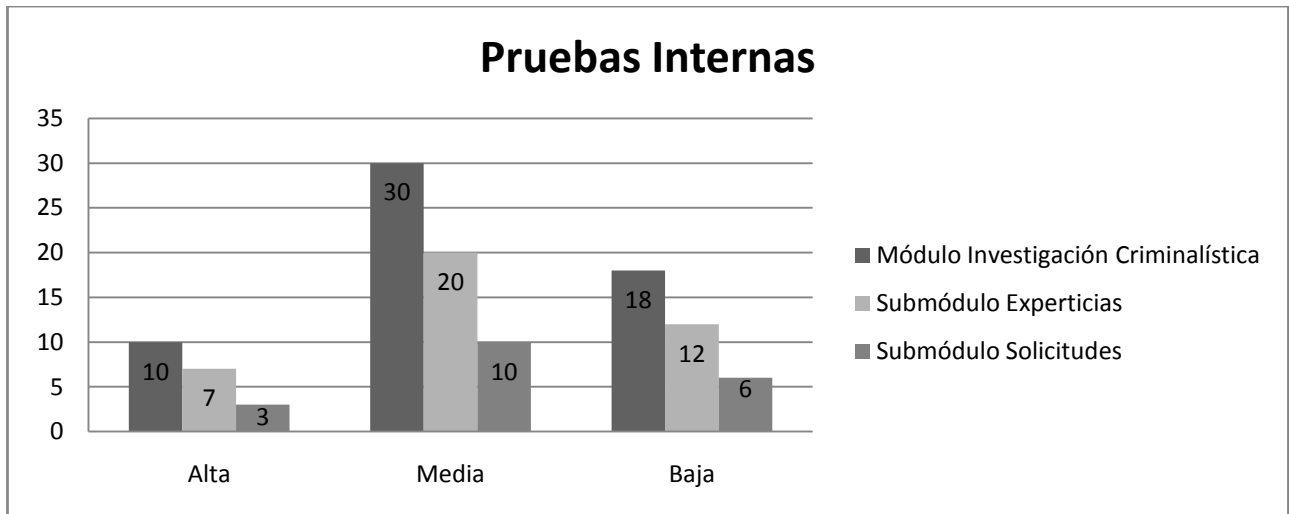


Figura 3-1: No Conformidades detectadas durante las Pruebas Internas

Las pruebas internas marcaron el inicio del plan de validación de la porción de software desarrollado; concluyeron con un total de cincuenta y ocho defectos encontrados, asociados a la totalidad del módulo, donde solo un 32,8 por ciento pertenecen al submódulo Solicitudes.

Pruebas Cruzadas: Pruebas realizadas al sistema por el resto de los equipos de desarrollo del proyecto. Se realizaron con el fin de encontrar la mayor cantidad de errores posibles en término de validaciones, pautas definidas por arquitectura de información, formato de los campos, entre otras.

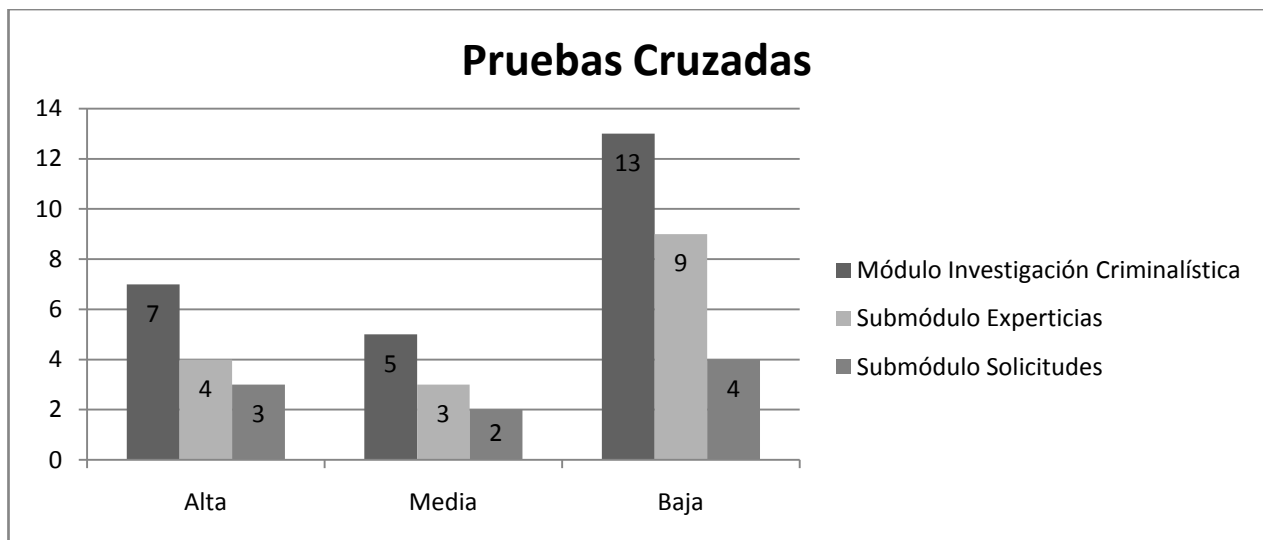


Figura 3-2: No Conformidades detectadas durante las Pruebas Cruzadas

Las pruebas cruzadas mostraron un comportamiento similar a las pruebas internas, pues las no conformidades encontradas en el submódulo solo representan un 36 por ciento del total de 25 problemas detectados en todo el módulo. También es apreciable la disminución del número de no conformidades encontradas entre una iteración y otra; de 58 a 25 para Investigación Criminalística y de 19 a 9 para el submódulo en cuestión.

Pruebas de Liberación: Pruebas realizadas por un tercero, en este caso Calisoft, institución encargada de validar que el software cuente con la calidad requerida para ser entregado a los clientes finales.

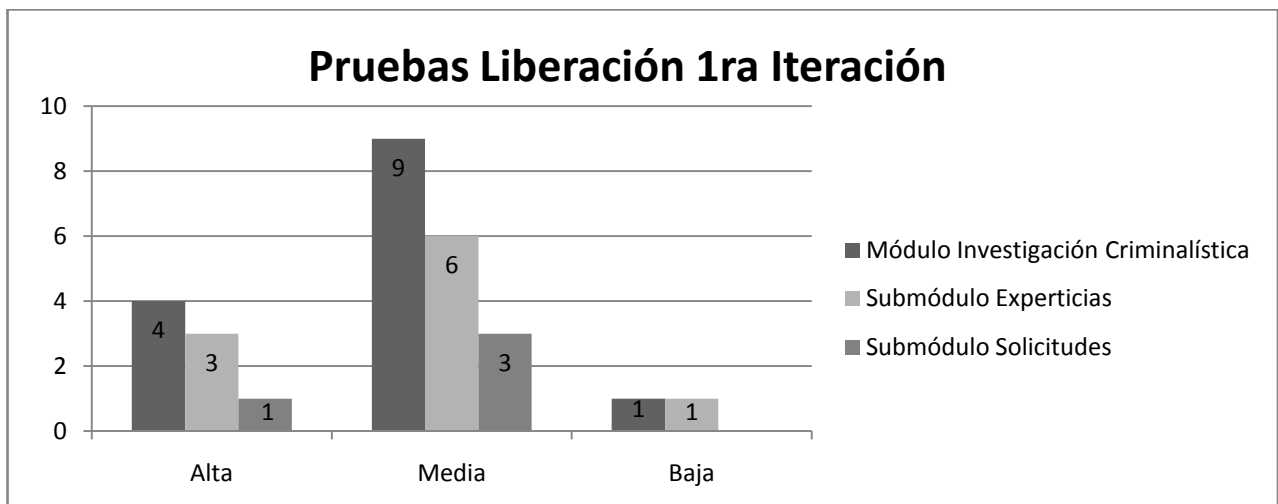


Figura 3-3: No Conformidades detectadas durante la 1ra iteración de las Pruebas de Liberación

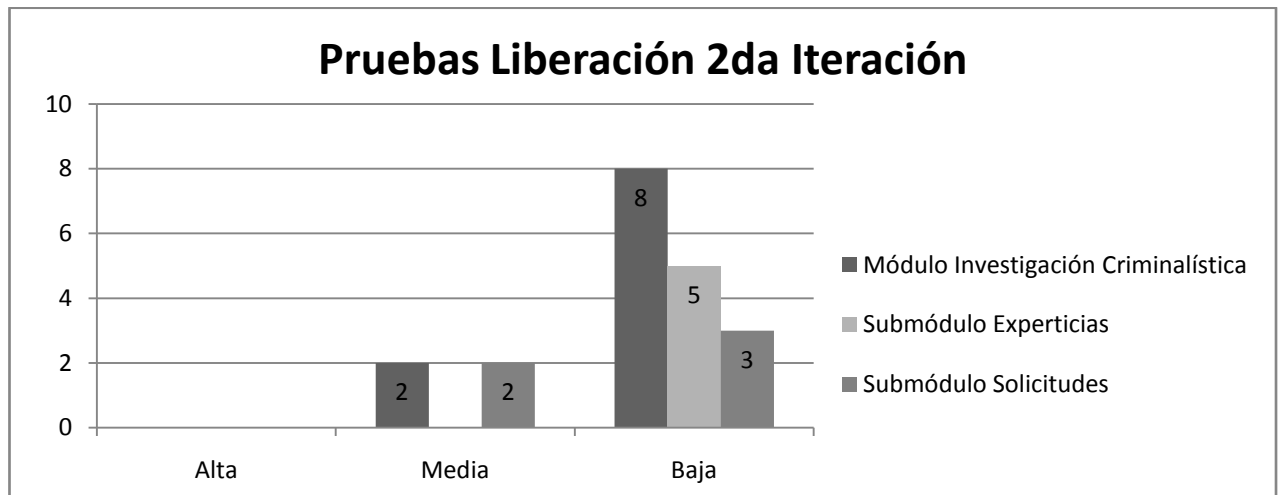


Figura 3-4: No Conformidades detectadas durante la 2da iteración de las Pruebas de Liberación

Las iteraciones desarrolladas por Calisoft se empeñaron en entregar un software libre de errores para desarrollar las pruebas de aceptación. Durante cada iteración fueron menos las no conformidades encontradas hasta culminar en una iteración final donde no se detectaron no conformidades.

Pruebas de Aceptación: Pruebas realizadas por los clientes para comprobar que el sistema hace lo que ellos esperan y necesitan.

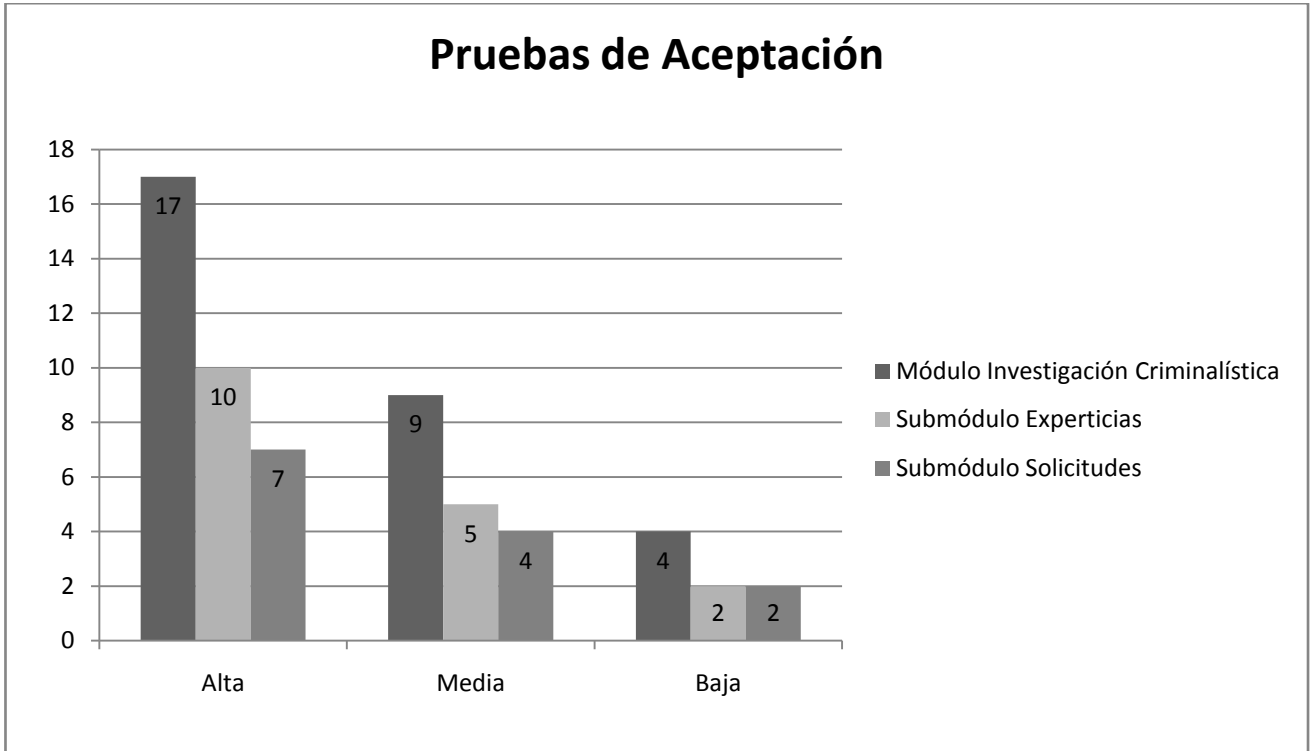


Figura 3-5: No Conformidades detectadas durante las Pruebas de Aceptación

Luego de entregar un software libre de errores las pruebas de aceptación mostraron un aumento de las no conformidades, dicho fenómeno ocurrió al presentar el software al cliente final. El cliente percibió que las necesidades que había planteado en un momento determinado habían cambiado, o sea, ya hacía varios meses desde el levantamiento de requisitos, tiempo durante el cual los procesos de la empresa habían sufrido modificaciones. Además, el enfoque que se le había dado a las funcionalidades solicitadas no era el correcto y lo que estaba implementado no era realmente lo que se necesitaba. Otra situación que se presentó fue que muchas de las funcionalidades a implementar que arrojó la captura de requisitos no eran realmente necesarias y sin embargo se habían quedado otras, de mayor importancia, sin especificar.

Pruebas Piloto: Pruebas realizadas por los clientes para comprobar el rendimiento de la aplicación y su respuesta un entorno real de trabajo, así como el cumplimiento de las funcionalidades requeridas.

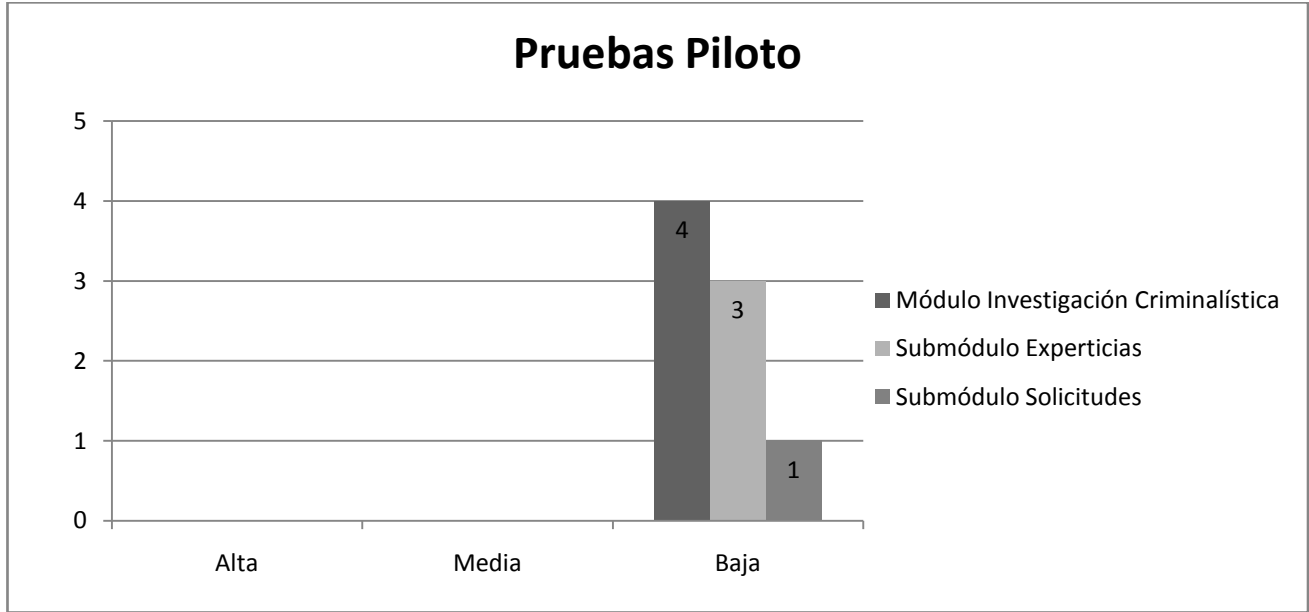


Figura 3-6: No Conformidades detectadas durante las Pruebas Piloto

El análisis de las peticiones de cambio solicitadas por el cliente durante las pruebas de aceptación y el trabajo de corrección de errores realizado por el equipo de trabajo volvió a dejar el software en un mínimo de no conformidades, lo que fue demostrado durante las pruebas piloto. Con solo cuatro no conformidades el sistema y específicamente el submódulo solicitudes pasó a la etapa de soporte.

Luego de concluida cada iteración de prueba se analizaron, por parte del equipo de desarrollo, las no conformidades encontradas para determinar cuáles realmente constituyeron defectos del sistema o presentaban alguna variación según la descripción de los casos de uso y necesitaban ser solucionadas. Las pruebas se realizaron de forma iterativa e incremental, comprobando en cada iteración que hubiesen sido corregidos los errores detectados en la iteración anterior, lo que contribuyó eficientemente a mejorar la calidad y eficiencia del software promoviendo así la satisfacción del cliente sobre el producto.

A continuación se muestra un gráfico resumen del total de las no conformidades encontradas en cada iteración de pruebas, evidenciándose la correcta evolución del software en cuanto a la reducción del número de defectos detectados.

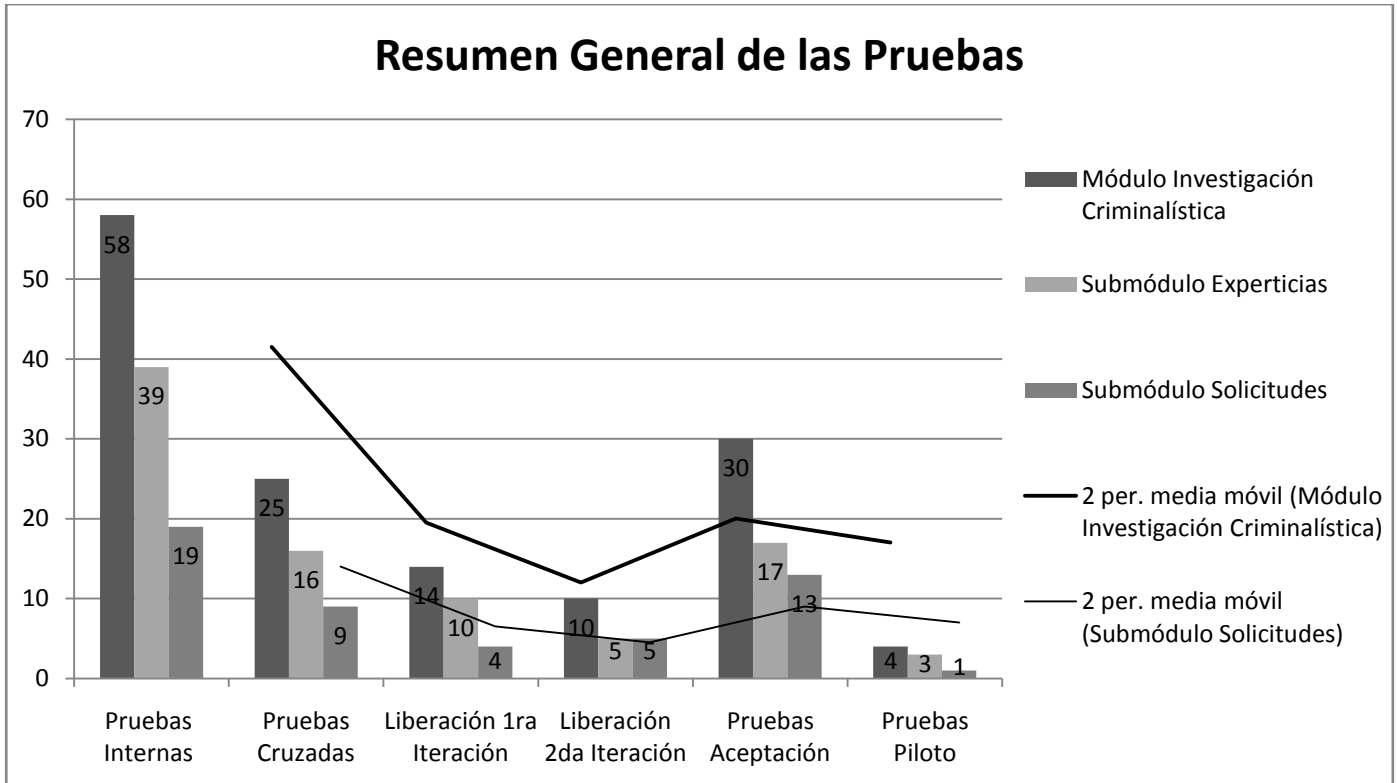


Figura 3-7: Resumen general de las pruebas realizadas al submódulo

Como se muestra en el gráfico anterior, el estado de las no conformidades del submódulo Solicitudes tuvo un comportamiento semejante al módulo Investigación Criminalística en su conjunto (ver línea móvil de la media) a lo largo de las iteraciones de prueba a que fue sometido el SIIPOL. De un total de 141 defectos detectados en el módulo Investigación Criminalística durante todas las iteraciones de prueba, 51 pertenecen al submódulo Solicitudes para un 36 por ciento.

Comparando con el submódulo Experticias perteneciente también al módulo, el número de no conformidades es mucho menor, esto se debe a que el módulo Solicitudes presenta menor cantidad de casos de uso y por tanto de funcionalidades; pero más aún a la calidad con que el equipo de desarrollo implementó la porción del sistema perteneciente al submódulo.

Como resultado principal, el submódulo obtuvo muy buena aceptación por parte del cliente, todas las no conformidades fueron resueltas así como la solución de la mayoría los pedidos de cambio solicitados, presentando en cada iteración de pruebas un sistema más sólido y funcional.

3.3 Conclusiones

En el capítulo se realizó un estudio de los métodos y niveles de pruebas comúnmente aplicados cuando se va a probar un determinado software con el objetivo de lograr una aplicación libre de errores y con la calidad requerida por el cliente.

Gracias a la realización las pruebas descritas en el capítulo se puede concluir que el submódulo Solicitudes tuvo una buena integración con el sistema general y las etapas de prueba validaron la solución cumpliendo con los requisitos y las necesidades del cliente final que tomó parte en dichas pruebas.

Conclusiones

El resultado del actual trabajo es una porción de software funcional, el submódulo Solicitudes perteneciente al módulo Investigación Criminalística, integrado al SIIPOL, con toda la documentación generada durante su diseño e implementación.

La investigación recoge todo el proceso de desarrollo del submódulo, en el cual se realizó un estudio del marco teórico conceptual, tecnologías, tendencias y procesos de desarrollo de software actuales; siendo la base a partir de la cual se diseñó e implementó el resultado de la Ingeniería de Requerimientos.

Posterior a la implementación de la solución, el sistema pasó por varias iteraciones de pruebas, en las cuales fueron detectadas y corregidas una serie de errores existentes, logrando obtener la validación del sistema.

Teniendo en cuenta los resultados obtenidos, se concluye que los objetivos presentados al inicio de la investigación han sido cumplidos en su totalidad.

Recomendaciones

A partir del presente trabajo se proponen las siguientes recomendaciones:

- Realizar un proceso de refactorización a todos los casos de uso, teniendo en cuenta los criterios del equipo de arquitectura, para optimizar el rendimiento de la aplicación.
- Continuar con las pruebas de calidad e integración, para certificar la eficiencia por parte del subsistema desarrollado, con el fin de entregar al cliente un producto de excelencia.
- Capacitar nuevos programadores para las próximas etapas de desarrollo del sistema, usando el trabajo como material de consulta.
- Tener en cuenta el resultado de la implementación de módulo y del sistema en general a la hora de realizar proyectos de iguales características, siempre cumpliendo con las políticas de confidencialidad de la información tratada.
- Sugerir un proyecto similar para la aplicación de las Ciencias Criminalísticas en Cuba.

Bibliografía

1. Cuerpo de Investigaciones Científicas Penales y Criminalísticas. [En línea] 2009. <http://www.cicpc.gov.ve>.
2. **Rodríguez Baryolo, Yunexis y Monagas Reyes, Miguel Ángel.** *Trabajo de Diploma: Ingeniería de Requerimientos del proceso de Criminalística del CICPC.* Ciudad de la Habana : UCI, 2007.
3. Secretaría de Seguridad Pública del Distrito Federal (SSP-DF). [En línea] 2009. <http://portal.ssp.df.gob.mx/Portal/ServiciosenLinea/SIP/IncidenciaDelictivaPrincipal.htm>.
4. Productos & Servicios. *Dmapas*. [En línea] 2009. [Citado el: 18 de Enero de 2009.] http://dmapas.cl/productos_stegpol.htm.
5. Ministerio del Poder Popular para la Comunicación y la Información. [En línea] 2009. http://www.minci.gob.ve/entrevistas/3/181453/se_redujo_violencia.html.
6. Proceso Ágil Unificado. *sitio web The Agile Unified Process (AUP)*. [En línea] 2009. <http://www.ambyssoft.com/unifiedprocess/agileUP.html>.
7. **Larman, Craig.** *UML y Patrones*.
8. Sitio Oficial JAVA. [En línea] 2009. <http://java.com/es/>.
9. **Flanagan, David.** *Java en pocas palabras.* México : McGRAW-HILL INTERAMERICANA EDITORES, S.A., 1998.
10. Java 2 Platform, Enterprise Edition (J2EE) Overview. [En línea] 2009. <http://java.sun.com/j2ee/overview.html>.
11. Red Hat Developer Studio. *sitio Web de Red Hat Developer Studio*. [En línea] 2009. <http://www.redhat.com>.
12. Madritel. *sitio web Madritel*. [En línea] 2009. <http://web.madritel.es/personales3/edcollado/ingsw/tema2/2-6.htm..>
13. Visual Paradigm for UML. *sitio Web de Visual Paradigm for UML*. [En línea] 2009. <http://www.visual-paradigm.com>.

14. Programación en Castellano. [En línea] 2009. <http://www.programacion.com/>.
15. **Walls, Craig y Breidenbach, Ryan**. *Spring in Action*. s.l. : Manning, 2005.
16. **Albiol, Francesc Rosés**. *Introducción a Hibernate*. 2003.
17. Oracle. [En línea] 2009. <http://www.oracle.com/index.html>.
18. **Presman, Roger S**. *Ingeniería de Software. Un enfoque práctico*.
19. **Rumbaugh, James, Jacobson, Ivar y Booch, Grady**. *El Lenguaje Unificado de Modelado. Manual de Referencia*.
20. Evaluación de Software. <http://teleformacion.uci.cu>. [En línea] [Citado el: 10 de marzo de 2009.] <http://teleformacion.uci.cu/mod/resource/view.php?id=14103>.

Glosario de términos

AJAX: Es una técnica de desarrollo para crear aplicaciones web, mediante la cual un grupo de acciones se ejecutan en navegador de los usuarios y mantiene comunicación asíncrona con el servidor en segundo plano.

Ajax4JSF: Es un framework libre y Open Source que adiciona capacidades a las páginas JSF, a través de componentes implementados.

Apache TomCat: es un servidor web con soporte de servlet y JSP. Incluye el compilador Jasper, que compila las páginas JSP convirtiéndolas en servlet.

Bean: En el lenguaje Java es un componente software reusable que evita programar los distintos componentes uno a uno. Existen con la finalidad de ahorrar tiempo al programar.

Bytecode: Es un código intermedio más abstracto que el código máquina. Habitualmente es tratado como un fichero binario que contiene un programa ejecutable.

CICPC: Cuerpo de Investigaciones Científicas Penales y Criminalísticas de Venezuela, institución gubernamental encargada de investigar hechos delictivos.

CNC: Coordinación Nacional de Criminalística: coordinación del CICPC encargada de los asuntos de investigaciones criminalísticas.

Enterprise JavaBean: Es una de las API que forman parte del estándar de construcción de aplicaciones empresariales J2EE.

Framework: Es una estructura de soporte definida en la cual otro proyecto de software puede ser organizado y desarrollado. Típicamente, un framework puede incluir soporte de programas, bibliotecas y un lenguaje interpretado entre otros software para ayudar a desarrollar y unir los diferentes componentes de un proyecto.

Herramienta CASE: Aplicación informática destinada a aumentar la productividad en el desarrollo de software reduciendo el coste de las mismas en términos de tiempo y de dinero, se utiliza para la modelación del sistema.

Hibernate: Framework de capa de persistencia para el lenguaje Java.

HQL: Lenguaje de consultas del framework Hibernate similar al SQL, pero que se refiere a clases y objetos no a tablas de la base de datos.

IDE: Entorno Integrado de Desarrollo, es un programa compuesto por un conjunto de herramientas para un programador. Puede dedicarse exclusivamente para un lenguaje de programación o bien para varios.

J2EE: Es una plataforma de programación para desarrollar y ejecutar software de aplicaciones en Lenguaje de programación Java con arquitectura de N niveles distribuida.

JSF: framework de la capa de presentación para Java.

MVC: Es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario y la lógica de control en tres componentes distintos. El patrón MVC se ve frecuentemente en aplicaciones web.

Red Hat: compañía responsable de la creación y mantenimiento de una distribución del sistema operativo GNU/Linux que lleva el mismo nombre: Red Hat Enterprise Linux, y de otra más, Fedora.

Rich-Faces: Conjunto de librerías para el framework JSF.

Servlet: Es un objeto que se ejecuta en un servidor o contenedor JEE, fue especialmente diseñado para ofrecer contenido dinámico desde un servidor web

SIGEP: Sistema de Gestión de Penitenciaria.

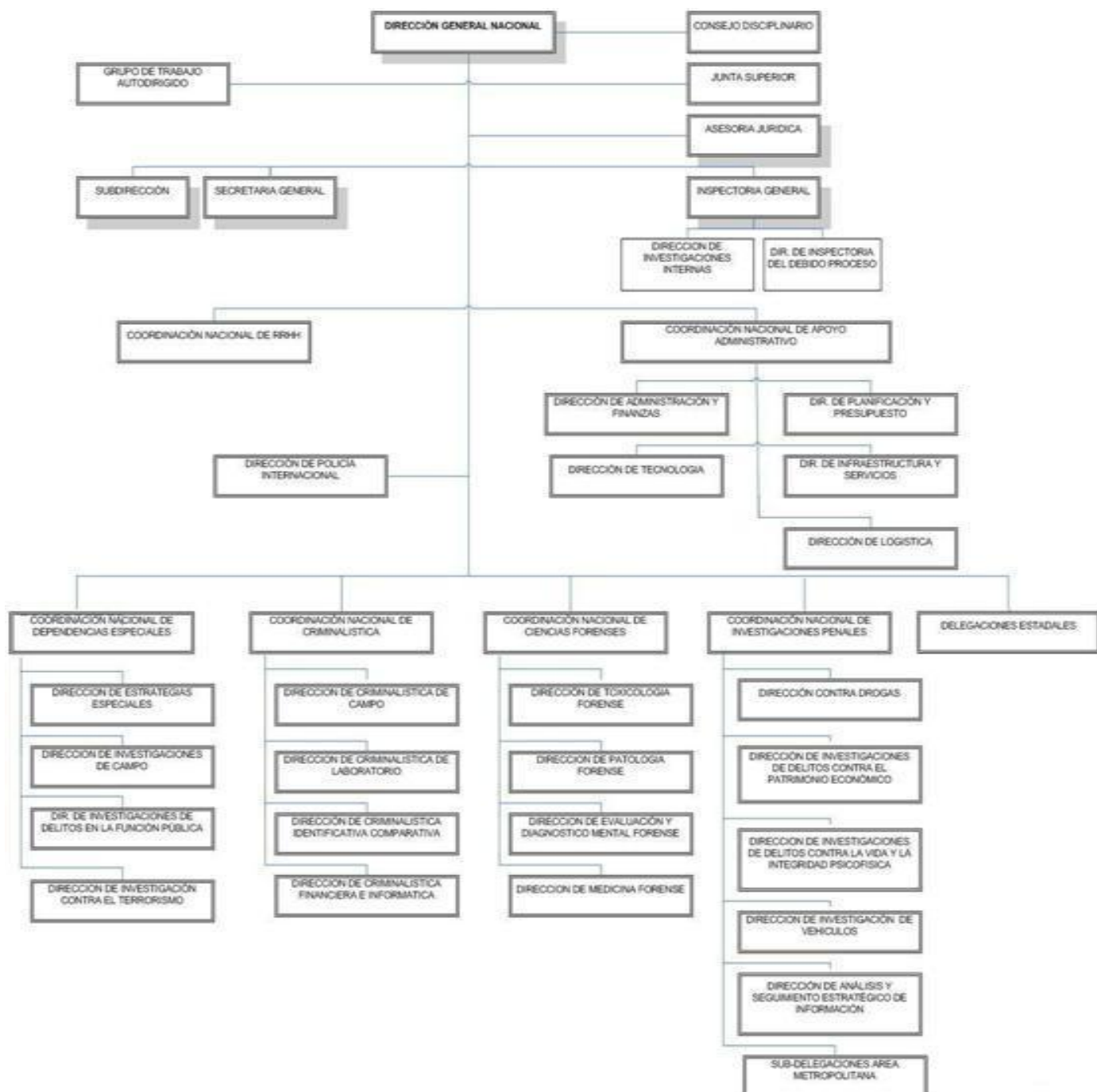
SIIPOL: Sistema de Investigación e Información Policial.

Spring: Framework de la capa de lógica de negocio para Java.

STEGPOL: Es un Sistema de Información Geográfica con tecnología de punta chileno.

Anexos

Anexo 1: Estructura organizacional del CICPC



Anexo 2: Tarjetas CRC para la capa de negocio del submódulo

<p>SolicitudUtil</p> <p>Super Classes: DiligenciaUtil</p> <p>Description: Clase encargada de manejar los aspectos comunes de las solicitudes</p> <p>Atributes:</p> <table border="1"> <thead> <tr> <th>Name</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>solicitudFacade</td> <td>SolicitudFacade</td> </tr> </tbody> </table> <p>Responsibilities:</p> <table border="1"> <thead> <tr> <th>Name</th> <th>Collaborator</th> </tr> </thead> <tbody> <tr> <td>Poblar diligencia</td> <td>Diligencia Funcionario FuncionarioDiligencia AccionRevision</td> </tr> <tr> <td>Asociar elementos al caso via diligencia</td> <td>TipoRelElemento Diligencia NTRelAveriguacionPreliminarElemento ActaProcesoElemento ActaProcesal Evidencia Estado ActaDisciplinaria</td> </tr> <tr> <td>Disociar elementos al caso via diligencia</td> <td>TipoRelElemento Diligencia NTRelAveriguacionPreliminarElemento ActaProcesoElemento ActaProcesal Evidencia Estado ActaDisciplinaria</td> </tr> </tbody> </table>	Name	Description	solicitudFacade	SolicitudFacade	Name	Collaborator	Poblar diligencia	Diligencia Funcionario FuncionarioDiligencia AccionRevision	Asociar elementos al caso via diligencia	TipoRelElemento Diligencia NTRelAveriguacionPreliminarElemento ActaProcesoElemento ActaProcesal Evidencia Estado ActaDisciplinaria	Disociar elementos al caso via diligencia	TipoRelElemento Diligencia NTRelAveriguacionPreliminarElemento ActaProcesoElemento ActaProcesal Evidencia Estado ActaDisciplinaria	<p>SolicitudIdentificacionIndividuoServiceImpl</p> <p>Super Classes: SolicitudCriminalisticaServiceImpl</p> <p>Sub Classes:</p> <p>Description:</p> <p>Atributes:</p> <table border="1"> <thead> <tr> <th>Name</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>solicitudIdentificacionIndividuoDao</td> <td>SolicitudIdentificacionIndividuoDao</td> </tr> <tr> <td>analisisInformacionFacade</td> <td>AnalisisInformacionFacade</td> </tr> </tbody> </table> <p>Responsibilities:</p> <table border="1"> <thead> <tr> <th>Name</th> <th>Collaborator</th> </tr> </thead> <tbody> <tr> <td>Ver solicitud de identificacion de individuo</td> <td></td> </tr> <tr> <td>Buscar personas identificadas</td> <td>PersonaIdentificada</td> </tr> <tr> <td>Obtener peculiaridades predefinidas</td> <td>Peculiaridad</td> </tr> </tbody> </table>	Name	Description	solicitudIdentificacionIndividuoDao	SolicitudIdentificacionIndividuoDao	analisisInformacionFacade	AnalisisInformacionFacade	Name	Collaborator	Ver solicitud de identificacion de individuo		Buscar personas identificadas	PersonaIdentificada	Obtener peculiaridades predefinidas	Peculiaridad																						
Name	Description																																																
solicitudFacade	SolicitudFacade																																																
Name	Collaborator																																																
Poblar diligencia	Diligencia Funcionario FuncionarioDiligencia AccionRevision																																																
Asociar elementos al caso via diligencia	TipoRelElemento Diligencia NTRelAveriguacionPreliminarElemento ActaProcesoElemento ActaProcesal Evidencia Estado ActaDisciplinaria																																																
Disociar elementos al caso via diligencia	TipoRelElemento Diligencia NTRelAveriguacionPreliminarElemento ActaProcesoElemento ActaProcesal Evidencia Estado ActaDisciplinaria																																																
Name	Description																																																
solicitudIdentificacionIndividuoDao	SolicitudIdentificacionIndividuoDao																																																
analisisInformacionFacade	AnalisisInformacionFacade																																																
Name	Collaborator																																																
Ver solicitud de identificacion de individuo																																																	
Buscar personas identificadas	PersonaIdentificada																																																
Obtener peculiaridades predefinidas	Peculiaridad																																																
<p>GestionSolicitudServiceImpl</p> <p>Super Classes: SolicitudCriminalisticaServiceImpl</p> <p>Description: Servicio encargado de manejar la logica de negocio asociada a las experticias asociadas a las solicitudes</p> <p>Atributes:</p> <table border="1"> <thead> <tr> <th>Name</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>solicitudExperticiaCriminalisticaDao</td> <td>SolicitudExperticiaCriminalisticaDao</td> </tr> <tr> <td>analisisInformacionFacade</td> <td>AnalisisInformacionFacade</td> </tr> <tr> <td>nomencladorComunFacade</td> <td>NomencladorComunFacade</td> </tr> <tr> <td>gestionAdministrativaFacade</td> <td>GestionAdministrativaFacade</td> </tr> <tr> <td>nomencladoresGestionAdministrativaFacade</td> <td>NomencladoresGestionAdministrativaFacade</td> </tr> </tbody> </table> <p>Responsibilities:</p> <table border="1"> <thead> <tr> <th>Name</th> <th>Collaborator</th> </tr> </thead> <tbody> <tr> <td>Filtrar por criterio una Solicitud de Experticia Criminalistica</td> <td>SolicitudExperticiaCriminalistica Funcionario DependenciaInterna</td> </tr> <tr> <td>Informes asociados a un Sitio de Suceso</td> <td>Informe SitioSuceso SolicitudExperticiaCriminalistica</td> </tr> <tr> <td>Obtener informes asociados a una solicitud</td> <td>Informe Solicitud</td> </tr> <tr> <td>Obtener informes segun solicitud</td> <td>Informe Solicitud</td> </tr> <tr> <td>Obtener informes no asociados segun el tipo de solicitud</td> <td>Informe SubInforme ActaInvestigativa</td> </tr> <tr> <td>Obtener solicitud por id</td> <td>SolicitudExperticiaCriminalistica</td> </tr> <tr> <td>Obtener funcionario dado experticia asignada</td> <td>Funcionario</td> </tr> <tr> <td>Obtener funcionario dado despacho</td> <td>Funcionario</td> </tr> <tr> <td>Obtener tipos de experticias criminalisticas segun despacho</td> <td>SubInforme</td> </tr> <tr> <td>Responder solicitud con notificacion</td> <td>Solicitud Funcionario</td> </tr> </tbody> </table>	Name	Description	solicitudExperticiaCriminalisticaDao	SolicitudExperticiaCriminalisticaDao	analisisInformacionFacade	AnalisisInformacionFacade	nomencladorComunFacade	NomencladorComunFacade	gestionAdministrativaFacade	GestionAdministrativaFacade	nomencladoresGestionAdministrativaFacade	NomencladoresGestionAdministrativaFacade	Name	Collaborator	Filtrar por criterio una Solicitud de Experticia Criminalistica	SolicitudExperticiaCriminalistica Funcionario DependenciaInterna	Informes asociados a un Sitio de Suceso	Informe SitioSuceso SolicitudExperticiaCriminalistica	Obtener informes asociados a una solicitud	Informe Solicitud	Obtener informes segun solicitud	Informe Solicitud	Obtener informes no asociados segun el tipo de solicitud	Informe SubInforme ActaInvestigativa	Obtener solicitud por id	SolicitudExperticiaCriminalistica	Obtener funcionario dado experticia asignada	Funcionario	Obtener funcionario dado despacho	Funcionario	Obtener tipos de experticias criminalisticas segun despacho	SubInforme	Responder solicitud con notificacion	Solicitud Funcionario	<p>SolicitudCriminalisticaServiceImpl</p> <p>Description: Servicio encargado de manejar la logica de negocio asociada a la Solicitud Criminalistica</p> <p>Atributes:</p> <table border="1"> <thead> <tr> <th>Name</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>investigacionPenalFacade</td> <td>InvestigacionPenalFacade</td> </tr> <tr> <td>solicitudUtil</td> <td>SolicitudUtil</td> </tr> <tr> <td>experticiaCriminalisticaDao</td> <td>ExperticiaCriminalisticaDao</td> </tr> <tr> <td>nomencladorInvestigacionPenalFacade</td> <td>NomencladorInvestigacionPenalFacade</td> </tr> </tbody> </table> <p>Responsibilities:</p> <table border="1"> <thead> <tr> <th>Name</th> <th>Collaborator</th> </tr> </thead> <tbody> <tr> <td>Guardar o enviar solicitud de experticia</td> <td>SolicitudExperticiaCriminalistica</td> </tr> </tbody> </table>	Name	Description	investigacionPenalFacade	InvestigacionPenalFacade	solicitudUtil	SolicitudUtil	experticiaCriminalisticaDao	ExperticiaCriminalisticaDao	nomencladorInvestigacionPenalFacade	NomencladorInvestigacionPenalFacade	Name	Collaborator	Guardar o enviar solicitud de experticia	SolicitudExperticiaCriminalistica
Name	Description																																																
solicitudExperticiaCriminalisticaDao	SolicitudExperticiaCriminalisticaDao																																																
analisisInformacionFacade	AnalisisInformacionFacade																																																
nomencladorComunFacade	NomencladorComunFacade																																																
gestionAdministrativaFacade	GestionAdministrativaFacade																																																
nomencladoresGestionAdministrativaFacade	NomencladoresGestionAdministrativaFacade																																																
Name	Collaborator																																																
Filtrar por criterio una Solicitud de Experticia Criminalistica	SolicitudExperticiaCriminalistica Funcionario DependenciaInterna																																																
Informes asociados a un Sitio de Suceso	Informe SitioSuceso SolicitudExperticiaCriminalistica																																																
Obtener informes asociados a una solicitud	Informe Solicitud																																																
Obtener informes segun solicitud	Informe Solicitud																																																
Obtener informes no asociados segun el tipo de solicitud	Informe SubInforme ActaInvestigativa																																																
Obtener solicitud por id	SolicitudExperticiaCriminalistica																																																
Obtener funcionario dado experticia asignada	Funcionario																																																
Obtener funcionario dado despacho	Funcionario																																																
Obtener tipos de experticias criminalisticas segun despacho	SubInforme																																																
Responder solicitud con notificacion	Solicitud Funcionario																																																
Name	Description																																																
investigacionPenalFacade	InvestigacionPenalFacade																																																
solicitudUtil	SolicitudUtil																																																
experticiaCriminalisticaDao	ExperticiaCriminalisticaDao																																																
nomencladorInvestigacionPenalFacade	NomencladorInvestigacionPenalFacade																																																
Name	Collaborator																																																
Guardar o enviar solicitud de experticia	SolicitudExperticiaCriminalistica																																																
	<p>SolicitudExperticiaRoboABancoServiceImpl</p> <p>Super Classes: SolicitudCriminalisticaServiceImpl</p> <p>Description: Servicio encargado de manejar la logica de negocio asociada a la Solicitud de Robo a banco</p> <p>Atributes:</p> <table border="1"> <thead> <tr> <th>Name</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>nomencladoresGestionAdministrativaFacade</td> <td>NomencladoresGestionAdministrativaFacade</td> </tr> <tr> <td>gestionAdministrativaFacade</td> <td>GestionAdministrativaFacade</td> </tr> </tbody> </table> <p>Responsibilities:</p> <table border="1"> <thead> <tr> <th>Name</th> <th>Collaborator</th> </tr> </thead> <tbody> <tr> <td>Obtener solicitud por id</td> <td>SolicitudExperticiaCriminalistica</td> </tr> <tr> <td>Guardar o enviar solicitud de experticia</td> <td>SolicitudExperticiaCriminalistica Novedad</td> </tr> </tbody> </table>	Name	Description	nomencladoresGestionAdministrativaFacade	NomencladoresGestionAdministrativaFacade	gestionAdministrativaFacade	GestionAdministrativaFacade	Name	Collaborator	Obtener solicitud por id	SolicitudExperticiaCriminalistica	Guardar o enviar solicitud de experticia	SolicitudExperticiaCriminalistica Novedad																																				
Name	Description																																																
nomencladoresGestionAdministrativaFacade	NomencladoresGestionAdministrativaFacade																																																
gestionAdministrativaFacade	GestionAdministrativaFacade																																																
Name	Collaborator																																																
Obtener solicitud por id	SolicitudExperticiaCriminalistica																																																
Guardar o enviar solicitud de experticia	SolicitudExperticiaCriminalistica Novedad																																																
	<p>SolicitudExperticiaReveladoRolloFotograficoServiceImpl</p> <p>Super Classes:</p> <p>Description: Servicio encargado de manejar la logica de negocio asociada a la Solicitud de rollo fotografico</p> <p>Atributes:</p> <table border="1"> <thead> <tr> <th>Name</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>solicitudExperticiaReveladoRolloFotograficoDao</td> <td>SolicitudExperticiaReveladoRolloFotograficoDao</td> </tr> <tr> <td>analisisInformacionFacade</td> <td>AnalisisInformacionFacade</td> </tr> <tr> <td>gestionAdministrativaFacade</td> <td>GestionAdministrativaFacade</td> </tr> <tr> <td>nomencladoresGestionAdministrativaFacade</td> <td>NomencladoresGestionAdministrativaFacade</td> </tr> <tr> <td>seguridadFacade</td> <td>SeguridadFacade</td> </tr> <tr> <td>experticiaCriminalisticaDao</td> <td>ExperticiaCriminalisticaDao</td> </tr> <tr> <td>carga</td> <td>Carga</td> </tr> </tbody> </table> <p>Responsibilities:</p> <table border="1"> <thead> <tr> <th>Name</th> <th>Collaborator</th> </tr> </thead> <tbody> <tr> <td>Generar numero de solicitud</td> <td></td> </tr> <tr> <td>Guardar y enviar solicitud de experticia</td> <td>RolloFotografico</td> </tr> <tr> <td>Obtener por id</td> <td>Entidad RolloFotografico</td> </tr> </tbody> </table>	Name	Description	solicitudExperticiaReveladoRolloFotograficoDao	SolicitudExperticiaReveladoRolloFotograficoDao	analisisInformacionFacade	AnalisisInformacionFacade	gestionAdministrativaFacade	GestionAdministrativaFacade	nomencladoresGestionAdministrativaFacade	NomencladoresGestionAdministrativaFacade	seguridadFacade	SeguridadFacade	experticiaCriminalisticaDao	ExperticiaCriminalisticaDao	carga	Carga	Name	Collaborator	Generar numero de solicitud		Guardar y enviar solicitud de experticia	RolloFotografico	Obtener por id	Entidad RolloFotografico																								
Name	Description																																																
solicitudExperticiaReveladoRolloFotograficoDao	SolicitudExperticiaReveladoRolloFotograficoDao																																																
analisisInformacionFacade	AnalisisInformacionFacade																																																
gestionAdministrativaFacade	GestionAdministrativaFacade																																																
nomencladoresGestionAdministrativaFacade	NomencladoresGestionAdministrativaFacade																																																
seguridadFacade	SeguridadFacade																																																
experticiaCriminalisticaDao	ExperticiaCriminalisticaDao																																																
carga	Carga																																																
Name	Collaborator																																																
Generar numero de solicitud																																																	
Guardar y enviar solicitud de experticia	RolloFotografico																																																
Obtener por id	Entidad RolloFotografico																																																

Anexo 3: Resto de las interfaces de usuario del submódulo**Anexo 3.1: IU Incluir Solicitud de Experticia a Entidades Financieras**

SIIPOL | SISTEMA DE INVESTIGACIÓN E INFORMACIÓN POLICIAL

Bienvenido(a): AAIL YADIEL RAMOS RODRIGUEZ 22/04/2009 ... 17:02 Inicio Ayuda Salir

AGENDA DE TRABAJO

- Notificaciones
- Asignaciones
- Recepciones
- Aprobaciones
- Revisiones
- Remisiones
- Borradores
- Archivos

MENÚ PRINCIPAL

- Acceso
- Gestión de Despacho
- Atención Telefónica
- Investigación Penal
- Aprehensión
- Archivos Históricos
- Investigación Criminalística
 - Solicitudes
 - Experticia a Entidades F...
 - Revelado Rollo Fotográf...
 - Registros de Informes
 - Consultas
- Asesoría Jurídica
- Investigación en Ciencias ...
- Análisis de Información
- Estadísticas
- Dotación de Equipos Polic...
- Evidencia
- Auditoría
- Administración

INCLUIR SOLICITUD DE EXPERTICIA A ENTIDADES FINANCIERAS

Dependencia a Responder

Organismo	Responsable
SUDEBANC	Kenny Lopez

Div. Contra Robo
 Div. Contra Hurto

No. Expediente:

Organismo Afectado

Entidad Financiera
 Casa de Cambio
 Transporte de Valores

Grupo Bancario	Entidad	Dirección	Tipo de Edificación
<input type="text" value="Banco Fondo Común"/>	<input type="text" value="Oficina Parque Central"/>	Distrito Capital, Libertador, Candelaria	<input type="text" value="Edificio"/>

Entidad Notificante

Dependencia del CICPC
 Organismo Afectado
 Otros

Funcionario

Nombres y Apellidos	C MAURA ZORAIDA CASANOVA DE RIVERO
Credencial	0017323
Cargo	Jefe de Despacho

Descripción

Anexo 3.2: IU Incluir Solicitud de Rollo Fotográfico

SIIPOL

SISTEMA DE INVESTIGACIÓN E INFORMACIÓN POLICIAL

Bienvenido(a): AAII.YADIEL RAMOS RODRIGUEZ

22/04/2009 ... 17:19


Inicio Ayuda Salir

AGENDA DE TRABAJO

- Notificaciones
- Asignaciones
- Recepciones
- Aprobaciones
- Revisiones
- Remisiones
- Borradores
- Archivos

MENÚ PRINCIPAL

- ▶ Acceso
- ▶ Gestión de Despacho
- ▶ Atención Telefónica
- ▶ Investigación Penal
- ▶ Aprehensión
- ▶ Archivos Históricos
- ▶ Investigación Criminalística
- ▶ Asesoría Jurídica
- ▶ Investigación en Ciencias ...
- ▶ Análisis de Información
- ▶ Estadísticas
- ▶ Dotación de Equipos Polic...
- ▶ Evidencia
- ▶ Auditoría
- ▶ Administración



INCLUIR SOLICITUD DE ROLLO FOTOGRAFICO

Dependencia a Responder

Dirección General

Importancia

Normal

Descripción

Incluir Rollo Fotográfico

No. Expediente

No. Temporal	Actas Procesales	Eliminar
1	K-09-0001-00001	Eliminar
2	K-09-0001-00045, K-09-0001-00030	Eliminar
Total: 2		

Enviar a Dependencia

Dependencia Del CICPC Organismo Externo

Dependencia

Departamento de Aprehensión

Funcionario Jefe

Nombres y Apellidos	SC HEBERTO JOSE ALFONSO OLANO
Credencial	0020802
Cargo	Jefe de Despacho

Responder a Dependencia

Dependencia Del CICPC Organismo Externo Ninguno

Dependencia

Asesoría Jurídica Nacional

Funcionario Jefe

Nombres y Apellidos	AAI JHONNY ALEXI LAYA PAZ CASTILLO
Credencial	0031357
Cargo	Adjunto

Dependencia	Funcionario	No. Credencial / Identificación	Eliminar
Dirección General	DTVE Kenny López Espinosa	8510061	Eliminar
Apoyo Administrativo	CJ Benito Artigas	0013133	Eliminar
Total: 2			

Anexo 3.3: IU Incluir Solicitud de Identificación de Individuo

SIIPOL | SISTEMA DE INVESTIGACIÓN E INFORMACIÓN POLICIAL

 Bienvenido(a): AAII YADIEL RAMOS RODRIGUEZ
 22/04/2009 ... 14:42
Inicio
Ayuda
Salir

AGENDA DE TRABAJO

MENÚ PRINCIPAL

- ▶ Acceso
- ▶ Gestión de Despacho
- ▶ Atención Telefónica
- ▶ Investigación Penal
- ▶ Aprehensión
- ▶ Archivos Históricos
- ▶ Investigación Criminalística
- ▶ Asesoría Jurídica
- ▶ Investigación en Ciencias ...
- ▶ Análisis de Información
- ▶ Estadísticas
- ▶ Dotación de Equipos Polic...
- ▶ Evidencia
- ▶ Auditoría
- ▶ Administración

INCLUIR SOLICITUD DE IDENTIFICACIÓN DE INDIVIDUO

Solicitud
Elementos

Reseñas asociadas

Tipo de Reseña
R_10 Identificación de INTERPOL
Adicionar

Reseña actual	R_22 Reseña para Peritación Civil
---------------	-----------------------------------

Tipo de Reseña	Eliminar
R_ A Reseña para Aspirante a CICPC	Eliminar
R_ 22 Reseña para Peritación Civil	Eliminar
Total: 2	

Datos de la Solicitud

Importancia
Normal

Descripción

Enviar a Entidad

Dependencia Interna Dependencia
 Entidad externa División de Lofoscopia

Dirigido a	IJ José Ramón Gonzalez Cedeño
Cargo	Jefe de la Div. Lofoscopia
Credencial	0015746

Responder a Entidades

Dependencia Interna Dependencia
 Entidad externa División contra Hurto

Dirigido a	AS Frenando J Barreto M
Cargo	Administrativo
Credencial	0013498

Adicionar

Dependencia	Dirigido a	Credencial	Eliminar
No existen registros			

Base Legal

Ley Artículo

NLC
Adicionar

De Conformidad con lo establecido en la Ley	Artículo	Eliminar
NLC	Artículo 1	Eliminar
Total: 1		

Personalizar Comunicación
Vista Previa
Incluir
Limpiar
Cancelar

Anexo 3.4: IU Tratar Reseñas para Solicitud – Editar Reseña

SIIPOL | SISTEMA DE INVESTIGACIÓN E INFORMACIÓN POLICIAL

 Bienvenido(a): AAIL YADIEL RAMOS RODRIGUEZ.
 22/04/2009 ... 14:52
Inicio
Ayuda
Salir

AGENDA DE TRABAJO

- Notificaciones
- Asignaciones
- Recepciones
- Aprobaciones
- Revisiones
- Remisiones
- Borradores
- Archivos

MENÚ PRINCIPAL

- ▶ Acceso
- ▶ Gestión de Despacho
- ▶ Atención Telefónica
- ▶ Investigación Penal
- ▶ Aprehensión
- ▶ Archivos Históricos
- ▶ Investigación Criminalística
- ▶ Asesoría Jurídica
- ▶ Investigación en Ciencias ...
- ▶ Análisis de Información
- ▶ Estadísticas
- ▶ Dotación de Equipos Polic...
- ▶ Evidencia
- ▶ Auditoría
- ▶ Administración

INCLUIR SOLICITUD DE IDENTIFICACIÓN DE INDIVIDUO

Solicitud
Reseña
Consultar Persona
Elementos

Datos Generales

Solicitado por Tarjeta de Huellas Dactilares

Datos generales

Cédula Primer Nombre Segundo Nombre

Primer Apellido Segundo Apellido Fecha Nacimiento

Natural de

Otro País

Venezuela

Desconocido

Profesión

Carrera

Especialidad

Profesión

Carrera	Especialidad	Profesión	Eliminar
Informática	Software de Gestión	Analista	Eliminar
Total: 1			

Nombres o Apodos Asociados

Nombre

Tipo de Nombre

Nombre	Tipo de Nombre	Eliminar
Oráculo	Apodo	Eliminar
Total: 1		

Dirección

Estado Municipio

Parroquia Tipo Dirección

Tipo Residencia Localización

Calle No. Residencia

Estado	Municipio	Parroquia	Tipo Dirección	Tipo Residencia	Caserío	Calle	No.	Eliminar
Amazonas	Alto Orinoco	Capital Alto Orinoco	Estudia	Casa	Piloto	5ta	5112	Eliminar
Observaciones								
Total: 1								

Anexo 3.5: IU Tratar Reseñas para Solicitud – Consultar Persona

SIIPOL | SISTEMA DE INVESTIGACIÓN E INFORMACIÓN POLICIAL

 Bienvenido(a): AAI-YADIEL RAMOS RODRIGUEZ.
 22/04/2009 ... 15:03
Inicio
Ayuda
Salir

AGENDA DE TRABAJO

- Notificaciones
- Asignaciones
- Recepciones
- Aprobaciones
- Revisiones
- Remisiones
- Borradores
- Archivos

MENÚ PRINCIPAL

- ▶ Acceso
- ▶ Gestión de Despacho
- ▶ Atención Telefónica
- ▶ Investigación Penal
- ▶ Aprehensión
- ▶ Archivos Históricos

INCLUIR SOLICITUD DE IDENTIFICACIÓN DE INDIVIDUO

Solicitud
Reseña
Consultar Persona
Elementos

Cédula

No. Pasaporte

Primer Nombre

Segundo Nombre

Primer Apellido

Segundo Apellido

Nombres y Apellidos ▲	Cédula	Pasaporte	Sexo	Estado	Natural De	Indocumentado	Validado SAIME	Asociar
Jose Luis Garcia Perez	V83563454			Sin Registros Policiales		NO	No	Asociar
Total: 1								

[Personalizar Comunicación](#)

Anexo 3.6: IU Completar Solicitud de Experticia de Avalúo

SIIPOL | SISTEMA DE INVESTIGACIÓN E INFORMACIÓN POLICIAL

Bienvenido(a): AAILYADIEL RAMOS RODRIGUEZ. 22/04/2009 ... 15:06 Inicio Ayuda Salir

AGENDA DE TRABAJO

- Notificaciones
- Asignaciones
- Recepciones
- Aprobaciones
- Revisiones
- Remisiones
- Borradores
- Archivos

MENÚ PRINCIPAL

- Acceso
- Gestión de Despacho
- Atención Telefónica
- Investigación Penal
- Aprehensión
- Archivos Históricos
- Investigación Criminalística
- Asesoría Jurídica

INCLUIR SOLICITUD DE EXPERTICIA CRIMINALÍSTICA

Solicitud **Elementos**

Intangibles Muebles **Inmuebles**

Elementos Asociados

Nombre ▲	Dirección	Disociar
No existen registros		

Elementos correspondientes al Acta Procesal

Nombre ▲	Dirección	Asociar
Café de Pedro	Falcón, Democracia, Piedra Grande	Asociar
SitioA	Barinas, Cruz Paredes, Barrancas, 112, d32, 321 21	Asociar
Total: 2		

Imprimir/Exportar

Personalizar Comunicación Vista Previa Incluir Limpiar Cancelar

Anexo 4: Caso de Prueba Gestionar Solicitud de Experticia Criminalística

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad	Flujo central
SC 1: Gestionar Solicitud de Experticia Criminalística	EC 1.1: El caso de uso se inicia cuando el actor selecciona la opción de realizar una acción sobre una Solicitud de Experticia.	Brinda la posibilidad de realizar varias acciones.	<ul style="list-style-type: none"> - Asignaciones - Actas Procesales - En curso - Ver Acta procesal - Tipo (Solicitud de Experticia Criminalística)
	EC 1.2: Selecciona la opción de realizar una nueva Solicitud de Experticia.	Brinda la posibilidad de introducir los datos de la Solicitud de Experticia Criminalística.	
	EC 1.3: Introduce los datos de una Solicitud de Experticia.	Brinda la posibilidad de seleccionar el despacho a responder, incluir los despachos interesados, escribir la descripción.	
	EC 1.4: Selecciona la opción de ver una vista previa.	Muestra una vista previa de cómo queda confeccionado la Solicitud de Experticia, con todos los datos que debe presentar.	
	EC 1.5: Selecciona la opción de Incluir la solicitud.	Valida los datos. Muestra un mensaje de información "Se ha incluido un nuevo elemento."	
	EC 1.6: El actor selecciona la opción de Cancelar.	Elimina los datos creados y muestra el escritorio de trabajo del usuario autenticado.	
	EC 1.7: El actor selecciona la opción de relacionar una nueva evidencia, persona o sitio del suceso.	Brinda la posibilidad de relacionar una nueva persona, sitio del suceso o evidencia relacionada al caso.	
	EC 1.8: El actor selecciona la opción de adicionar a un despacho interno como destino del informe respuesta.	Brinda la posibilidad de que el actor seleccione un nuevo despacho a enviar la respuesta de la solicitud y lo adicione a	

		una lista de despachos.	
	EC 1.9: Selecciona un despacho interno	Muestra el funcionario responsable del despacho	
	EC 1.10: Selecciona adicionar el despacho al listado de envíos de respuestas.	Muestra el despacho en el listado de envío de respuestas.	
	EC 1.11: El actor selecciona la opción de configurar el encabezado de la entidad.	Brinda la posibilidad de seleccionar un nuevo encabezado o hacerlo editable para que el actor lo modifique a su gusto.	
	EC 1.12: El actor selecciona la opción que permite imprimir o exportar a PDF.	Brinda la posibilidad de Imprimir o exportar a PDF.	
	EC 1.13: Existen datos incompletos.	Muestra el mensaje de información “Existen campos vacíos que son obligatorios, por favor, complete estos datos.”	
	EC 1.14: Existen datos incorrectos	Muestra el mensaje de información “Existen campos escritos incorrectamente, por favor, rectifique estos datos.”	
SC 2: Registrar Solicitud de Experticia Criminalística.	EC 2.1: El actor selecciona la opción de asociar la comunicación a una solicitud de experticia Criminalística.	Muestra los datos predeterminados, tomados de la Comunicación asociada y brinda la posibilidad de introducir los datos de la Solicitud de Experticia Criminalística.	<ul style="list-style-type: none"> - Menú principal - Gestión de despacho - Registros - Correspondencia de Entrada - Tipo de Comunicación (Solicitud) - Tipo de Diligencia (Solicitud de Experticia Criminalística)
	EC 2.2: El actor selecciona el tipo de experticia e introduce la descripción de la solicitud.	Valida los datos , crea Solicitud de Experticia Criminalística.	
	EC 2.3: El actor selecciona la opción de Incluir la solicitud.	Valida los datos. Muestra un mensaje de información “Se ha	

		incluido un nuevo elemento.”	
	EC 2.4: El actor selecciona la opción de Cancelar.	Elimina los datos creados y muestra el escritorio de trabajo del usuario autenticado.	
	EC 2.5: El actor selecciona la opción de ver la comunicación escaneada adjunta como archivo digital .organismo externo.	Muestra la imagen de la comunicación digital.	
	EC 2.6: El actor selecciona la opción de incluir un elemento.	Brinda la posibilidad de incluir un elemento.	
	EC 2.7: El actor selecciona la opción de disociar una persona, evidencia o sitio del suceso que se encontraba en el sistema.	Elimina la relación del elemento con la solicitud.	
	EC 2.8: Existen datos incompletos.	Muestra el mensaje de información “Existen campos vacíos que son obligatorios, por favor, complete estos datos.”	
	EC 2.9: Existen datos incorrectos.	Muestra el mensaje de información “Existen campos escritos incorrectamente, por favor, rectifique estos datos.”	
SC 3: Ver datos de una Solicitud de Experticia Criminalística	EC 3.1: Selecciona la opción de ver los datos de la solicitud.	Muestra los datos de Solicitud de Experticia Criminalística y permite imprimirlos o exportarlos a PDF y de acuerdo a los permisos del usuario permite: Modificar los datos de la entidad.	- Revisiones - Experticias Criminalísticas - Solicitudes - Vínculo Experticia
	EC 3.2: Selecciona la opción de salir de la vista actual.	Muestra la vista anterior.	
	EC 3.3: El actor selecciona la opción de Modificar los datos de la solicitud.	Brinda la posibilidad de modificar los datos de la solicitud.	
	EC 3.4: El actor selecciona la	Brinda la posibilidad de	

	opción de imprimir o exportar a PDF una Solicitud de Experticia Criminalística.	Imprimir o exportar a PDF.	
	EC 3.5: El actor selecciona la opción de ver los datos de la evidencia seleccionada.	Muestra los datos de la evidencia seleccionada.	
SC 4: Modificar datos de una Solicitud de Experticia Criminalística	EC 4.1: Selecciona la opción de modificar los datos de la Solicitud de Experticia Criminalística	Muestra los datos de Solicitud de Experticia Criminalística y brinda la posibilidad de cambiar los valores que originaron la no aceptación de la solicitud.	<ul style="list-style-type: none"> - Aprobaciones - Experticias Criminalísticas - En Curso - Solicitudes - Rectificar
	EC 4.2: Modifica los datos que necesite y selecciona la opción de actualizar la solicitud.	Valida los datos y actualiza los datos de Solicitud de Experticia Criminalística, muestra un mensaje de información “Se han actualizado los cambios sobre el elemento.”	
	EC 4.3: El actor selecciona la opción de Cancelar.	Elimina los datos creados, muestra el escritorio de trabajo del usuario autenticado.	
	EC 4.4: El actor selecciona la opción de ver la comunicación escaneada adjunta como archivo digital.	Muestra la imagen de la comunicación digital.	
	EC 4.5: Existen datos incompletos.	Muestra el mensaje de información “Existen campos vacíos que son obligatorios, por favor, complete estos datos.” Y muestra un indicador sobre los campos vacíos.	
	EC 4.6: Existen datos incorrectos	Muestra el mensaje de información “Existen campos escritos incorrectamente, por favor, rectifique estos datos.”	
	EC 4.7: Relacionar elemento	Muestra un listado y	

	a la solicitud	permite cancelar la operación.	
	EC 4.8: Selecciona la opción de cancelar la operación.	Regresa a la vista anterior.	
	EC 4.9: El actor selecciona la opción de asociar una Persona.	Muestra un listado de personas relacionadas al caso o al informe del cual se originó la solicitud.	
	EC 4.10: El actor selecciona la opción de asociar una Evidencia.	Muestra un listado de evidencias relacionadas al caso o al informe del cual se originó la solicitud.	
	EC 4.11: El actor selecciona la opción de asociar un Sitio del Suceso.	Muestra un listado de sitios de suceso relacionadas al caso o al informe del cual se originó la solicitud.	
	EC 4.12: El actor selecciona la opción de disociar de la Solicitud un elemento asociado por él.	Elimina la relación de la Persona, Sitio del suceso o Evidencia seleccionada de la lista de elementos vinculados a la solicitud y disocia el elemento de la solicitud.	
	EC 4.13: El actor selecciona la opción de imprimir o exportar a PDF una Solicitud de Experticia Criminalística	Brinda la posibilidad de Imprimir o exportar a PDF	
SC 5: Incluir nuevo Elemento asociado a la Solicitud de Experticia	EC 5.1: Brinda la posibilidad de incluir un nuevo elemento.	Incluye un nuevo elemento.	
	EC 5.2: Selecciona la opción de incluir una nueva Evidencia.	Brinda la posibilidad de seleccionar el tipo de Evidencia a incluir:	
	EC 5.3: Selecciona la opción de incluir un nuevo Objeto.	Brinda la posibilidad de incluir un nuevo Objeto.	
	EC 5.4: Selecciona la opción de convertir el elemento en evidencia.	Brinda la posibilidad de convertir el elemento en Evidencia y muestra en una lista la nueva	

		Evidencia asociada a la solicitud.	
	EC 5.5: El actor selecciona la opción de incluir una nueva Persona	Brinda la posibilidad de introducir una nueva Persona y muestra la nueva Persona asociada a la Solicitud de Experticia Criminalística y sus datos.	
	EC 5.6: El actor selecciona la opción de incluir un nuevo Sitio Suceso.	Brinda la posibilidad de introducir datos.	
	EC 5.7: Introduce la dirección del sitio y selecciona la opción de incluirlo asociado a la solicitud.	Crea un nuevo sitio del suceso y asocia el sitio a la Solicitud de Experticia Criminalística.	
	EC 5.8: Selecciona la opción de incluir nueva Arma.	Brinda la posibilidad de incluir una nueva Arma.	
	EC 5.9: Selecciona la opción de incluir nuevo Vehículo.	Brinda la posibilidad de incluir nuevo vehículo.	