

Universidad de las Ciencias Informáticas
Facultad 4



Título: Análisis y diseño de los Módulos Traslados y Conducción del Sistema de Gestión Penitenciario Cubano.

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

Autores: Dayneris de la C. Gálvez Casanova
José Livan Echemendía Lazo

Tutor: Ing. Javier Heredia Ruiz
Co-Tutora: Ing. Yanay Viera Lorenzo
Mayo, 2010

DECLARACIÓN DE AUTORÍA

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

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

Dayneris de la Caridad Gálvez Casanova

Firma del Autor

José Livan Echemendía Lazo

Firma del Autor

Ing. Javier Heredia Ruiz

Firma del Tutor

Ing. Yanay Viera Lorenzo

Firma del Co-Tutor

DATOS DE CONTACTO

Tutor: Ing. Javier Heredia Ruiz

Graduado de Ingeniero en Ciencias Informáticas en el año 2007, ha participado como tribunal de tesis, tutor y co-tutor, se ha desempeñado como especialista de calidad del proyecto ERP-Cuba, luego como analista en el centro de desarrollo GEITEL de la Facultad 10, se desempeña como profesor del departamento de Sistemas Digitales en la misma facultad, ha presentado trabajo en el evento científico UCIENCIA en el año 2008, en la cuarta edición del XVI FORUM de Ciencia y Técnica a nivel de base, cuenta con 2 años y medio de experiencia laboral.

Correo: jheredia@uci.cu

Co-Tutora: Ing. Yanay Viera Lorenzo

Graduada de Ingeniera en Ciencias Informáticas en el año 2007, se desempeña como analista principal del proyecto SIGEP-Cu, profesora de Ingeniería de Software de la Facultad 15, ha sido tutora, co-tutora y tribunal de tesis, ha participado en eventos científicos a nivel de base, cuenta con 2 años y medio de experiencia laboral.

Correo: yviera@uci.cu

AGRADECIMIENTOS

No me quedan dudas de que esta es la parte más difícil, recoger en solo dos páginas los nombres de todas las personas a las cuales les agradezco infinitamente por haberme brindado su apoyo incondicional durante estos 5 años y con los cuales siempre estaré en deuda, porque no me imagino que hubiese sido de mí si en los mejores y en los no tan buenos momentos no hubiese podido contar con ellos. Por cada uno de ellos siento un profundo cariño y una gran admiración, por lo que doy gracias a Dios porque existan.

Quisiera comenzar por las personas más importantes de mi vida. Mi mamá Elizabeth, mi papá Diosdado y mi abuela Zoraida (“Yaya”) muchas gracias por el amor y la comprensión que me han dado, por estar a mi lado en cada paso de mi vida, por creer en mí, por ser mi guía, mi sostén, mi motor impulsor, sin ustedes no estudiara haciendo realidad mis sueños que son también sus sueños, por ustedes soy todo lo que soy hoy, a ustedes les debo todo.

A mi hermanita Daynoris por su amor y cariño, por estar siempre pendiente de mí y ser muchas veces mi confidente. A mis tías Sonia y María Eugenia por ser otras dos madres para mí. A mis primos Oniel, Melissa Milagro y Maura Elisa por estar siempre a mi lado. A mi sobrinito Jesús Dariel por ser un motivo más por el cual poder seguir adelante.

Muchas gracias a Adelina (“Ade”) por quererme como a una hija, por escucharme y darme sus consejos, por ser una de las personas que más he querido en esta universidad. Al profe Hugo por su amistad sincera.

A mi compañero de tesis por ser un amigo genial, por reír y llorar conmigo, por la gran amistad y confianza brindada durante estos 5 años, muchas gracias a ti por ser mucho más que un amigo un hermano para mí.

A Greisy que fue la primera persona que me extendió su mano amiga cuando acababa de llegar a la escuela y con la siempre he contado, por su dedicación y preocupación.

A todos los que he considerado mis amigos en esta universidad, darle las gracias por haberlos conocido, por tenerme presente, por saber que puedo contar con ustedes más de una vez, por ser siempre los mismos.

A Javier y Yanay los tutores por dedicarnos tanto tiempo y esfuerzo en aras de que todo saliera bien.

A la Revolución y a Fidel por la valiosa oportunidad de estudiar en esta universidad.

A todos muchísimas Gracias.

Dayneris

Agradezco a todas las personas que estudiaron junto a mí en los mejores momentos y en los más difíciles en estos 5 años de mi vida. A mis padres por su gran esfuerzo y apoyo incondicional

ante cualquier situación durante el transcurso de mi carrera universitaria y en mi vida. A mi familia en general que ha sido mi eslabón fundamental para poder llegar a la culminación de mis estudios exitosamente. A mi compañera de tesis por haber transitado estos años conmigo y compartir todas las alegrías y las tristezas vividas y convertirse en una hermana más. A mi tutor y co-tutor(a) que con su apoyo hicieron posible la realización de este trabajo. A la Revolución y al Comandante en Jefe Fidel Castro Ruz, por haberme dado la posibilidad de estudiar en esta Universidad para poder optar por un título universitario.

A todos muchas Gracias.

José Liván

DEDICATORIA

De Dayneris:

A mi mamá por no apartarme un minuto de su mente, por darme toda la fuerza del mundo, por su gran sacrificio y por ser la persona que más me ha apoyado y ayudado.

A mi papá, mi abuela y mi hermana por estar siempre a mi lado y confiar en mí.

De José Liván:

Dedico el presente trabajo a mis padres que han sido mi guía y serán por siempre la fuente de mi inspiración.

A mi hermano, mi cuñada, mis sobrinos, mis tías, mis primos que han estado conmigo a cada momento y siempre han mantenido una gran admiración hacia mí.

A todas aquellas personas que de una forma u otra aportaron su granito de arena para hacer realidad este sueño.

PENSAMIENTO

“Es fácil conseguir la admiración de los que nos rodean, pero es mejor empeñar nuestro esfuerzo por lograr su efecto”.

Fidel Castro.

RESUMEN

El presente trabajo abarca el análisis y diseño de los módulos *Traslados y Conducción* del proyecto Sistema de Gestión Penitenciario Cubano. Se describen las técnicas y herramientas utilizadas, así como la aplicación de patrones dentro del flujo de desarrollo de software siguiendo la metodología y procesos del Proceso Unificado de Desarrollo. Los artefactos se generaron usando como lenguaje de modelado el Lenguaje Unificado de Modelado y auxiliados por el Visual Paradigm como herramienta de Ingeniería de Software Asistida por Computadora. Mediante el flujo de trabajo de análisis y diseño se obtuvieron artefactos que posibilitan a los desarrolladores la automatización de una plataforma capaz de satisfacer las necesidades de los clientes. Para garantizar la calidad de los artefactos generados se aplicaron métricas de diseño, a través de las cuales se obtuvieron resultados positivos.

Palabras claves. *Análisis, diseño, traslado, conducción.*

ÍNDICE

INTRODUCCIÓN	16
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA	19
1.1 Introducción.....	19
1.2 Sistemas Penitenciarios.....	19
1.2.1 Registro Legal.....	21
1.3 Subsistema Registro Legal del SIGEP-Cu.....	21
1.3.1 Módulo Traslados	23
1.3.1.1 Causas para efectuar el proceso de traslado de reclusos.....	23
1.3.1.2 Traslados interprovinciales	24
1.3.1.3 Traslados interpenales.....	24
1.3.1.4 Traslado Expreso.....	24
1.3.1.5 Traslado Ordinario.....	24
1.3.2 Módulo Conducción.....	25
1.3.2.1 Causas para efectuar el proceso de conducción de reclusos.....	25
1.4 Estado del arte de los Sistemas Penitenciarios.....	25
1.4.1 Soluciones nacionales	26
1.4.2 Soluciones extranjeras.....	27
1.5 Surgimiento del Sistema de Gestión Penitenciario Cubano SIGEP-Cu	29
1.6 Acerca de los requerimientos	31
1.6.1 Objetivos de la captura de requisitos.....	31

1.6.2 Especificación de los requerimientos del sistema.....	32
1.6.3 Características que debe poseer una Especificación de Requisitos de Software (ERS)	32
1.6.4 Técnicas para identificar requerimientos.....	34
1.6.5 Técnicas de validación de requisitos	35
1.6.6 Aplicación de patrones de casos de uso	36
1.7 Metodología de desarrollo de software a utilizar: Rational Unified Process (RUP).....	38
1.8 Herramientas y lenguajes a utilizar	39
1.8.1 Visual Paradigm.....	39
1.8.2 Lenguaje unificado de modelado (UML).....	40
1.9 Conclusiones.....	41
CAPÍTULO 2: CARACTERISTICAS DEL SISTEMA	42
2.1 Introducción.....	42
2.2 Requisitos de software propuestos por el proyecto SIGEP-Cu.....	42
2.2.1 Requisitos Funcionales Módulo Traslados.....	43
2.2.2 Requisitos Funcionales Módulo Conducción	44
2.3 Modelado del sistema	45
2.3.1 Definición de actores y casos de uso del sistema	45
2.3.2 Diagrama de Casos de Uso del Sistema del Módulo Traslados	48
2.3.3 Diagrama de Casos de Uso del Sistema del Módulo Conducción.....	49
2.3.4 Descripción de Casos de Uso del Sistema	50
2.4 Modelo conceptual	54
2.4.1 Propuesta del Modelo Conceptual del Módulo Traslados	55

2.4.2 Propuesta del Modelo Conceptual del Módulo Conducción	55
2.4.3 Descripción del Modelo Conceptual	55
2.4.3.1 Descripción del Modelo Conceptual del Módulo Traslados.....	56
2.4.3.2 Descripción del Modelo Conceptual del Módulo Conducción	56
2.5 Validación de los requisitos funcionales.....	57
2.6 Conclusiones.....	58
CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA.....	59
3.1 Introducción.....	59
3.2 Modelo del Análisis	59
3.2.1 Definición del modelo de paquetes.....	59
3.2.2 Definición del modelo de clases del análisis.....	61
3.2.3 Diagramas de interacción. Colaboración.....	62
3.3 Arquitectura del sistema.....	62
3.3.1 Decisiones arquitectónicas	64
3.4 Patrones a utilizar	64
3.4.1 Patrones de asignación de Responsabilidades GRASP	65
3.4.2 Patrones Estructurales.....	65
3.4.3 Patrón de acceso a datos (DAO)	66
3.5 Modelo del diseño.....	66
3.5.1 Diagrama de clases con estereotipos WEB.....	66
3.5.2 Diagrama de interacción del diseño. Secuencia.....	67
3.6 Métricas para el Modelo de Diseño	68

3.7 Conclusiones.....	74
CONCLUSIONES	75
REFERENCIAS BIBLIOGRÁFICAS.....	76
BIBLIOGRAFÍA	76
GLOSARIO DE TÉRMINOS	78

ÍNDICE DE TABLAS

TABLA 1 REQUISITOS FUNCIONALES MÓDULO CONDUCCIÓN	44
TABLA 2 REQUISITOS FUNCIONALES MÓDULO CONDUCCIÓN	45
TABLA 3 ACTORES DEL SISTEMA MÓDULO TRASLADOS	47
TABLA 4 ACTORES DEL SISTEMA MÓDULO CONDUCCIÓN	48
TABLA 5 ESPECIFICACIÓN DEL CASO DE USO REGISTRAR EJECUCIÓN DE CONDUCE	54

ÍNDICE DE FIGURAS

FIGURA 1 PATRÓN CONCORDANCIA. REUSABILIDAD	37
FIGURA 2 PATRÓN MÚLTIPLES ACTORES. ROLES COMUNES.....	37
FIGURA 3 DIAGRAMA DE CASOS DE USO DEL SISTEMA DEL MÓDULO TRASLADOS.....	48
FIGURA 4 DIAGRAMA DE CASOS DE USO DEL SISTEMA DEL MÓDULO CONDUCCIÓN	49
FIGURA 5 MODELO CONCEPTUAL DEL MÓDULO TRASLADOS	55
FIGURA 6 MODELO CONCEPTUAL DEL MÓDULO CONDUCCIÓN.....	55
FIGURA 7 PROTOTIPO ELEMENTAL DE INTERFAZ GRÁFICA DE USUARIO CU REGISTRAR EJECUCIÓN DE CONDUCE.....	57
FIGURA 8 DIAGRAMA DE PAQUETES RELACIONADOS CON EL MÓDULO TRASLADOS	60
FIGURA 9 DIAGRAMA DE PAQUETES RELACIONADOS CON EL MÓDULO CONDUCCIÓN	60
FIGURA 10 ESTEREOTIPOS DE LAS CLASES DEL ANÁLISIS.....	61
FIGURA 11 DIAGRAMA DE CLASES DEL ANÁLISIS CU REGISTRAR EJECUCIÓN DE CONDUCE	61
FIGURA 12 DIAGRAMA DE COLABORACIÓN CU REGISTRAR EJECUCIÓN DE CONDUCE	62
FIGURA 13 ARQUITECTURA PROPUESTA POR GRAILS.....	64
FIGURA 14 DIAGRAMA DE CLASES CON ESTEREOTIPOS WEB CU REGISTRAR EJECUCIÓN DE CONDUCE.....	67
FIGURA 15 DIAGRAMA DE INTERACCIÓN DEL DISEÑO CU REGISTRAR EJECUCIÓN DE CONDUCE	68
FIGURA 16 TAMAÑO OPERACIONAL DE CLASE MÓDULO TRASLADOS.....	68
FIGURA 17 TAMAÑO OPERACIONAL DE CLASE MÓDULO CONDUCCIÓN	69
FIGURA 18 REPRESENTACIÓN DE LA RESPONSABILIDAD DE LAS CLASES DEL MÓDULO TRASLADOS	69
FIGURA 19 REPRESENTACIÓN DE LA RESPONSABILIDAD DE LAS CLASES DEL MÓDULO CONDUCCIÓN	69
FIGURA 20 REPRESENTACIÓN DE LA REUTILIZACIÓN DE LAS CLASES DEL MÓDULO TRASLADOS.....	69
FIGURA 21 REPRESENTACIÓN DE LA REUTILIZACIÓN DE LAS CLASES DEL MÓDULO CONDUCCIÓN.....	70
FIGURA 22 REPRESENTACIÓN DE LA COMPLEJIDAD DE LAS CLASES DEL MÓDULO TRASLADOS.....	70
FIGURA 23 REPRESENTACIÓN DE LA COMPLEJIDAD DE LAS CLASES DEL MÓDULO CONDUCCIÓN.....	70
FIGURA 24 RELACIONES ENTRE CLASES MÓDULO TRASLADOS	71
FIGURA 25 RELACIONES ENTRE CLASES MÓDULO CONDUCCIÓN.....	71
FIGURA 26 ACOPLAMIENTO DE LAS CLASES DEL MÓDULO TRASLADOS.....	71
FIGURA 27 ACOPLAMIENTO DE LAS CLASES DEL MÓDULO CONDUCCIÓN	71
FIGURA 28 REUTILIZACIÓN DE LAS CLASES DEL MÓDULO TRASLADOS	72
FIGURA 29 REUTILIZACIÓN DE LAS CLASES DEL MÓDULO CONDUCCIÓN	72
FIGURA 30 COMPLEJIDAD DE MANTENIMIENTO PARA LAS CLASES DEL MÓDULO TRASLADOS	72
FIGURA 31 COMPLEJIDAD DE MANTENIMIENTO PARA LAS CLASES DEL MÓDULO CONDUCCIÓN.....	73
FIGURA 32 CANTIDAD DE PRUEBAS PARA LAS CLASES DEL MÓDULO TRASLADOS	73
FIGURA 33 CANTIDAD DE PRUEBAS PARA LAS CLASES DEL MÓDULO CONDUCCIÓN	73

INTRODUCCIÓN

A medida que la era de la información se consolida, en conjunto aumentan los sistemas informáticos, por lo cual la mayoría de los países tratan de transformar todos sus procesos manuales en automatizados o sea de informatizar sus sociedades. Cuba no ha constituido la excepción en este ámbito pues enfoca sus líneas investigativas a ello; hasta la actualidad en nuestro país a nivel social se han informatizado un sin número de ramas o sectores sociales, entre los que se encuentran: la salud, la educación, el deporte, el turismo, la economía, las Fuerzas Armadas Revolucionarias (FAR), el Ministerio del Interior (MININT) y dentro de este último el Sistema Penitenciario, en el cual se centrará la investigación de este trabajo.

El proceso de automatización del Sistema Penitenciario a nivel nacional comienza en el año 1989, con un sistema informático para el Control Penal. Este permitía la gestión de los principales datos del recluso y algunos aspectos del trabajo de la especialidad. A medida que se avanzaba en el tiempo se fueron dando nuevos pasos para la mejora de este sistema y en cumplimiento de la orden 43/99 del Viceministro Primero del Interior se diseña, programa e implanta el Sistema para el Control del Recluso (SACORE), el cual utilizaría la tecnología existente en ese entonces en el MININT. Dicha tecnología se considera de forma general ineficiente ya que las computadoras eran 486, sin conectividad interna ni externa, la programación se hacía en lenguajes que actualmente no se utilizan para estos tipos de sistemas y el Gestor de Base Datos utilizado en el aquel entonces era propietario.

En sus seis años de explotación el SACORE se ha ido perfeccionando a partir de los requerimientos y solicitudes de los usuarios, conteniendo como subsistemas principales Control Penal, Tratamiento Educativo y Orden Interior (los dos últimos con varias opciones y salidas pendientes). Una vez arrancada su explotación se determina la necesidad de adicionar o modificar opciones en algunos subsistemas como es el caso particular de Servicios Médicos. Además se desarrollaron otros dos sistemas automatizados, que complementan la información del SACORE: el sistema de incidencias y el de capacidades (SAIDEP y SACDEP) los cuales pertenecen actualmente al Centro Nacional y el SACORE se encuentra en todos los centros penitenciarios nacionales. Esta situación trae consigo que el sistema se encuentre constantemente en cambios, transformaciones o mantenimientos, para

la puesta en práctica de algunas de las opciones necesitadas, por ello surge la idea de la creación del Sistema de Gestión Penitenciario Cubano SIGEP-Cu, el cual se encarga de informatizar el Sistema Penitenciario Nacional, aprovechando el desarrollo de las tecnologías, para automatizar los procesos de trabajo que se ejecutan para el control, tratamiento y atención a los acusados, asegurados y sancionados en los establecimientos penitenciarios y los correspondientes a los tres niveles de mando, incluyendo las relaciones con otras Líneas del Organismo, Órganos de Justicia y otros de la Administración Central del Estado.

El subsistema de Registro Legal está compuesto por varios módulos entre ellos se encuentra el de Traslados y Conducción a los cuales estará dirigido el objetivo de la investigación.

Por esta razón se plantea:

Problema a resolver:

La carencia de un modelo de solución que permita automatizar los procesos de los módulos Traslados y Conducción en el proyecto SIGEP-Cu.

Objetivo General:

Realizar el modelado del análisis y del diseño de los módulos Traslados y Conducción del Sistema Penitenciario Cubano.

Objeto de Estudio:

Procesos de Registro Legal en el Sistema Penitenciario Cubano.

Campo de Acción:

Procesos de Traslados y Conducción del Sistema Penitenciario Cubano.

Tareas a cumplir:

1. Analizar y fundamentar las tendencias actuales en el modelado del negocio.
2. Analizar y fundamentar las tecnologías actuales en la Ingeniería de Requerimientos.
3. Caracterizar los procesos de Registro Legal en el Sistema Penitenciario Cubano.
4. Estructurar el modelo del sistema para los módulos Traslados y Conducción del SIGEP-Cu.
5. Validar los requisitos identificados, mediante la técnica de prototipo de interfaz.
6. Realizar el modelo del análisis de los módulos Traslados y Conducción del SIGEP-Cu.
8. Realizar el modelo del diseño de los módulos Traslados y Conducción del SIGEP-Cu.

Idea a defender

Si se realiza el análisis y el modelado del diseño de los módulos Traslados y Conducción del Sistema Penitenciario Cubano, se podría facilitar la posterior implementación de estos módulos en el proyecto SIGEP-Cu.

Este informe está estructurado en tres capítulos, como se describe a continuación:

Capítulo 1: Se enuncian los principales conceptos asociados al dominio del problema. Se realiza un estudio de algunas soluciones de software existentes relacionadas con los Sistemas Penitenciarios tanto a nivel nacional como internacional o sea el Estado del Arte y al finalizar se hace referencia a la metodología de desarrollo de software y herramientas seleccionadas para realizar este trabajo.

Capítulo 2: Se hace alusión a todo lo referente a las características del sistema, donde se plasman los requisitos definidos por el proyecto SIGEP-Cu y a los cuales se les realizan el modelado del sistema, posteriormente el modelo conceptual y el diccionario de datos correspondiente y por último el validado de los requisitos previamente identificados.

Capítulo 3: Se realiza la modelación del análisis, teniendo en cuenta los requisitos de Traslado y Conducción previamente definidos en el proyecto; además se lleva a cabo el diseño de la solución, obteniendo la arquitectura base, los patrones de diseño, uso y beneficios de los mismos; se alcanzará además la descripción de las principales clases utilizadas en la solución, dentro de las cuales se tendrá el diagrama de clases y las métricas para la validación del diseño correspondiente.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

1.1 Introducción

En el presente capítulo se realiza una revisión de los conceptos a los cuales se harán referencia en todo el contexto del trabajo, tal es el caso de los términos Sistema Penitenciario y Registro Legal. Se brinda información acerca de los procesos de Traslados y Conducción para los asegurados, acusados y sancionados en los establecimientos penitenciarios del país. Se hace referencia a las principales características de soluciones de software relacionadas con el Sistema Penitenciario a nivel nacional e internacional. Por último se describirán las herramientas, técnicas y metodologías a utilizarse en el desarrollo de la solución que se propone.

1.2 Sistemas Penitenciarios

Un Sistema es un conjunto de cosas que ordenadamente entre sí contribuyen a un fin. [1]

Al asignar al régimen penitenciario la calificación de Sistema Penitenciario se hace referencia al conjunto de normas, procedimientos y dependencias dispuestas por el Estado para la ejecución del régimen penitenciario entre los que se encuentran además los principios, programas, recursos humanos, dependencias e infraestructura que se encuentran relacionadas y destinadas a este régimen.

Manuel Osorio, creador del diccionario de Ciencias Jurídicas, Políticas y Sociales, asocia el Sistema Penitenciario con régimen penitenciario, definiendo éste régimen como: "Conjunto de normas legislativas o administrativas encaminadas a determinar los diferentes sistemas adoptados para que los penados cumplan sus condenas. Se encamina a obtener la mayor eficacia en la custodia o en la readaptación social de los delincuentes. Esos regímenes son múltiples, varían a través de los tiempos y van desde el aislamiento absoluto y de tratamiento rígido, hasta el sistema de puerta abierta con libertad vigilada". El Sistema Penitenciario, es el conjunto de normas generales, establecidas y específicas referidas a las penas en sí, el modo de su cumplimiento y el tratamiento de los penados y procesados. [2]

En el caso de la República de Cuba, tal sistema está constituido por la legislación vigente, los métodos que se emplean para lograr su funcionamiento, las diferentes dependencias encargadas de su aplicación, los equipos de trabajo y la infraestructura carcelaria.

El Reglamento del Sistema Penitenciario de Cuba establece que, el Sistema Penitenciario Cubano es el encargado de garantizar el proceso de ejecución de la sanción de privación de libertad, de la sanción de trabajo correccional con internamiento, la medida de seguridad reeducativa de internamiento y la medida cautelar de prisión provisional.

Este sistema, dirigido por la Dirección de Establecimientos Penitenciarios del Ministerio del Interior, se sustenta en la integración de principios, conceptos, procedimientos, fuerzas y medios que garantizan el funcionamiento de los centros destinados al internamiento y el tratamiento a los internos.

La Dirección de Establecimientos Penitenciarios, conforme con lo establecido por la Constitución y demás leyes, las resoluciones judiciales y el presente Reglamento, es la encargada de la ejecución y el logro de los objetivos implícitos en las sanciones y medidas privativas de libertad. Para garantizar su cumplimiento tiene las funciones generales siguientes:

- a) Elaborar y proponer la implementación del Sistema Penitenciario, supervisando y evaluando los resultados de su aplicación.
- b) Exigir y controlar el estricto cumplimiento de la legalidad socialista en la ejecución de las sanciones y medidas privativas de libertad.
- c) Establecer, instrumentar y evaluar la aplicación de los programas educativos con el objetivo de promover en los internos el desarrollo de una cultura general integral, preparándolos para su reinserción a la sociedad.
- d) Establecer, coordinar, promover y exigir conforme con la ley, la participación e intervención de otros órganos del MININT, Organismos de la Administración Central del Estado, entidades estatales y judiciales y las organizaciones sociales y de masas, para garantizar la ejecución de la sanción y el desarrollo de los programas educativos.
- e) Organizar la ubicación, designación y funcionamiento de los centros penitenciarios y demás lugares de internamiento.

- f) Disponer de las capacidades instaladas en el país para lograr el equilibrio entre éstas y la cifra de internos de cada provincia.
- g) Ubicar a los internos en su provincia de residencia, salvo cuando sea indispensable ubicarlos en otra provincia para garantizar la continuidad de los procesos judiciales o para asegurar adecuadas condiciones de vida y trabajo.
- h) Realizar controles, inspecciones y auditorias en los centros penitenciarios y otros lugares de internamiento. [3]

1.2.1 Registro Legal

Es el proceso del Sistema Penitenciario encargado de garantizar el cumplimiento de la legalidad, la organización, control, registro y tramitación de la documentación legal del procesado o penado durante el ingreso, reingreso, permanencia y egreso de los lugares de internamiento.[4]

1.3 Subsistema Registro Legal del SIGEP-Cu

El Sistema de Gestión Penitenciaria Cubano (SIGEP-Cu) estará constituido por un conjunto de sistemas y subsistemas estructurados según las áreas que conforman la actividad penitenciaria. Dentro de los subsistemas que conforman el SIGEP-Cu, se encuentra Registro Legal. El mismo está dirigido a garantizar el cumplimiento de la legalidad, la organización, control, registro y tramitación de la documentación legal de los procesados y penados, durante el cumplimiento de las medidas privativas o restrictivas de libertad, sus objetivos fundamentales son la coordinación, planificación y ejecución de los ingresos, egresos y traslados de los individuos sometidos a una medida privativa o restrictiva de libertad. Tiene como alcance la gestión del ingreso, reingreso, egreso y traslado de los individuos, el seguimiento al debido proceso, el suministro de información legal y estadística a los principales clientes internos y externos.[5]

Dentro del subsistema de Registro Legal se desarrollan los módulos siguientes:

- **Solicitudes:** Las solicitudes son peticiones que realizan los internos, sus familiares y los establecimientos penitenciarios a los tribunales o a la Dirección de Establecimientos Penitenciarios

(DEP). Estas pueden ser: de otorgamiento de beneficios, traslado interpenal, conducciones, entre otras. Este módulo permite registrar y controlar estas solicitudes y los documentos que las avalan.

- **Decisiones:** Las decisiones son las acciones ordenadas por las instancias competentes (Autoridades judiciales competentes, los órganos provinciales o la DEP). Las decisiones se comunican a las jefaturas provinciales o centros penitenciarios a través de documentos de carácter legal. Estas determinan la ejecución obligatoria de procesos en los establecimientos penitenciarios. Este módulo permite registrar y controlar las decisiones y los documentos que las avalan.
- **Situación Legal:** Permite gestionar toda la información relativa a la situación legal de los privados de libertad. En la situación legal del interno debe recogerse los datos de las causas por las que cumple el interno, los regímenes por lo que progresa, los datos de los delitos cometidos, las sanciones accesorias, la responsabilidad civil, los trámites legales pendientes, el historial de conducciones y traslados del interno y cualquier otra información que permita establecer y controlar su situación legal. Debe posibilitar, además, evidenciar todas las conversiones en la situación legal del interno que se hayan evidenciado en su tránsito por el sistema penitenciario.
- **Control de Capacidades:** Este módulo permite definir la estructura física del área de reclusión Legal y mantener un control sobre las capacidades penitenciarias, los niveles de ocupación, la disponibilidad y las afectaciones. Permite registrar las capacidades asignadas por la DEP y la jefatura provincial, de modo que se conozca con antelación los nuevos ingresos que recibirá el centro penitenciario. Cuando un interno es trasladado a un centro penitenciario se verifica que tenga capacidad asignada y que la documentación que avala el ingreso del interno está correcta. Si se cumplen los requisitos establecidos se le permite la entrada al interno y se registra. Este paso es requerido para que se pueda ejecutar el ingreso, pues este solo actúa sobre los internos a los que se le ha registrado la entrada al centro penitenciario.
- **Designación:** Este módulo permite, a partir de criterios establecidos y fundamentados jurídicamente para compartimentar a los internos, que cada Jefatura autorizada establezca la designación de los establecimientos penitenciarios a su alcance para el cumplimiento de la sanción de privación de libertad, medida cautelar de prisión provisional y medida reeducativa de internamiento.

- **Egreso:** El objetivo de este módulo es registrar el egreso de los internos de un centro penitenciario. El egreso sólo puede registrarse si existen decisiones judiciales que los avalan, excepto en el caso de bajas definidas por el sistema penitenciario. En este último, para registrar el egreso definitivo deben registrarse los documentos que confirman la defunción del interno. El sistema es capaz de identificar a los internos que les corresponde egreso en una fecha determinada.
 - **Conducción:** Este módulo permite gestionar la planificación de las conducciones. Aquí se agrupan los traslados a hospital, a tribunal y las funerarias y domicilio especificadas en la ley como trámites. Además el módulo controla la ejecución de las salidas planificadas y el retorno de los internos.
 - **Traslados:** Este módulo gestiona la planificación y ejecución de los traslados inter-unidades o inter-provinciales. Para la planificación se toma como base las decisiones de traslado inter-penal registradas en el área de Registro Legal. Los traslados pueden ser individual o colectivos, este último para el caso en que las decisiones de traslado de los internos involucrados, coinciden en fecha y destino, pudiéndose planificar como una única salida. De igual manera el módulo controlará la ejecución de los traslados, registrando la salida, las incidencias en el trayecto y la llegada al destino.
- De los módulos antes enunciados que integran el subsistema Registro Legal, se centrará la investigación en los de Traslados y Conducción.

1.3.1 Módulo Traslados

El proceso de traslado consiste en el movimiento de un interno o grupo de ellos, de un lugar de internamiento a otro. El traslado se promueve a solicitud del centro penitenciario a través del Órgano Provincial y del Municipio Especial Isla de la Juventud (MEIJ) de Establecimientos Penitenciarios; por solicitud del interno o sus familiares y por indicaciones de la Jefatura de la Dirección de Establecimientos Penitenciarios (DEP).

1.3.1.1 Causas para efectuar el proceso de traslado de reclusos

Las causas que se tienen en cuenta para efectuar el proceso de traslado de un recluso de una unidad penitenciaria a otra son las siguientes:

1. Cuando lo requiera el estado de salud del interno.

2. Por intereses operativos del Sistema Penitenciario y de los Órganos Operativos.
3. Por motivos de orden interno en los lugares de internamiento.
4. Para descongestionar los lugares de internamiento.
5. Para incrementar las condiciones de seguridad del interno.
6. Por promoción o regresión de régimen o por decidirse la ubicación del interno en condiciones de mínima seguridad.
7. Por captura en territorio distinto desde el cual se evadió el interno.
8. Por residir en provincia distinta a donde extingue.
9. Por ser portador de VIH-Sida.

1.3.1.2 Traslados interprovinciales

Los traslados interprovinciales, se solicitan al Departamento de Control Legal de la DEP mediante el correo electrónico, especificando número de físico del interno, nombres y apellidos, situación legal, clasificación y motivo que origina la solicitud.

1.3.1.3 Traslados interpenales

Es el traslado de un interno de un centro penitenciario a otro, previamente amparado por una orden judicial de traslado. La cual es enviada por el tribunal al establecimiento penitenciario donde se encuentra recluido el interno a trasladar.

1.3.1.4 Traslado Expreso

Los traslados ordenados por el Departamento de Control Legal de la DEP, se pueden cumplimentar según la clasificación de expreso y estos se efectúan en la fecha y hora que se establezca en la orden.

1.3.1.5 Traslado Ordinario

Los traslados ordenados por el Departamento de Control Legal de la DEP, se pueden cumplimentar según la clasificación de ordinario y se realizan dentro de los veinte (20) días a partir de su recibo. Se

utiliza esta clasificación para la reubicación de internos en virtud de lo ordenado por las autoridades facultadas y según lo dispuesto por el Tribunal Supremo Popular (TSP).

1.3.2 Módulo Conducción

El proceso de conducción es la acción de presentar bajo custodia a un interno en determinado lugar donde su permanencia es temporal y es custodiado todo el tiempo por los educadores auxiliares.

1.3.2.1 Causas para efectuar el proceso de conducción de reclusos

Las causas que se tienen en cuenta para efectuar el proceso de conducción de un recluso a continuación se enuncian:

1. La autoridad competente de los órganos de Instrucción Legal, de la Técnica Investigativa o Estación de Policía para la práctica de acciones investigativas y de instrucción, u otras diligencias. Los funcionarios de estos órganos garantizan su custodia y control.
2. Al Tribunal para la celebración del juicio oral o notificación de alguna resolución judicial.
3. Consulta o ingreso a hospitales.
4. La funeraria, domicilio u hospital, según corresponda. La ejecución de este movimiento se ajusta a las regulaciones que se establecen en los procedimientos de trabajo para el orden interior.
5. Actos de carácter civil u otros previstos en las leyes.
6. Actividades educativas previstas en el Reglamento y los procedimientos de trabajo del sistema educativo.
7. Al domicilio con fines educativos y de tratamiento a la familia del interno, cuando circunstancias especiales así lo aconsejen, de acuerdo con lo regulado en los procedimientos de trabajo del sistema educativo.

1.4 Estado del arte de los Sistemas Penitenciarios.

La investigación realizada a los sistemas automatizados existentes, mostró que dichos sistemas tienen en común la posibilidad de identificar a cada uno de los individuos que ingresan al Sistema Penitenciario, registrando sus datos personales, rasgos físicos y señas particulares, entre otros datos

de interés. El análisis de cada uno de los sistemas que a continuación se mencionarán, parte de las funcionalidades que brindan.

1.4.1 Soluciones nacionales

El Sistema Automatizado para el Control del Recluso (SACORE) [6], surge para dar cumplimiento a la Orden 43/99 del Vice-Ministro Primero del Ministerio del Interior de Cuba y tiene las siguientes características:

- Garantiza respuestas inmediatas a las solicitudes de información de los diferentes órganos e instituciones del estado como son: Jefatura del MININT, Ministerio de Justicia, Tribunales, Fiscalías, MINED, INDER, FMC, MINFAR.
- Recoge prácticamente la totalidad de la información de los reclusos en todas las especialidades.
- Tiene más de 200 reportes impresos.
- Permite la recuperación dinámica a partir de una solicitud de búsqueda.
- Los partes que se emiten son obtenidos de forma automatizada.
- Permite el traslado automático de todos los datos del recluso al nivel nacional.

Como se mencionó anteriormente el Sistema de Control de Reclusos ha sido sometido a varias transformaciones en sus subsistemas debido a solicitudes de los usuarios. Aunque se ha logrado que el Sistema Informativo existente en los tres niveles de mando sea el resultado del sistema automatizado implantado, aún no están totalmente informatizados y organizados, entre otros, los procesos priorizados de:

- Ingreso, evaluación y diagnóstico a los penales.
- Aprobación, ejecución y control de los traslados de Unidades.
- Digitalización del Expediente carcelario y demás documentos originados durante el ingreso y tránsito del interno por el Sistema Penitenciario.
- Control del cumplimiento y los resultados de la política penitenciaria en la educación del interno y funcionamiento del sistema de colectivo.
- Convenios y control de la política laboral de la población penal y su seguridad social, así como del pago de sus haberes y descuentos de sus adeudos.

- Evaluación de la trayectoria criminal y conductual de los internos, caracterizaciones, evaluaciones y estudios criminológicos y demás actividades de los Grupos Multidisciplinarios y Trabajadores Sociales.
- Organización y control de las actividades y movimientos de la población penal previstas en el orden del día.
- Control del estado y efectividad de los sistemas de seguridad.
- Organización de los sistemas de trabajo sobre la base de la disponibilidad de fuerzas y medios de enfrentamiento y las prioridades y características de cada puesto de trabajo a cubrir.
- Funcionamiento como sistema de la Red de Centros de Dirección y Puestos de Mando del Sistema Penitenciario Nacional (SPN) y sus flujos de información.
- Administración centralizada de las capacidades de los Establecimientos Penitenciarios sobre la base de su destino y clasificación de la población Legal del país y estado de las fuerzas y medios del SPN, así como de su aseguramiento.
- Dirección sobre la base del comportamiento de la situación operativa en los tres niveles de mando.
- Gestiones legales con los órganos de Justicia y Sistema de Policía.

Además el sistema de enseñanza del MININT no se ha desarrollado con técnicas modernas de la informática, por lo que la preparación del personal que se selecciona e ingresa al Sistema Penitenciario, no se corresponde con la que se requiere para realizar los procesos que se están proyectando. Por otra parte, la introducción de nuevas tecnologías no han generado todas las transformaciones funcionales y organizativas en los sistemas de trabajo y aún no es óptimo el aprovechamiento de éstas. Aún no se garantizan los niveles necesarios de seguridad de la información.

1.4.2 Soluciones extranjeras

A nivel internacional se han visto algunas soluciones que de una u otra manera han implementado en alguno de sus módulos servicios sobre el tema tratado. Por su parte el Sistema de Gestión Penitenciario Venezolano (SIGEP-Ve), que es el que mas se acerca a dar solución al problema planteado se desarrolló a partir del año 2006 por profesionales cubanos en la Universidad de las Ciencias Informáticas (UCI). Este sistema da respuesta a las necesidades de gestión, información y

apoyo a la toma de decisiones de la Dirección General de Custodia y Rehabilitación del Recluso (DGCR), en sus tres niveles, que se describen a continuación:

- **Operativo:** Integrado por los Establecimientos penitenciarios (Internados Judiciales y Centros Penitenciarios), Centros de Tratamiento Comunitario (CTC) y Unidades Técnicas de Apoyo al Sistema Penitenciario (UTASP).
- **Táctico:** Integrado por las Coordinaciones Regionales.
- **Estratégico:** Integrado por la Dirección General de Custodia y Rehabilitación del Recluso.

Con la puesta en marcha de este sistema en la República Bolivariana de Venezuela se logra:

- Aumentar la eficacia, profesionalismo y equidad en el sistema penitenciario venezolano, con el objetivo de lograr un incremento de la confianza en el sistema penitenciario en general.
- Generar y diseminar información vital para el funcionamiento de los establecimientos penitenciarios. Esto permite generar estadísticas confiables y actualizadas sobre la situación jurídica de los privados de libertad, condiciones de vida y salud, actividades de rehabilitación y reinserción, la situación operativa, la actividad administrativa, entre otras.
- Permitir la comunicación en línea con tribunales, sistemas de identificación y antecedentes Legales, que complementan la información necesaria para la gestión de los procesos vinculados a los privados de libertad.

Para alcanzar lo antes planteado, se establece como objetivo general desarrollar e implantar un sistema informático que soporte las decisiones estratégicas del Ministerio del Interior y Justicia y de la Dirección General de Custodia y Rehabilitación del Recluso, que contribuya a garantizar el respeto a los derechos de los internos, su actividad de rehabilitación y reinserción en la sociedad.

El Sistema de Gestión Penitenciario Venezolano (SIGEP-Ve) permite a los establecimientos penitenciarios (Internados Judiciales, Centros Penitenciarios, Centros de Tratamiento Comunitario) y otras sedes (Coordinaciones Regionales y UTASP), recopilar y controlar la información operativa que se genera en este tipo de centros. Esta solución controla el tránsito de los privados de libertad por todos los componentes del sistema penitenciario, y en general maneja datos sobre la población Legal y audita los procesos legales para garantizar un cumplimiento justo de la sentencia.

La solución incluye además la gestión de los servicios médicos y alimenticios y el tratamiento ofrecido en estos establecimientos penitenciarios, así como el apoyo para la toma de decisiones estratégicas de la DGCR.

La solución está diseñada para ofrecer al personal, servicios en línea que faciliten la ejecución de sus tareas. Contar con una base de datos centralizada que soporte toda la información de la institución, permitir a la dirección estratégica poseer una visión global del sistema y por tanto tomar decisiones en base a una información íntegra y precisa, mejorando el desempeño operacional de la Dirección General. El sistema se beneficia además con los enlaces a sistemas de otras instituciones (TSJ, ONIDEX y Ministerio Público) agilizando y asegurando el intercambio de información entre estos y los establecimientos penitenciarios.

Los sistemas ofrecen una solución escalable capaz de adaptarse a los requerimientos de los distintos tipos de establecimientos penitenciarios y de cualquier tamaño.

Los sistemas poseen una completa integración entre los diferentes módulos, además de una interfaz gráfica común que facilita su comprensión y uso.

Sin embargo la principal desventaja del sistema SIGEP-Ve es que sólo da respuesta a las necesidades para la toma de decisiones de la DGCR, motivo por el cual no puede ser utilizado en los establecimientos penitenciarios cubanos.

1.5 Surgimiento del Sistema de Gestión Penitenciario Cubano

SIGEP-Cu

Por todo lo antes expuesto, se decide a partir del año 2009 crear un sistema capaz de informatizar el Sistema Penitenciario Nacional en conmemoración al 50 aniversario del MININT. Con el objetivo de aprovechar el desarrollo de las tecnologías de la informática y las comunicaciones que se llevan a cabo en el país. Y de esta forma lograr automatizar los procesos de trabajo que se ejecutan para el control, tratamiento y atención a los acusados, asegurados y sancionados en los centros penitenciarios y los correspondientes a los tres niveles de mando, incluyendo las relaciones con otros Organismos de la Administración Central del Estado.

Tiene como objetivos específicos:

1. Garantizar el proceso de ingreso de todos los acusados, asegurados y sancionados remitidos por los órganos instrucción, fiscalías y tribunales al sistema penitenciario.
2. Garantizar la correcta evaluación y clasificación en régimen de los internos.
3. Lograr el control de los procesos jurídicos, trámites legales, otorgamiento de beneficios y seguridad de los internos en los centros penitenciarios.
4. Informatizar el funcionamiento de los consejos de dirección que se realizan en los centros penitenciarios.
5. Proveer a los especialistas de tratamiento educativo de una herramienta que les permita realizar un trabajo profundo y detallado en la preparación para la reinserción social de los internos: incorporación al estudio, trabajo, actividades culturales, políticas, recreativas, etc.
6. Lograr una adecuada gestión de los trámites legales de los internos mediante la integración con los diferentes órganos de justicia.
7. Garantizar información operativa y oportuna para la toma de decisiones estratégicas a los tres niveles de mando.
8. Mantener identificado al interno en su tránsito por el sistema penitenciario.
9. Conocer la cantidad de acusados, asegurados y sancionados, así como su ubicación y situación legal.
10. Registrar las personas vinculadas al interno y los tipos de influencia que ejercen sobre el mismo.
11. Controlar la aplicación de medidas disciplinarias y suspensión de beneficios.
12. Controlar las requisas, ocupaciones y hallazgos a los internos en los centros penitenciarios.
13. Realizar la planificación de visitas familiares, institucionales y pabellones, así como realizar el control de la ejecución de las mismas.
14. Administrar las capacidades, conocer las disponibilidades y afectaciones de la infraestructura física de las instalaciones.
15. Controlar las incidencias que ocurren en los establecimientos penitenciarios.
16. Elaborar partes informativos para los tres niveles de mando.
17. Controlar el estado de salud del interno.
18. Controlar las consultas, chequeos médicos, planes de vacunación y análisis clínicos (programados y no programados) y control de enfermedades.

1.6 Acerca de los requerimientos

¿Qué es un requerimiento?

1. La IEEE Standard Glossary of Software Engineering Terminology define un requerimiento como condición o capacidad que necesita un usuario para resolver un problema o lograr un objetivo.
2. Condición o capacidad que tiene que ser alcanzada o poseída por un sistema o componente de un sistema para satisfacer un contrato, estándar u otro documento impuesto formalmente.
3. Una representación documentada de una condición o capacidad, como se expresa en el enunciado 1 o en el 2.

Para una mejor comprensión de lo que realmente debe hacer un software se definió lo que es la Ingeniería de Requisitos, quien se encargará de la identificación, análisis y especificación de los requisitos o requerimientos. Los requisitos reciben varias clasificaciones pero una de las principales es por lo que hacen en el sistema resultando en:

- Funcionales: Describen los servicios o funciones que se esperan del sistema.
- No funcionales: Describen las cualidades o propiedades que debe tener el sistema.

1.6.1 Objetivos de la captura de requisitos

Los objetivos de la captura de requisitos son:

1. Definir el ámbito del sistema.
2. Definir una interfaz de usuarios para el sistema, enfocada a las necesidades y metas del usuario.
3. Establecer y mantener un acuerdo entre clientes y otros involucrados sobre lo que el sistema debería hacer.
4. Proveer a los desarrolladores un mejor entendimiento de los requisitos del sistema.
5. Proveer una base para estimar recursos y tiempo de desarrollo del sistema.
6. Proveer una base para la planeación de los contenidos técnicos de las iteraciones.

Lograr una comunicación efectiva entre los usuarios y el equipo de proyecto con el objetivo de llegar a un entendimiento de lo que hay que hacer, es la clave del éxito en la producción de un software.

Las áreas de esfuerzo del análisis de requerimientos están en:

- Reconocimiento del problema como lo ve el usuario.

- Evaluación del problema y síntesis de la evaluación: Como evaluación se entiende la definición de los datos observables, la evaluación del flujo y contenido de la información. Toda esta evaluación se sintetiza en la definición en detalle de los datos, funciones y el comportamiento del sistema.
- Modelado: Creación de modelos que ayuden a entender la entidad a construir.
- Construcción de un prototipo de alto nivel del sistema.
- Revisión por parte del usuario.
- Firma del contrato si las partes están de acuerdo.

1.6.2 Especificación de los requerimientos del sistema.

Una especificación de los requerimientos del sistema tradicionalmente ha sido vista como un documento que comunica los requerimientos del cliente a la comunidad técnica que construirán el sistema. La colección de requerimientos que constituyen la especificación y su representación actúan como el puente entre los dos grupos y debe ser entendible tanto por el cliente como por la comunidad técnica. Una de las tareas más difíciles en la creación de un sistema, es aquella de comunicar a todos los subgrupos, especialmente en un solo documento. Este tipo de comunicación usualmente requiere diferentes formalismos y lenguajes.

1.6.3 Características que debe poseer una Especificación de Requisitos de Software (ERS)

Una ERS debe ser, según la norma IEEE 830 – 1998 (IEEE '98b) [10]:

Correcta: La ERS es correcta sí, y sólo sí, contiene todos los requisitos que el software debe satisfacer. No hay una herramienta o procedimiento que garantice esta característica, sin embargo, limitarse a obtener de una vez lo que “quiere” el cliente pone en riesgo el logro de esta característica. Realizar un proceso iterativo, donde se involucren los interesados, seguir la traza de los requisitos y validarlos con los interesados contribuyen a lograr una ERS correcta.

No ambigua: La ERS es no ambigua sí, y sólo sí, cada requisito tiene una única interpretación. Para elaborar una ERS no ambigua es necesario mantener y documentar un acuerdo entre el equipo de desarrollo y los interesados respecto a los requisitos. Cuando se usa el lenguaje natural para

documentar los requisitos pueden introducirse ambigüedades, sin embargo, este es fácilmente comprendido por los interesados, a diferencia de los lenguajes formales, que requieren conocimientos específicos pero permiten reducir la ambigüedad. Anteriormente se sugirió usar una combinación de representaciones a lo cual habría que añadir que si en dicha combinación se hace uso del lenguaje natural debe mantenerse un glosario de términos para disminuir la ambigüedad.

Completa: La ERS es completa sí, y sólo sí, incluye los siguientes elementos:

- Todos los requisitos funcionales y no funcionales son conocidos y documentados en la ERS.
- Están definidas todas las responsabilidades del sistema respecto a los datos de entrada, tanto válidos como no válidos y respecto a los datos de salida.
- En los documentos de la ERS todas las figuras, tablas, diagramas y definiciones de términos están nombrados y referenciados.

Desarrollar de forma iterativa la ERS de software, validarla y usar múltiples representaciones contribuyen a lograr la completitud.

Consistente: La ERS es consistente sí, y sólo sí, ningún subconjunto de la misma entra en contradicción con otro subconjunto. Las revisiones técnicas y el uso de diferentes representaciones para los requisitos contribuyen a reconocer la inconsistencia.

Clasificada por importancia: La ERS cumple con esta característica si cada requisito puede ser identificado de manera única y tiene un atributo que indica su importancia desde el punto de vista de los interesados. Todos los requisitos no son igualmente importantes, algunos son críticos para el sistema y otros son deseables, conocer estos atributos es útil para planificar las iteraciones.

Comprobable: La ERS es comprobable sí, y sólo sí, se puede comprobar por una persona o máquina mediante un procedimiento finito y cuya relación costo-beneficio sea aceptable que cada requisito está en el sistema desarrollado.

Modificable: La ERS es modificable sí, y sólo sí, puede ser modificada fácilmente manteniendo la completitud y consistencia. Mantener la traza de los requisitos, organizarlos adecuadamente y usar referencias cruzadas contribuye a hacerla modificable.

Posible de rastrear: La ERS es posible de rastrear si está claro el origen de cada requisito y es posible determinar los elementos relacionados en etapas posteriores del desarrollo.

1.6.4 Técnicas para identificar requerimientos

Los requisitos una vez definidos necesitan ser validados. La validación de requisitos tiene como misión demostrar que la definición de los requisitos define realmente el sistema que el usuario necesita o el cliente desea. Es necesario asegurar que el análisis realizado y los resultados obtenidos de la etapa de definición de los requisitos son correctos. Pocas son las propuestas existentes que ofrecen técnicas para la realización de la validación y muchas de ellas consisten en revisar los modelos obtenidos en la definición de requisitos con el usuario para detectar errores o inconsistencias.

Aún así, existen algunas técnicas que pueden aplicarse para ello:

Desarrollo Conjunto de Aplicaciones (Joint Application Development): Es una técnica que consiste en realizar sesiones conjuntas entre los analistas de sistemas y los expertos del dominio.

Desarrollo de prototipos: Los prototipos son versiones reducidas, demos o conjunto de pantallas (que no son totalmente operativos) de la aplicación pedida. Esta técnica se resume en: “No sé exactamente lo que quiero, pero lo sabré cuando lo vea”.

ETHICS (Implementación Efectiva de Sistemas Informáticos desde los puntos de vista Humano y Técnico): Se enfoca en los aspectos sociales que están presentes en el desarrollo del software, dado que un sistema no tendrá éxito sino es utilizado eficientemente por los empleados.

Puntos de vista: Todos los sistemas ocupan de un grupo de usuarios interesados (stakeholders), cada uno puede tener intereses diferentes, incluso en muchos casos contradictorios. Existen métodos que toman los puntos de vistas de los usuarios para encontrar cosas en común.

Según Ian Sommerville [9] las técnicas más utilizadas para identificar los requisitos de un software son:

Escenarios: Consiste en realizar una descripción de lo que los usuarios esperan que haga el sistema. Esta información se obtiene en conversaciones del analista con el cliente.

Talleres estructurados: Esta técnica es muy eficaz pues participa un gran número de personas, generalmente él o los analistas y los usuarios finales que utilizarán el software. Cada miembro participante opina y expone sus criterios respecto a lo que considera como posible funcionalidad del futuro sistema.

Etnografía: Es una técnica de observación que se puede utilizar para entender los requerimientos sociales y organizacionales. El valor de esta técnica es que ayuda a los analistas a descubrir los requerimientos implícitos que reflejan los procesos reales más que los formales en los que está involucrada. No es efectivo utilizar esta técnica para descubrir requerimientos del dominio. [9] Otros autores opinan que también se pueden utilizar técnicas más tradicionales como la revisión de la documentación técnica, las entrevistas y la Ingeniería inversa.

Ingeniería inversa: Se basa en la observación y el análisis detallado de otros sistemas ya existentes con características similares al que se quiere desarrollar. Es necesario aclarar que aunque muchas de las futuras funcionalidades del software que queremos desarrollar se pueden obtener de este análisis no se debe realizar una copia exacta del sistema que se esté analizando.

1.6.5 Técnicas de validación de requisitos

La técnica de validación de requisitos utilizada en el proyecto SIGEP-Cu es por Prototipo de interfaz de usuario, algunas de sus características y objetivos se describen a continuación:

Prototipos de interfaz de usuario: Algunas propuestas se basan en obtener de la definición de requisitos prototipos que, sin tener la totalidad de la funcionalidad del sistema, permitan al usuario hacerse una idea de la estructura de la interfaz del sistema.

Los prototipos de interfaz de usuarios son la presentación de la interfaz del producto que representa la funcionalidad; de manera que permita que el usuario verifique que el sistema va a satisfacer sus necesidades. Un prototipo es una versión inicial de un sistema de software que se utiliza para demostrar los conceptos, probar las opciones de diseño y entender mejor el problema y su solución. Un prototipo puede revelar errores u omisiones en los requerimientos propuestos, favorece la comunicación entre clientes y desarrolladores, da una primera visión del producto.

Otras técnicas que se usan para validar requisitos:

Listas de chequeo: Son frecuentemente usadas en inspecciones o revisiones de artefactos generados en el proceso de producción de software; son listas de aspectos que deben ser completados o verificados.

Auditorías: Revisar la documentación con esta técnica, consiste en un chequeo de los resultados contra una checklist (Listas de Chequeo) predefinida o definida a comienzos del proceso, es decir sólo una muestra es revisada.

Casos de Prueba: El objetivo del proceso de diseño de casos de prueba es crear un conjunto de casos de prueba que sean efectivos descubriendo defectos en los programas y muestren que el sistema satisface sus requerimientos.

1.6.6 Aplicación de patrones de casos de uso

El modelado de casos de uso no difiere de otros tipos de desarrollo en cuanto a que se reutilicen soluciones que se consideren eficientes. De ahí que las exitosas técnicas y diseños que han sido utilizados una y otra vez en los modelos de casos de uso, se formalizan como patrones expresando buenos diseños de modelos de casos de uso

La práctica en el uso de casos de uso ha avanzado en un conjunto de patrones que posibilitan con más exactitud reflejar los requisitos reales, por lo que el trabajo con los sistemas lo hace más fácil, y mucho más simple su mantenimiento. Dado un contexto y un problema a resolver, estas técnicas han mostrado ser la solución adoptada en la comunidad del desarrollo de software. Se presentan a modo de herramientas que permiten resolver los problemas que se les planteen a los desarrolladores de una forma ágil y sistemática. Estos patrones se enfocan hacia el diseño y las técnicas utilizadas en modelos de alta calidad, y no en cómo modelar usos específicos.

A continuación se presenta una descripción de los patrones de casos de uso aplicados en la definición del modelo de casos de uso para los módulos Traslado y Conducción.

Preciso y Legible

Este patrón plantea que las descripciones de los casos de uso sean lo suficientemente legibles como para que los clientes comprendan hasta donde se están describiendo las funcionalidades del sistema a construir. Además le permitirá al usuario evaluar y precisar cada caso de uso con el fin de definir que es lo que está listo para comenzar a implementar.

Concordancia

Extrae una subsecuencia de acciones que aparecen en diferentes lugares del flujo de casos de uso y es expresado por separado.

- Reusabilidad

Consta de 3 casos de uso. El primero llamado subsecuencia común, modela una secuencia de acciones que aparecerán en múltiples casos de uso en el modelo. Los otros casos de uso modelan el uso del sistema que comparte la subsecuencia común de acciones. De manera que deben existir al menos dos de ellos.

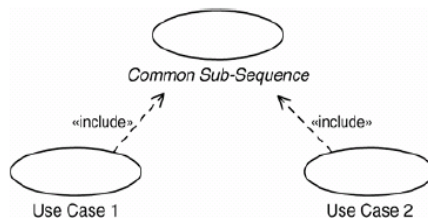


Figura 1 Patrón Concordancia. Reusabilidad

Múltiples actores

- Roles comunes

Puede suceder que los dos actores jueguen el mismo rol sobre el CU. Este rol es representado por otro actor, heredado por los actores que comparten este rol. Es aplicable cuando, desde el punto de vista del caso de uso, solo exista una entidad externa interactuando con cada una de las instancias del caso de uso.

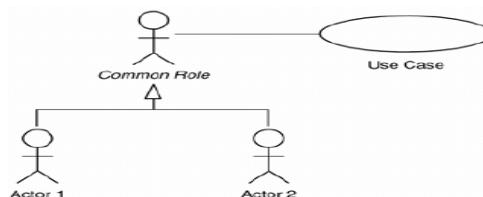


Figura 2 Patrón Múltiples actores. Roles comunes

Escenario más Fragmentos

Cuando se esta describiendo un caso de uso, se deben escribir los eventos del flujo principal como un escenario simple sin considerar posibles fallos quedando claramente identificado. Seguido de esto se pondrán los flujos que muestran condiciones alternativas que podrán ocurrir.

El nombre revela la intención

La utilización de este patrón permite tener los casos de uso nombrados correctamente, ya que propone crear los nombres de los casos de uso con un verbo que identifique la función que se realiza o una frase que represente la meta del actor primario.

1.7 Metodología de desarrollo de software a utilizar: Rational Unified Process (RUP)

El desarrollo de software nunca ha sido una tarea fácil, una prueba de ello es la gran cantidad de propuestas metodológicas para llevar a cabo un proyecto de software. Estas propuestas se encuentran divididas en dos grupos fundamentales: [11]

- **Las metodologías tradicionales o pesadas**, las cuales se centran fundamentalmente en el control del proceso, estableciendo rigurosamente las actividades involucradas, los artefactos que se deben producir, y las herramientas y notaciones que se usarán.
- **Las metodologías ágiles**, las cuales se centran fundamentalmente en el factor humano y en el producto de software, dando mayor valor al individuo, a la colaboración con el cliente y al desarrollo incremental del software con iteraciones muy cortas.

Teniendo en cuenta que el proyecto que se desarrollará tiene una complejidad alta, que el equipo de trabajo supera a las 60 personas y que no cuenta con una amplia experiencia, es conveniente garantizar la calidad del proceso desde el principio y ganar en organización, por lo que se decide adoptar una metodología tradicional.

Se decide utilizar Rational Unified Process (RUP) como metodología de referencia para el desarrollo del software, por ser un proceso:

- **Iterativo e incremental**, lo cual permite dividir el proyecto en pequeños subproyectos para desarrollarlo en distintas etapas e iteraciones que resultan en un incremento del producto.

- **Dirigido por casos de uso**, el cual es uno de los métodos más utilizados y efectivos para reflejar los requisitos. Estos no solo sirven para especificar los requisitos, ellos son los encargados de guiar el ciclo de vida del proyecto.

- **Centrado en la arquitectura**, lo cual permite organizar o estructurar el sistema en sus partes más relevantes e ir refinando esta estructura progresivamente.

RUP define “un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas software, para diferentes áreas de aplicación, diferentes tipos de organizaciones, diferentes niveles de aptitud y diferentes tamaños de proyecto”. [12]

RUP propone flujos de trabajo en los que se definen las secuencias de actividades, quienes las deben desarrollar y los artefactos a generar. Entre estos flujos se encuentra el modelación del negocio donde se describen los procesos de negocio, identificando quienes participan y que actividades requieren automatización y requisitos que define qué es lo que el sistema debe hacer para lo cual se identifican las funcionalidades requeridas.

Utiliza el Leguaje Unificado de Modelado (Unified Modeling Lenguaje, UML), el cual ha sido seleccionado por el proyecto SIGEP-Cu para visualizar, especificar y construir los artefactos del sistema automatizado a definir.

1.8 Herramientas y lenguajes a utilizar

Las herramientas CASE permiten aumentar la calidad del software a través de la mejora de la productividad en el desarrollo y mantenimiento del software. Aumentan el conocimiento informático de una empresa ayudando a la búsqueda de soluciones para los requisitos. Permiten la reutilización del software, portabilidad y estandarización de la documentación, además del uso de las distintas metodologías propias de la Ingeniería del Software.

1.8.1 Visual Paradigm

Visual Paradigm es una herramienta CASE concebida para soportar el ciclo de vida completo del proceso de desarrollo del software a través de la representación de todo tipo de diagramas. Constituye una herramienta de software libre de probada utilidad para el analista. Dentro de sus características se aprecia que soporta BPMN y UML versión 2.1.

Fue diseñado para una amplia gama de usuarios interesados en la construcción de sistemas de software de forma fiable a través de la utilización de un enfoque Orientado a Objetos. Se caracteriza por:

1. Software libre.
2. Disponibilidad en múltiples plataformas (Windows, Linux).
3. Diseño centrado en casos de uso y enfocado al negocio que generan un software de mayor calidad.
4. Uso de un lenguaje estándar común a todo el equipo de desarrollo que facilita la comunicación.
5. Capacidades de ingeniería directa e inversa.
6. Modelo y código que permanece sincronizado en todo el ciclo de desarrollo
7. Disponibilidad de múltiples versiones, para cada necesidad.

Por estas características es que se utiliza Visual Paradigm for UML 3.1 Enterprise Edition como herramienta de desarrollo de software, principalmente por ser una herramienta robusta, multiplataforma y de software libre.

1.8.2 Lenguaje unificado de modelado (UML)

UML 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 [13]. UML ofrece un estándar para describir un "plano" del sistema (modelo), 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.

Es importante resaltar que UML es un "lenguaje" para especificar y no para describir métodos o procesos. Se utiliza para definir un sistema, para detallar los artefactos en el sistema y para documentar y construir. En otras palabras, es el lenguaje en el que está descrito el modelo. Se puede aplicar en el desarrollo de software entregando gran variedad de formas para dar soporte a una metodología de desarrollo de software pero no especifica en sí mismo qué metodología o proceso usar.

1.9 Conclusiones.

En este capítulo se realizó un estudio de algunos sistemas que actualmente se utilizan en el país, para apoyar la gestión de los Sistemas de Gestión Penitenciarios. Se da a conocer las deficiencias de los mismos con el objetivo de darle solución a la situación actual de este sistema. Además se sentaron las bases teóricas de las tecnologías y herramientas definidas por el equipo de arquitectura del proyecto SIGEP-Cu, teniendo como principales recursos las tecnologías libres, eslabón indispensable en las nuevas concepciones de informatización que se aplican en el país.

A partir de lo estudiado, están creadas las condiciones para conocer la solución propuesta, una vez introducidos los conceptos fundamentales para su visión.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

2.1 Introducción

Este capítulo recoge los principales conceptos tratados dentro del dominio del problema propuesto. Se relacionan los requerimientos funcionales que debe cumplir el sistema que se desarrollará, dichos requisitos fueron propuestos por el proyecto SIGEP-Cu y son el resultado del estudio realizado en relación al subsistema de Registro Legal y a los módulos Traslado y Conducción. Se representan los modelos, los diagramas y las especificaciones de casos de uso del sistema de los módulos antes mencionados. Se muestran también, las relaciones de los conceptos más importantes a través de la propuesta de los modelos conceptuales, que junto a las descripciones de estos y los diccionarios de datos que se generan como parte de los procesos que se realizan para trasladar y conducir los internos en cada uno de los establecimientos penitenciarios, constituirán las principales guías para el posterior diseño del sistema.

2.2 Requisitos de software propuestos por el proyecto SIGEP-Cu

La captura de requisitos es la actividad mediante la cual el equipo de desarrollo de un sistema de software extrae, de cualquier fuente de información disponible, las necesidades que debe cubrir dicho sistema. El proceso de captura de requisitos puede resultar complejo, principalmente si el entorno de trabajo es desconocido para el equipo de analistas, y depende mucho de las personas que participen en él. Por la complejidad que todo esto puede implicar, la ingeniería de requisitos ha trabajado desde hace años en desarrollar técnicas que permitan hacer este proceso de una forma más eficiente y precisa.

Los requerimientos funcionales son acciones fundamentales que deben tener lugar en el software al recibir información, procesarla y producir resultados. Su definición debe ser clara y libre de ambigüedades. Estos describen lo que el sistema debe hacer. [15]

A continuación se muestran los requerimientos funcionales definidos por el proyecto SIGEP-Cu para la realización de los módulos Traslado y Conducción.

2.2.1 Requisitos Funcionales Módulo Traslados

Nº	Funcionalidad	Descripción
RF1.6.1	Buscar internos a trasladar	Para registrar un traslado se debe buscar los internos a trasladar.
RF1.6.2	Registrar los datos del traslado	Una vez que se tengan los internos que se quieren trasladar se registran los datos del traslado.
RF1.6.3	Actualizar del traslado	Los datos del traslado deben poder actualizarse. Además el sistema permitirá asignar los educadores que deberán participar en el traslado y los móviles en los que se trasladarán los internos
RF1.6.4	Asignar recursos humanos al traslado	El sistema permitirá asignar los recursos humanos que deberán participar en el traslado.
RF1.6.5	Consultar traslados	El sistema permitirá visualizar todos los traslados.
RF1.6.6	Registrar ejecución de traslado	Decir si el traslado se ejecutó o no.
RF1.6.7	Buscar traslado	El sistema permitirá buscar los traslados que existen en el sistema.
RF1.6.8	Crear Plan de seguridad	El sistema permitirá crear el Plan de Seguridad de un traslado con las características específicas para tener un Plan de Seguridad.
RF1.6.9	Actualizar Plan de Seguridad	El sistema permitirá actualizar el Plan de Seguridad de un traslado.
RF1.6.10	Asignar móvil al traslado	Permite asignar los carros en que se trasladarán los internos.

RF1.6.11	Generar el modelo de traslado	Permite generar el modelo del traslado.
RF1.6.12	Generar informe de situación legal	Permite generar el informe de situación legal del interno.

Tabla 1 Requisitos Funcionales Módulo Conducción

2.2.2 Requisitos Funcionales Módulo Conducción

Nº	Funcionalidad	Descripción
RF1.5.1	Buscar internos a conducir	Para registrar un conduce se debe buscar los internos a trasladar.
RF1.5.2	Registrar los datos del conduce	Una vez que se tengan los internos que se quieren trasladar se registran los datos del conduce.
RF1.5.3	Actualizar conduce	Los datos del conduce deben poder actualizarse antes de que sea aprobado. El sistema permitirá actualizar los educadores y el móvil en que se realizará el conduce.
RF1.5.4	Asignar recursos humanos al conduce	El sistema permitirá asignar los educadores que deberán participar en el conduce y el resto del personal.
RF1.5.5	Consultar conducciones	El sistema permitirá visualizar todos los conduce.
RF1.5.6	Registrar ejecución de conduce	Decir si el conduce se ejecutó o no. En el caso de las presentaciones a juicio se especifican otros datos.

RF1.5.7	Generar modelo de conduce	El sistema deberá imprimir el modelo de conduce con los datos especificados, una vez que se hayan asignado los recursos y los datos del conduce estén completos.
RF1.5.8	Generar informe de situación legal	El sistema deberá generar un informe de la situación legal del interno que se presente a juicio como acusado.
RF1.5.9	Buscar conduce	El sistema permitirá buscar conduce por determinados parámetros.
RF1.5.10	Asignar móviles	El sistema permitirá asignar los móviles para el conduce.

Tabla 2 Requisitos Funcionales Módulo Conducción

2.3 Modelado del sistema

2.3.1 Definición de actores y casos de uso del sistema

El caso de uso es una técnica para capturar información de cómo un sistema o negocio trabaja, o de cómo se desea que trabaje. No pertenece estrictamente al enfoque orientado a objeto, es una técnica para captura de requisitos de software. Los casos de uso describen bajo la forma de acciones y reacciones el comportamiento de un sistema desde el punto de vista del usuario. Permiten definir los límites del sistema y las relaciones entre este y el entorno. Son descripciones de la funcionalidad del sistema independientes de la implementación. Los casos de uso describen qué hace el sistema, no cómo lo hace, por lo que en su modelación se hace necesario tener en cuenta la separación de los objetivos entre las vistas externas e internas. [15]

A partir de la aplicación de los patrones antes detallados se definieron los siguientes casos de uso (CU) para el módulo traslados y conducción:

Casos de uso Módulo Traslados

- Registrar los datos del traslado.

- Actualizar traslado
- Asignar recursos humanos al traslado
- Buscar traslado
- Registrar ejecución de traslado
- Consultar traslado
- Gestionar Plan de seguridad
- Asignar móvil al traslado
- Generar modelo de traslado

Casos de uso Módulo Conducción

- Registrar los datos del conduce
- Actualizar conduce
- Asignar recursos humanos al conduce
- Buscar conduce
- Registrar ejecución de conduce
- Consultar conducciones
- Asignar móvil al conduce
- Generar modelo de conduce

Actores del Sistema

Un actor no es parte del sistema, es un rol de un usuario que puede intercambiar información o puede ser un recipiente pasivo de información y representa a un ser humano, a un software o a una máquina que interactúa con el sistema. [15]

A continuación se muestran los actores del sistema de los módulos Traslados y Conducción.

Actor	Objetivo
Oficial de Control Penal	Registrar cuáles internos serán trasladados y los datos del traslado. Registrar ejecución de traslado. Generar el modelo de traslado
Jefe de Orden Interior	Asignar los recursos humanos para cada traslado. Generar modelo del plan de seguridad.

	Crea y actualiza el Plan de Seguridad.
Consultor del traslado	Consultar traslado. Los traslados pueden ser consultados por: Oficial de Guardia Operativo, Oficial de Control Penal, Jefe del centro y Segundos Jefes de Unidad, Jefe de Orden interior.
Administrador del traslado	Actualizar los traslados. El Administrador del traslado puede ser: Jefe del centro y el Oficial de Control Penal.
Jefe del centro	Asignar móvil al traslado.

Tabla 3 Actores del Sistema Módulo Traslados

Actor	Objetivo
Oficial de Control Penal	Generar el modelo para conduce.
Jefe de Orden Interior	Asignar los recursos humanos para cada conduce.
Jefe del centro	Asignar móvil al conduce.
Consultor del conduce	Consultar conducciones. Los conduce pueden ser consultados por: Oficial de Guardia Operativo, Oficial de Control Penal, Jefe del centro y Segundos Jefes de Unidad, Jefe de Orden interior.
Administrador de conduce	Actualizar los datos y recursos de los conduce. Los conduce pueden actualizados por: Jefe de Orden Interior, Jefe del centro y Registrador de conduce.

<p>Registrador de conduce</p>	<p>Registrar los conduce.</p> <p>Los conduce pueden ser registrados por: Oficial de Control Penal y Oficial de guardia operativo.</p> <p>El Oficial de Guardia Operativo introduce los datos de conduce para funerarias, hospitales, etc. que no sean presentación a juicio conduce.</p> <p>El Oficial de Control Penal introduce los datos de la citación a juicio, genera el modelo de conduce.</p>
-------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Tabla 4 Actores del Sistema Módulo Conducción

2.3.2 Diagrama de Casos de Uso del Sistema del Módulo Traslados

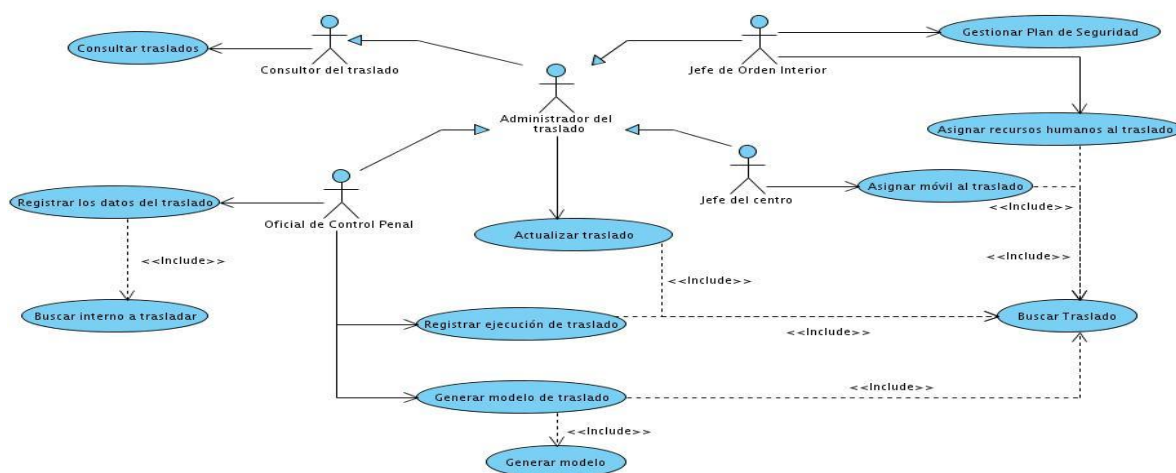


Figura 3 Diagrama de Casos de Uso del Sistema del Módulo Traslados

A continuación se enuncian los casos de uso comunes que por estar detallados con anterioridad como estrategia del análisis del proyecto SIGEP-Cu, no se le realiza la descripción de sus procesos, debido a que son utilizados por varios módulos del mismo:

- Buscar interno a trasladar

A continuación se presenta la descripción del caso de uso del sistema Registrar ejecución de conduce correspondiente al módulo Conducción. Las descripciones de los demás casos uso del sistema aparecen en los anexos.

2.3.4 Descripción de Casos de Uso del Sistema

Caso de Uso Registrar ejecución de conduce

Objetivo	Registrar si el conduce fue ejecutado o cancelado.	
Actores	Registrador de conduce	
Resumen	El Caso de Uso (CU) se inicia cuando el Registrador de conduce registra el si el conduce fue ejecutado o no.	
Referencias	RF1.5.6	
Precondiciones	El Registrador de conduce debe estar autenticado en el sistema.	
Postcondiciones	Queda actualizado el conduce.	
Flujo de eventos		
Flujo básico Registrar ejecución de conduce		
	Actor	Sistema
1.	El Registrador de conduce selecciona "Registro Legal" en el menú de la página Principal, a continuación selecciona el módulo "Conducción" y después la opción "Registrar ejecución de conduce".	
2.		El sistema ejecuta el CU incluido "Buscar conduce".
3.	El Registrador de conduce selecciona un conduce y después la opción "Siguiente".	

4.		El sistema muestra los datos del conduce seleccionado y permite que sólo se pueda modificar el estado del conduce.
5.	<p>El Registrador de conduce selecciona “Ejecutado” como estado del conduce y selecciona la opción “Aceptar”.</p> <p>Si el conduce no fue ejecutado, ver flujo alternativo 5a. “Registrar no ejecución de conduce”.</p> <p>Si el motivo del conduce es “Al Tribunal para la celebración del juicio oral o notificación de alguna resolución judicial”, ver el flujo alternativo 5b. “Registrar ejecución de juicio”</p>	
6.		El sistema registra el estado del conduce.
7.		El sistema notifica la ejecución del conduce en la hoja historial del expediente legal de cada interno involucrado en el conduce.
8.		Termina el CU.
Flujos alternos		
* Cancelar Registrar ejecución de conduce		
	Actor	Sistema
1.	El Registrador de conduce presiona el botón “Cancelar”.	
2.		Regresa al paso 1 del flujo básico “Registrar ejecución del conduce”
5a. Registrar no ejecución de conduce		
	Actor	Sistema

1.	El Registrador de conduce selecciona "Cancelado" como estado del conduce y después la opción "Aceptar".	
2.		El sistema muestra el mensaje "Introduzca el motivo por el cual no fue ejecutado el conduce" y muestra el campo: <ul style="list-style-type: none"> • Motivo de no ejecución
3.	El Registrador de conduce introduce el motivo y selecciona la opción "Aceptar".	
4.		El sistema registra el motivo por el cual no fue ejecutado el conduce.
5.		Termina el CU.
5b. Registrar ejecución de juicio		
	Actor	Sistema
1.		El sistema muestra los siguientes campos: <ul style="list-style-type: none"> • Recurso de casación
2.	El Registrador de conduce introduce el dato. Si el Registrador del conduce no introduce que es un recurso de casación continúa en el paso 5 del flujo 5b. Registrar ejecución de juicio.	
3.		Si el Registrador del conduce introduce que está en recurso de casación, el sistema muestra los siguientes datos: <ul style="list-style-type: none"> • Tribunal de casación • Sala del tribunal de casación • Municipio del tribunal de casación • Provincia del tribunal de casación

		<ul style="list-style-type: none"> Fecha de elevación al tribunal
4.	El Registrador del conduce introduce los datos y selecciona la opción "Aceptar".	
5.		<p>El sistema notifica dentro de la "Situación Legal" del interno, en la pestaña "Datos del proceso" del proceso por el que fue a juicio y los datos de la casación.</p> <p>Si el proceso no estaba registrado, ver el flujo alternativo 5a. 5. a. "Nueva causa"</p>
6.		El sistema registra la información.
7.		Termina el CU
5a. 5. a. Nueva causa		
	Actor	Sistema
1.		El sistema registra un nuevo proceso correspondiente al proceso al que pertenece la citación a juicio con los datos que se tienen del proceso.
2.		Se ejecuta el CU "Registrar trámite legal pendiente de causa pendiente".
3.		El sistema le asigna el estado de "Pendiente" al nuevo proceso.
		Termina el CU.
Relaciones	CU Incluidos	Buscar conduce
	CU Extendidos	Registrar trámite legal pendiente de causa pendiente
Requisitos no funcionales	No aplicable	

Asuntos pendientes	No aplicable
--------------------	--------------

Tabla 5 Especificación del Caso de Uso Registrar ejecución de conduce

2.4 Modelo conceptual

Teniendo en cuenta las múltiples definiciones sobre Modelo Conceptual se puede concluir que es una representación de manera visual de los conceptos u objetos del mundo real que suelen ser importantes para una determinada situación problemática o alguna que otra área de interés para una organización o empresa. No se caracterizan por representar los términos que guarden relación con los componentes de software, solo se centrarán en las clases conceptuales del dominio del problema. Se podría plantear que una clase conceptual puede ser la idea u objeto físico (símbolo, definición y extensión).

En el Modelo de Dominio no se muestra comportamiento. Las clases conceptuales pueden tener atributos pero no métodos. La finalidad de esta actividad es la de estipular una especificación del dominio del problema y los requerimientos desde la perspectiva de la clasificación por objetos y desde el punto de vista de entender los términos empleados en el dominio. Para descomponer el dominio del problema hay que identificar los conceptos, los atributos y las asociaciones del dominio que se consideran importantes. El resultado puede expresarse en un modelo conceptual, el cual se muestra gráficamente en un grupo de diagramas que describen los conceptos (objetos).

El modelo conceptual no es una descripción de los componentes del software; representa los conceptos en el dominio del problema en el mundo real. [13]

Entre los elementos que componen estos diagramas pueden existir asociaciones, que no es más que una relación entre dos conceptos que indica alguna conexión significativa e interesante entre ellos. [13]

2.4.1 Propuesta del Modelo Conceptual del Módulo Traslados

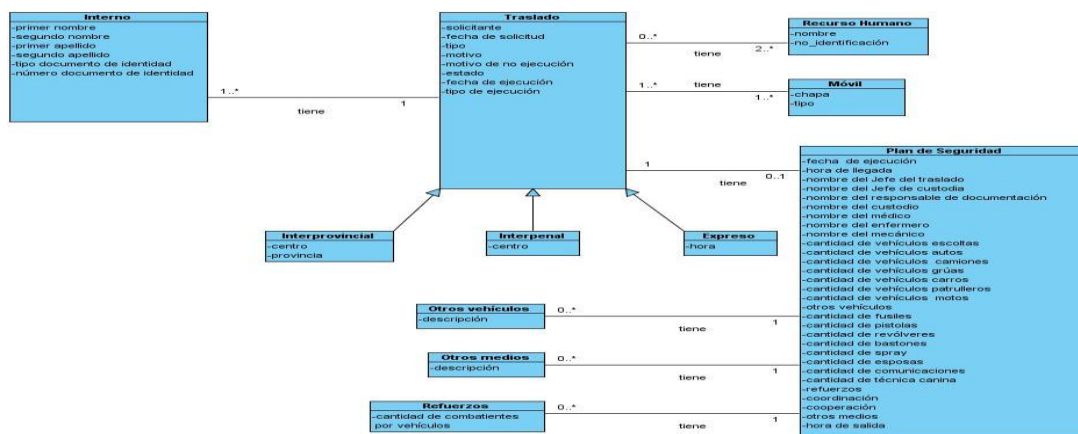


Figura 5 Modelo Conceptual del Módulo Traslados

2.4.2 Propuesta del Modelo Conceptual del Módulo Conducción

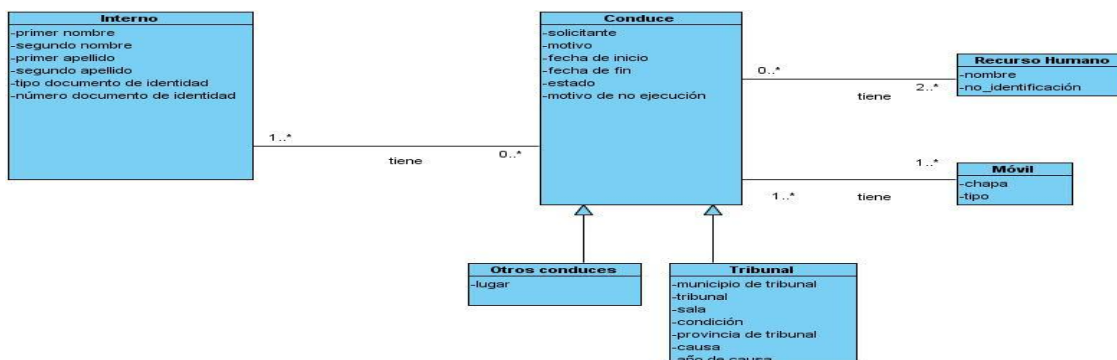


Figura 6 Modelo Conceptual del Módulo Conducción

2.4.3 Descripción del Modelo Conceptual

La descripción del modelo conceptual es importante ya que brinda la información necesaria para el mejor entendimiento del mismo. Uno de los principales aspectos a tener en cuenta dentro de este apartado son las asociaciones que se establecen entre los conceptos que se representan, ellas muestran la forma en que se interactúan y se relaciona en el dominio donde se enmarca el problema. Los modelos anteriores, describen las siguientes relaciones entre conceptos:

2.4.3.1 Descripción del Modelo Conceptual del Módulo Traslados

Interno tiene traslado: Relación que se establece entre las entidades interno y traslado que posibilita determinar cuales son los internos asociados a un determinado traslado.

Traslado tiene Plan de Seguridad: Relación que se establece entre las entidades traslado y plan de seguridad con el fin de confeccionar dicho plan para internos con condiciones especiales, por lo cual es necesario que el traslado se ejecute individualmente y se especifiquen datos adicionales los cuales se detallan en el documento antes mencionado.

Traslado tiene Recurso Humano: Relación que se establece entre las entidades traslado y recurso humano con el objetivo de asignar todos los educadores auxiliares que se destinarán para trasladar los internos.

Traslado tiene Móvil: Relación que se establece entre las entidades traslado y móvil para asignar los móviles que se destinarán para trasladar los internos.

Plan de Seguridad tiene Otros vehículos: Relación que se establece entre las entidades Plan de Seguridad y Otros vehículos con el fin de especificar, en caso de ser necesario los otros tipos de vehículos que se necesitan para llevar a cabo el traslado.

Plan de Seguridad tiene Otros Medios: Relación que se establece entre las entidades Plan de Seguridad y Otros Medios que da la posibilidad de registrar, en caso de ser necesario la relación de otros medios que se estimen necesario utilizar para realizar el traslado.

Plan de Seguridad tiene Refuerzos: Relación que se establece entre las entidades Plan de Seguridad y Refuerzos para almacenar la cantidad de combatientes por vehículos en caso de que se necesiten utilizar refuerzos para ejecutar el traslado.

2.4.3.2 Descripción del Modelo Conceptual del Módulo Conducción

Interno tiene Conduce: Relación que se establece entre las entidades interno y conduce que posibilita determinar cuales con los internos asociados a un determinado conduce.

Conduce tiene Recurso Humanos: Relación que se establece entre las entidades conduce y recurso humano con el objetivo de asignar todos los educadores auxiliares que se destinarán para conducir a los internos.

Conduce tiene Móvil: Relación que se establece entre las entidades conduce y móvil para asignar los móviles que se destinarán para conducir a los internos.

2.5 Validación de los requisitos funcionales.

La validación de requisitos es considerada la actividad donde clientes y usuarios, con ayuda de los desarrolladores, revisan los requerimientos definidos para confirmar que realmente reflejan sus necesidades y que definen el producto deseado. El objetivo de la validación de requisitos es descubrir problemas en los requisitos identificados, antes de comprometer recursos a su implementación. Como consecuencia de esta actividad se suele producir una nueva iteración de adquisición de requisitos, debido a que conforme se perfila el sistema, suelen ir apareciendo nuevas necesidades hasta entonces ocultas, sobre todo cuando se utilizan prototipos.

En el desarrollo de la ingeniería de requerimientos aplicada a los módulos Traslado y Conducción se utilizó la técnica de validación de prototipos. Esta técnica hace posible que el usuario tenga una idea clara del producto que va a recibir. Se tiene una mayor fluidez en la comunicación desarrollador-cliente. El desarrollador porque es quién está desarrollando esos prototipos y los usuarios porque de su aceptación o no, depende el desarrollo del producto que en un momento van a utilizar.

A continuación se presentan uno de los prototipos que valida una de las funcionalidades antes identificadas. Los prototipos que validan las demás funcionalidades se podrán encontrar en los anexos.

El prototipo muestra una ventana de diálogo con el título "Registrar ejecución de conduce". Contiene los siguientes campos de entrada:

- Solicitante: campo de texto.
- Motivo: menú desplegable.
- Estado: menú desplegable.
- Lugar: campo de texto.
- Fecha de inicio: campo de texto.
- Fecha de fin: campo de texto.

En la parte inferior derecha hay dos botones: "Aceptar" y "Cancelar".

Figura 7 Prototipo elemental de interfaz gráfica de usuario CU Registrar ejecución de conduce

2.6 Conclusiones.

Con la confección de este capítulo fue posible identificar y describir las funcionalidades que presentarán los módulos Traslado y Conducción del proyecto SIGEP-Cu. Se validaron los requisitos funcionales propuestos por dicho proyecto para estos dos módulos. Se modeló el sistema quedando definido actores, casos de uso, así como las especificaciones de cada caso de uso. Fue posible entender lo que el cliente requiere y utilizar técnicas que posibilitan su especificación de forma sencilla y a la vez que facilite su integración con las demás fases del desarrollo de software. Este proceso se tornó satisfactorio e importante ya que se interactuó con el ambiente del problema, adquiriendo mayor conocimiento de las operaciones que se incluyen en la solución propuesta. Con la definición, especificación y validación de las funcionalidades obtenidas se tienen los artefactos necesarios para analizar cada funcionalidad a nivel de casos de uso para su posterior diseño.

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

3.1 Introducción

Este capítulo recoge los elementos considerados para elaborar el análisis y diseño de los módulos Traslado y Conducción pertenecientes al subsistema Registro Legal. En el mismo se explican los aspectos importantes dentro del modelo del análisis y diseño propuesto, sentando las bases para la implementación de un sistema que satisfaga las necesidades del cliente y responda eficientemente a los requerimientos identificados. Lo cual comprende el modelado de los dos módulos antes mencionados, por lo que se definen las clases del análisis y se esboza un diagrama de paquetes con sus relaciones. Se desarrollan las clases del diseño, conformando el modelo del diseño. Igualmente se hace referencia a las decisiones arquitectónicas tomadas por el proyecto SIGEP-Cu.

La importancia del diseño del software se puede describir con una sola palabra -calidad-. El diseño es el lugar en donde se fomenta la calidad en la ingeniería del software. El diseño es la única forma de convertir exactamente los requisitos de un cliente en un producto o sistema de software finalizado. [15]

3.2 Modelo del Análisis

Durante el análisis, se trabajará con los requisitos que se describen en la captura de requerimientos, refinándolos y estructurándolos. El objetivo de hacerlo es conseguir una comprensión más precisa de los requisitos y una descripción de los mismos que sea fácil de mantener y que nos ayude a estructurar el sistema entero, incluyendo su arquitectura. [14]

3.2.1 Definición del modelo de paquetes

La definición de un modelo de paquetes donde se representaran los módulos que se relacionan con Traslado y Conducción, dentro del subsistema Registro Legal y con otros subsistemas del SIGEP-Cu permitió definir la estructura y las dependencias entre cada uno de ellos. Los paquetes son utilizados como contenedores de elementos. Cada elemento puede estar contenido dentro de un único paquete, sin embargo un paquete puede contener otros paquetes.

Dentro del subsistema Registro Legal los módulos Traslado y Conducción se relacionan con los siguientes módulos:

- Situación legal
- Solicitudes

Por otra parte, dentro del subsistema Gestión de información para el análisis se relacionan con el módulo:

- Alertas

Y además se relacionan con el módulo:

- ETES

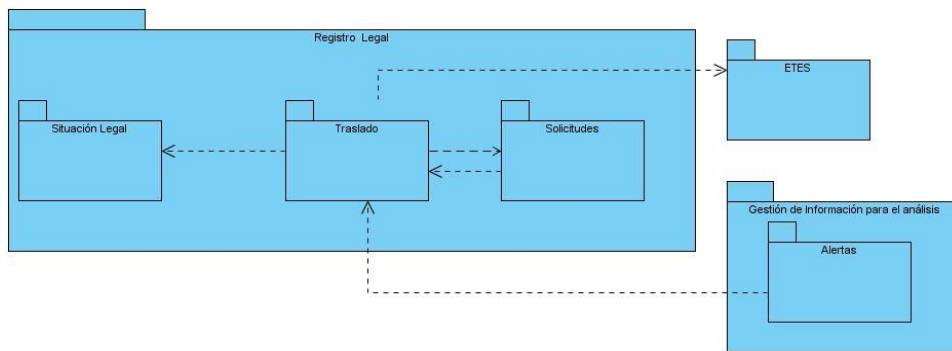


Figura 8 Diagrama de paquetes relacionados con el módulo Traslados

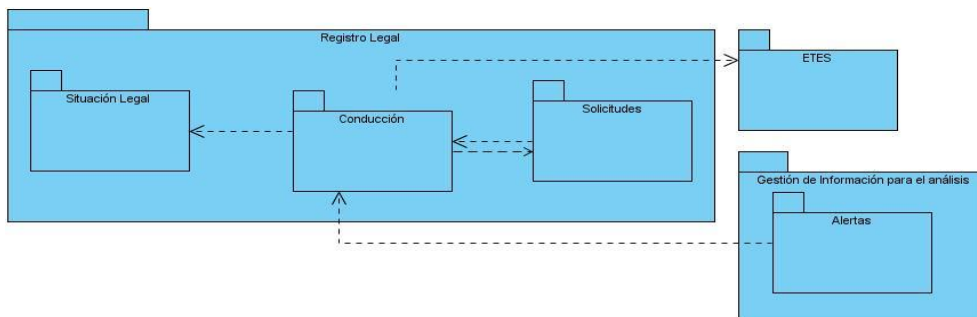


Figura 9 Diagrama de paquetes relacionados con el módulo Conducción

3.2.2 Definición del modelo de clases del análisis

El diagrama de clases del análisis es un artefacto que permitió representar los conceptos en un dominio del problema. Se modelaron las clases y sus relaciones, lo que permitió representar una vista estática del sistema. El modelado de los diagramas de clases del análisis propone la utilización de tres estereotipos de clases estandarizados en UML para ayudar a los desarrolladores a distinguir el ámbito de las diferentes clases.




NOMBRE	CARACTERÍSTICAS	REPRESENTACIÓN
Entidad	Modelan información que posee larga vida y que es a menudo persistente.	 Clase de entidad
Interfaz	Modelan la interacción entre el sistema y sus actores.	 Clase de interfaz
Control	Coordinan la realización de uno o unos pocos casos de uso coordinando las actividades de los objetos que implementan la funcionalidad del caso de uso.	 Clase de control

Figura 10 Estereotipos de las clases del análisis

A continuación se presenta el diagrama de clases del análisis del caso de uso Registrar ejecución de conduce correspondiente al módulo Conducción, el resto de los diagramas se podrán encontrar en los anexos.

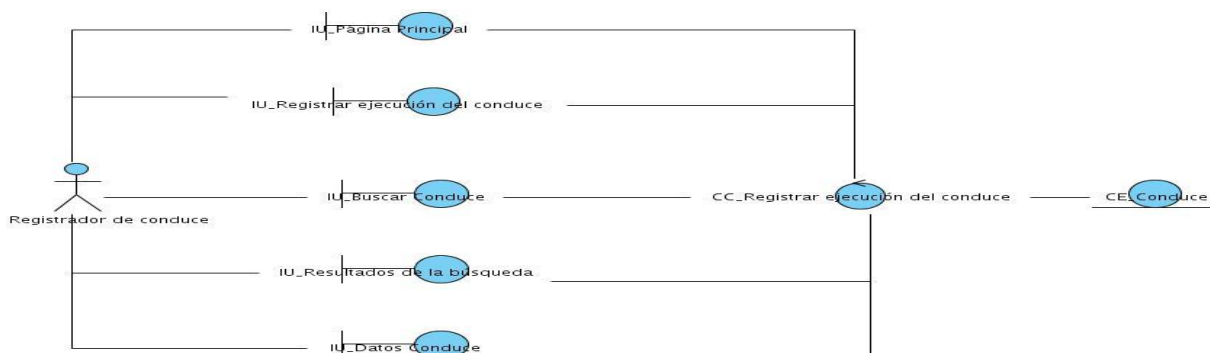


Figura 11 Diagrama de clases del análisis CU Registrar ejecución de conduce

3.2.3 Diagramas de interacción. Colaboración

Un diagrama de colaboración muestra una interacción organizada en torno a los objetos que efectúan operaciones; es parecido a un diagrama de objetos que muestra los objetos y los enlaces existentes entre ellos que se necesitan para implementar una operación de nivel más elevado. Una colaboración modela los objetos y los enlaces significativos dentro de una interacción. Los objetos y los enlaces son significativos solamente en el contexto proporcionado por la interacción. Un rol describe un objeto, y un rol en la asociación describe un enlace dentro de una colaboración. Un diagrama de colaboración muestra los roles en la interacción en una disposición geométrica. Los mensajes se muestran como flechas, ligadas a las líneas de la relación, que conectan a los roles. La secuencia de mensajes, se indica con los números secuenciales que preceden a las descripciones del mensaje. A continuación presentamos el diagrama de colaboración del caso de uso Registrar ejecución de conduce correspondiente al módulo Conducción, los diagramas de los demás CU se podrán encontrar en los anexos.

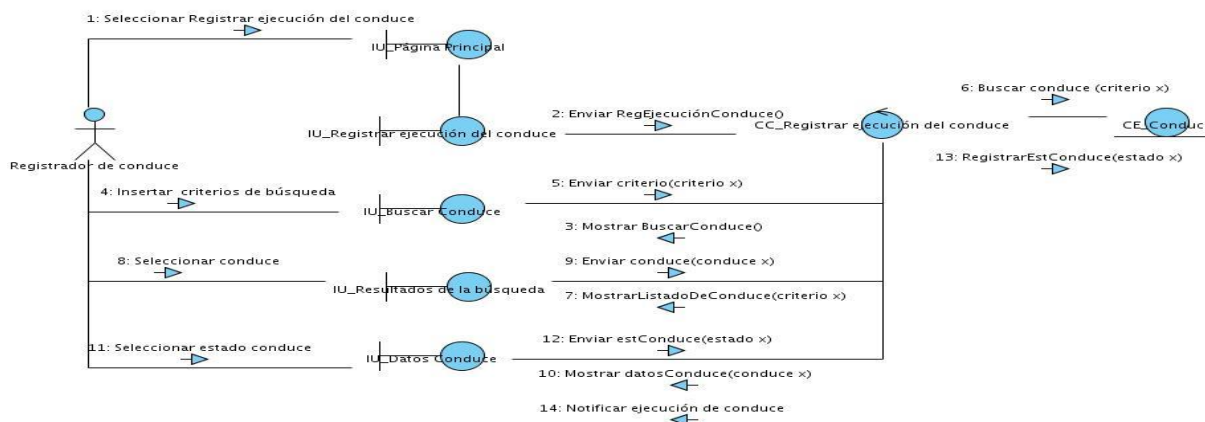


Figura 12 Diagrama de colaboración CU Registrar ejecución de conduce

3.3 Arquitectura del sistema

La arquitectura proporciona una visión global del sistema a construir. Describe la estructura y la organización de los componentes del software, sus propiedades y las conexiones entre ellos. Los

componentes del software incluyen módulos de programas y varias representaciones de datos que son manipulados por el programa. [15]

La arquitectura de software es quien le da forma al software para que soporte todos los requisitos. Establece los fundamentos para que todos los involucrados en el equipo de desarrollo trabajen en una línea común que permita alcanzar los objetivos y necesidades del sistema. Se necesita una arquitectura robusta, que guíe el proceso de desarrollo, y que defina de manera abstracta los componentes que llevan a cabo alguna tarea, sus interfaces y la comunicación entre ellos. [15]

La arquitectura del proyecto SIGEP-Cu está basada principalmente en las funcionalidades y facilidades que propone el framework de desarrollo Grails. Se define un sistema con arquitectura de tres capas, Capa de Presentación, Capa de Servicios de Negocio y Capa de Acceso a Datos. Además está instalada sobre la tecnología Cliente-Servidor.

La aplicación está dividida en tres capas lógicas fundamentales:

- **Capa de Presentación:** En esta capa se encuentran las Vistas y la Lógica de Presentación. En la Lógica de Presentación se maneja todo el flujo web utilizando la implementación del patrón Modelo Vista Controlador que nos brinda Spring MVC. Las Vistas son los recursos que junto al modelo generado por los controladores le permiten al cliente visualizar la información, estos pueden ser páginas HTML documentos en formato PDF, hojas de cálculo, entre otras.
- **Capa de Servicios de Negocio:** Encapsula toda la lógica de la aplicación en fachadas de negocio que son utilizadas por los controladores en la capa de presentación y se exponen algunos procesos de negocio a través de interfaces de servicios. A estas fachadas de negocio se le aplican la seguridad a nivel de métodos y de objetos de negocio, auditorias, cache, política de transacciones, entre otros.
- **Capa de Acceso a Datos:** Maneja los objetos de acceso a datos abstrayéndolos del mecanismo de persistencia usado; a través de interfaces que exponen las operaciones de persistencia definidas para cada uno de los DAOs e implementadas utilizando Hibernate, que es un framework de mapeo a objeto-relacional (ORM).

3.3.1 Decisiones arquitectónicas

Se decide trabajar con el Framework de desarrollo Grails. Grails es una plataforma para desarrollo de aplicaciones web basada en el lenguaje de tipado dinámico Groovy, se compila y ejecuta en una JVM (Máquina virtual de Java). Su desarrollo fue inspirado en SmallTalk, Python y Ruby. Entre sus ventajas se encuentran que se integra con el lenguaje Java en cualquier nivel, además incorpora componentes que van desde los flujos webs hasta la capa de acceso a datos. El lenguaje de programación que se utiliza es Groovy, el mismo está basado en herramientas existentes y poderosas como: Spring, Hibernate, Sitemesh, entre otras. Grails mediante Groovy abstrae al desarrollador de la complejidad de programar directamente en la plataforma JEE usando de manera subliminal y muy sencilla Spring e Hibernate quienes han sido proyectos nacidos como OpenSource que han mejorado la manera de desarrollar en Java, por tanto representa el siguiente nivel de abstracción para JEE.

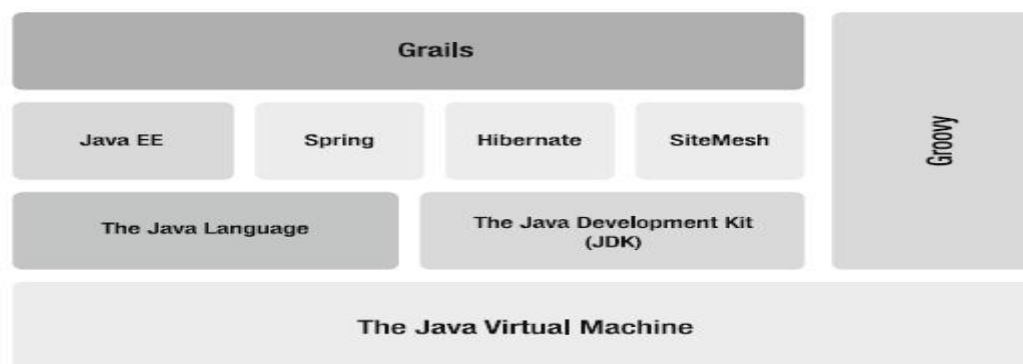


Figura 13 Arquitectura propuesta por Grails

3.4 Patrones a utilizar

Cada patrón describe un problema que ocurre una y otra vez en nuestro ambiente, y luego describe el núcleo de la solución a ese problema, de tal manera que puedes usar esa solución un millón de veces más, sin hacer jamás la misma cosa dos veces.

3.4.1 Patrones de asignación de Responsabilidades GRASP

Los patrones GRASP (General Responsibility Assignment Software Patterns), o sea patrones generales de software para asignar responsabilidades, son patrones generales de software para asignación de responsabilidades, aunque se considera que más que patrones, son una serie de "buenas prácticas" de aplicación recomendable en el diseño de software.

En los patrones GRASP se codifican algunos de los principios, que se aplican al preparar los diagramas de interacción.

- Bajo acoplamiento.

Este patrón soluciona el inconveniente de dar soporte a una dependencia escasa y a un aumento de la reutilización.

- Alta cohesión.

Este patrón es utilizado para mantener la complejidad dentro de los límites manejables.

El diseño obtenido cumple con los patrones de Bajo acoplamiento y Alta cohesión permitiendo la colaboración entre los elementos del diseño, sin verse afectados la reutilización de los mismos y el entendimiento de estos cuando se encuentran aislados. La creación de clases controladoras facilitó realizar las operaciones del sistema, debido a que estas operaciones reflejan los procesos de la empresa o dominio y no es factible manejarse en la capa de interfaz o presentación.

3.4.2 Patrones Estructurales

- Facade.

El propósito de utilizar este patrón es proveer de una interfaz unificada y sencilla que haga de intermediaria entre un cliente y una interfaz o grupo de interfaces más complejas. Este patrón permite que una biblioteca de software sea más fácil de usar y entender. Esto es posible porque el Facade implementa métodos convenientes para tareas comunes, puede reducir la dependencia de código externo en los trabajos internos, permitiendo así más flexibilidad en el desarrollo de sistemas.

3.4.3 Patrón de acceso a datos (DAO)

- DAO

La utilización de este patrón permitirá acceder a la fuente de datos y encapsular los objetos clientes, ocultando tanto la fuente como el modo de acceder a ella. Los DAOs deben implementar los métodos del interface (InterfaceDAO) que declaran. Pero además pueden implementar otros métodos que no están en el interfaz. DAO permite el acceso a reglas de validación, esto es posible porque tiene capacidad de especificar relaciones entre tablas.

3.5 Modelo del diseño

El proceso del diseño establece un conjunto de pasos que permiten al diseñador describir todos los aspectos del sistema a construir. Constituye un enfoque de la ingeniería de software que modela un sistema como un grupo de objetos que interactúan entre sí, representando un dominio en términos de conceptos clasificados de acuerdo a su dependencia funcional.

El modelo del diseño es un modelo de objetos que describe la realización física de los casos de usos centrándose en como los requisitos funcionales y no funcionales, junto con otras restricciones relacionadas en el entorno de implementación, tienen un impacto en el sistema a considerar y sirve de abstracción a la implementación y al código fuente del sistema. Contiene protocolo, cápsula, realización de casos de usos, señales, eventos, subsistema de diseño, paquetes de diseño, interfaces, clases del diseño, clases de prueba, diseño de pruebas.

3.5.1 Diagrama de clases con estereotipos WEB

La aplicación a desarrollar es Web, es decir, que tiene como elemento significativo de su arquitectura un navegador y un protocolo de comunicación HTTP. Para el modelado de este tipo de aplicaciones se aprovecha una de las características que le da flexibilidad a la notación de UML. La misma consiste en un conjunto de mecanismos de extensión. El uso de estereotipos permite enriquecer el significado de los elementos clásicos de su notación. A continuación se muestran uno de estos diagramas, el resto se encuentran en los anexos.

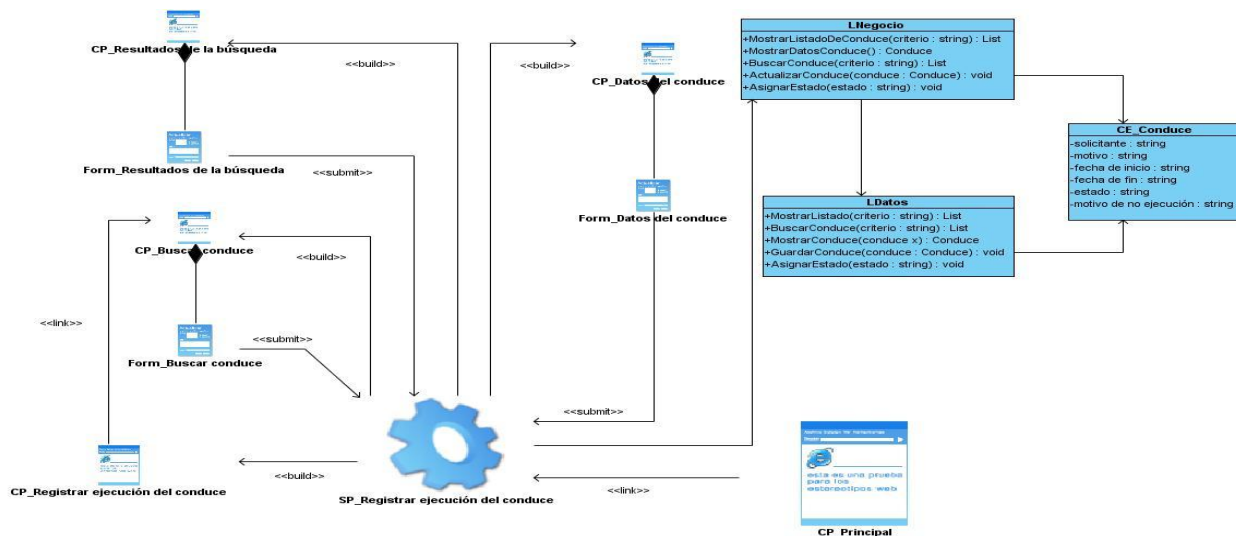


Figura 14 Diagrama de clases con estereotipos web CU Registrar ejecución de conduce

3.5.2 Diagrama de interacción del diseño. Secuencia

En el flujo de diseño se utiliza fundamentalmente el diagrama de secuencia. En él se incluyen las interacciones entre las clases del diseño, a través de mensajes. Un mensaje significa una operación en la clase a la que va el mensaje. El objetivo de realizar diagramas de secuencia hace que se tenga en cuenta el orden y el momento en que se envían los mensajes a los objetos. Los mensajes que se envían entre las clases pueden ser síncronos, donde el tipo normal de llamada del mensaje le pasa el control al objeto llamado hasta que el método finalice, o asíncronos donde se devuelve el control directamente al objeto que realiza la llamada. Este tipo de diagrama se destaca por tener la línea de vida y el foco de control que representa el período de tiempo durante el cual un objeto ejecuta una acción. A continuación se muestra el diagrama de secuencia del caso de uso Registrar ejecución de conduce correspondiente al módulo Conducción, el resto aparecen en los anexos.

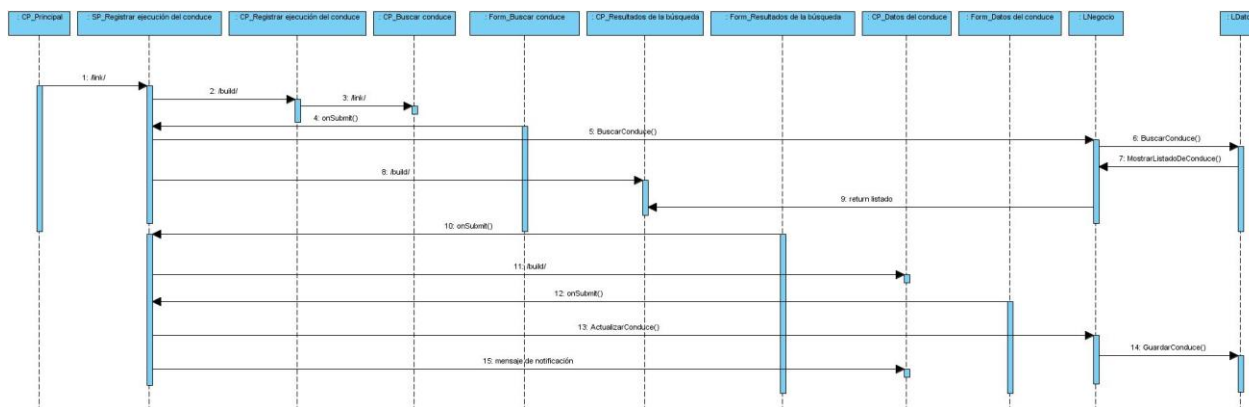


Figura 15 Diagrama de interacción del diseño CU Registrar ejecución de conduce

3.6 Métricas para el Modelo de Diseño

Para realizar la validación a las clases del diseño se seleccionaron las métricas Tamaño operacional de clase y Relaciones entre clases. Dichas métricas validan que los patrones Alta Cohesión y Bajo Acoplamiento fueron bien utilizados.

Tamaño operacional de clase (TOC): Numero de métodos. Responsabilidad, aumenta para TOC+, Complejidad de implementación, aumenta TOC+, Reutilización, disminuye TOC+.

Relaciones entre clases (bajo) (RC): Numero de relaciones de uso de una clase. Acoplamiento, aumenta RC+. Complejidad del mantenimiento, aumenta RC+. Reutilización, disminuye RC+. Cantidad de Pruebas, aumenta RC+.

Para aplicar la métrica TOC se tuvo en cuenta la cantidad de procedimientos que tenían las clases de los módulos Traslado y Conducción, a partir del promedio de los procedimientos y mediante un criterio se obtuvo la categoría (baja, media, alta) para la Responsabilidad, Complejidad y Reutilización.

Luego de efectuar los cálculos para las clases de los módulos antes mencionados se obtuvieron los siguientes resultados:

Total de clases	18
Promedio de procedimientos	4.333333333

Figura 16 Tamaño operacional de clase Módulo Traslados

Total de clases	16
Promedio de procedimientos	4

Figura 17 Tamaño operacional de clase Módulo Conducción

Se lograron las siguientes gráficas para la Responsabilidad, Complejidad y Reutilización de ambos módulos:



Figura 18 Representación de la responsabilidad de las clases del Módulo Traslados



Figura 19 Representación de la responsabilidad de las clases del Módulo Conducción



Figura 20 Representación de la reutilización de las clases del Módulo Traslados

Reutilización	Cantidad de clases	Promedio
Alta	16	30.76923077
Media	0	0
Baja	0	0

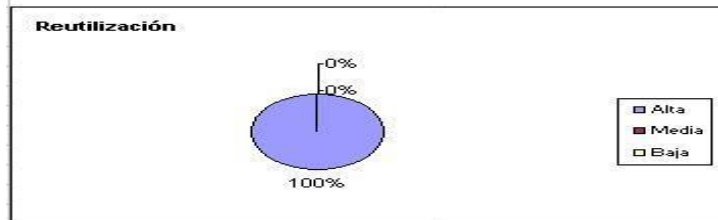


Figura 21 Representación de la reutilización de las clases del Módulo Conducción

Complejidad	Cantidad de clases	Promedio
Baja	18	34.61538462
Media	0	0
Alta	0	0

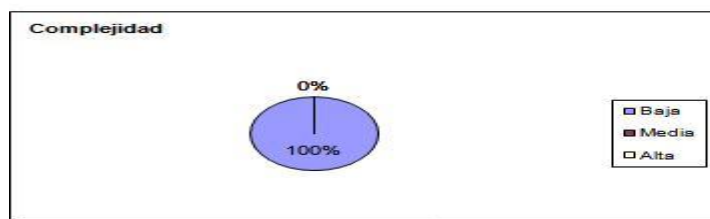


Figura 22 Representación de la complejidad de las clases del Módulo Traslados

Complejidad	Cantidad de clases	Promedio
Baja	16	30.76923077
Media	0	0
Alta	0	0

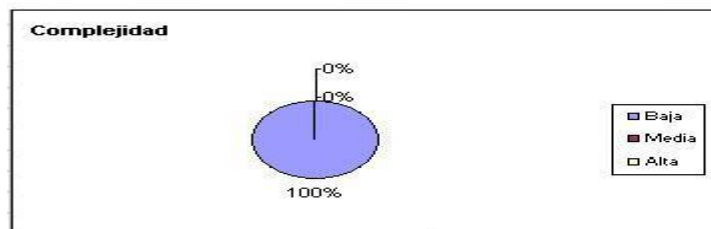


Figura 23 Representación de la complejidad de las clases del Módulo Conducción

Cuando la responsabilidad y complejidad (Figura 43, 44, 47, 48) de las clases entre media y baja es mayor que el 80% se puede decir que cumple con el patrón de alta cohesión, además de que presentan una reutilización alta.

Para aplicar la métrica RC se tuvo en cuenta la cantidad de relaciones que tenían las clases de los módulos Traslado y Conducción, a partir del promedio de las relaciones y mediante un criterio se

obtuvo la categoría (baja, media, alta) para el Acoplamiento, Complejidad de Mantenimiento, Reutilización y Cantidad de Pruebas.

Luego de efectuar los cálculos para un:

Total de clases	18
Promedio de asociaciones de uso	1.277777778

Figura 24 Relaciones entre clases Módulo Traslados

Total de clases	16
Promedio de asociaciones de uso	1.1875

Figura 25 Relaciones entre clases Módulo Conducción

Se obtuvieron las siguientes gráficas para cada uno de los módulos:

Para el acoplamiento.

Acoplamiento	Cantidad de clases	Promedio
Ninguno	0	0
Bajo	14	17.94871795
Medio	3	3.846153846
Alto	1	1.282051282

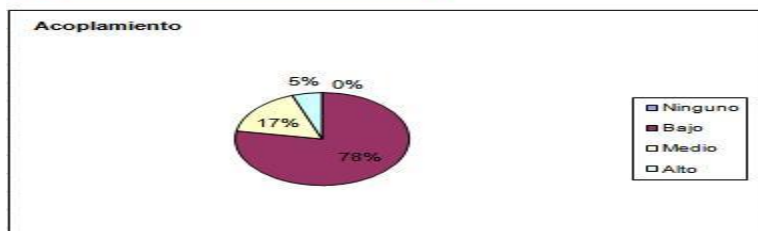


Figura 26 Acoplamiento de las clases del Módulo Traslados

Acoplamiento	Cantidad de clases	Promedio
Ninguno	0	0
Bajo	13	16.66666667
Medio	3	3.846153846
Alto	0	0

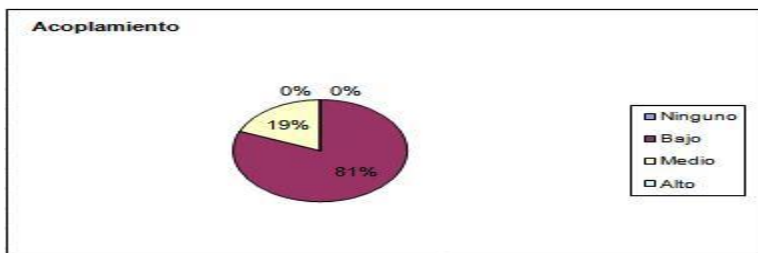


Figura 27 Acoplamiento de las clases del Módulo Conducción

Para la reutilización.

Reutilización	Cantidad de clases	Promedio
Baja	0	0
Media	4	5.128205128
Alta	14	17.94871795



Figura 28 Reutilización de las clases del Módulo Traslados

Reutilización	Cantidad de clases	Promedio
Baja	0	0
Media	3	3.846153846
Alta	13	16.66666667



Figura 29 Reutilización de las clases del Módulo Conducción

Para la complejidad de Mantenimiento.

Complejidad de Mantenimiento	Cantidad de clases	Promedio
Baja	14	17.94871795
Media	4	5.128205128
Alta	0	0

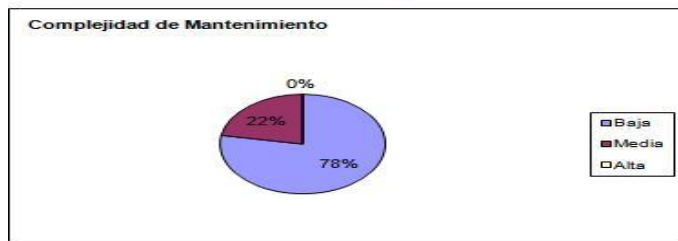


Figura 30 Complejidad de Mantenimiento para las clases del Módulo Traslados

Complejidad de Mantenimiento	Cantidad de clases	Promedio
Baja	13	16.66666667
Media	3	3.846153846
Aita	0	0

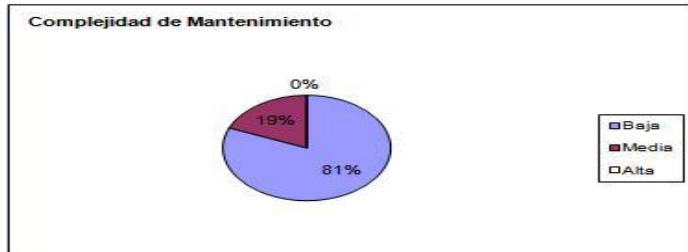


Figura 31 Complejidad de Mantenimiento para las clases del Módulo Conducción

Para la cantidad de Pruebas.

Cantidad de Pruebas	Cantidad de clases	Promedio
Baja	14	17.94871795
Media	4	5.128205128
Aita	0	0

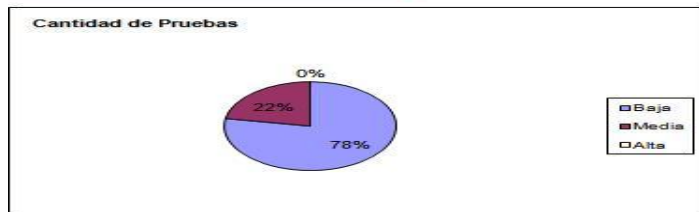


Figura 32 Cantidad de Pruebas para las clases del Módulo Traslados

Cantidad de Pruebas	Cantidad de clases	Promedio
Baja	13	16.66666667
Media	3	3.846153846
Aita	0	0

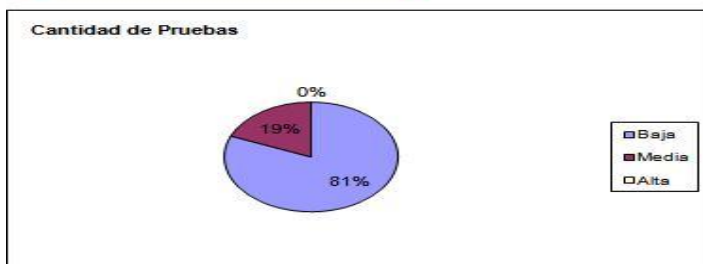


Figura 33 Cantidad de Pruebas para las clases del Módulo Conducción

3.7 Conclusiones

El diseño del software sirve como fundamento para continuar con el desarrollo del software y de la ingeniería del software. La culminación de este capítulo es un paso de gran importancia para la implementación del sistema. Con el análisis de los casos de uso identificados previamente se evidenció el flujo de las acciones entre las clases del análisis para cada funcionalidad descrita. Posibilitando así, la correcta transformación de los mismos a un diseño que indique como debe ser implementado el software.

Mediante la aplicación de las métricas TOC y RC se cumple en el diseño con los patrones de alta cohesión y bajo acoplamiento para el diseño de las clases, sus procedimientos y relaciones.

CONCLUSIONES

Con la culminación de la investigación realizada en el presente trabajo de diploma, donde se realiza un estudio del subsistema de Registro Legal enfocado a los módulos Traslados y Conducción, fue posible desarrollar todas las tareas, con el fin de cumplir los objetivos propuestos. Con el estudio y comprensión de la notación, metodología, herramienta y lenguaje, que fue de vital importancia para el desarrollo de la solución que se propone, se obtuvieron resultados que permitieron llegar a las siguientes conclusiones:

- A partir de los requerimientos funcionales propuestos se definieron los casos de uso y actores que intercambian información en el sistema.
- Se estudiaron los casos de uso identificados, a partir del desarrollo de las clases del análisis, posibilitando el flujo de interacción entre clases del análisis.
- Se diseñaron los diagramas de clases para cada uno de los módulos.

Posterior a la investigación, se desarrollaron los siguientes artefactos:

- Definición de actores y casos de uso del sistema.
- Descripción de los casos de uso del sistema.
- Modelo de casos de uso del sistema. Análisis.
- Diccionario de datos.
- Definición de las clases del análisis.
- Diagramas de colaboración.
- Modelo de paquetes.
- Definición de las clases del diseño.
- Modelo de diseño.
- Diagramas de secuencia del diseño.

De forma general los resultados obtenidos a partir del análisis y diseño del sistema fueron los esperados, tomando como referencia la aplicación de métodos y métricas para validar la solución del mismo.

REFERENCIAS BIBLIOGRÁFICAS

- [1] ÓCEANO, E. G. Diccionario de la Lengua española y de nombres propios. Barcelona, España, Océano, p. 84-494.
- [2] OSORIO, M. Diccionario de Ciencias Jurídicas, Políticas y Sociales.
- [3] Reglamento del Sistema Penitenciario de Cuba
- [4] PÉREZ, E. A. Conceptualización del proceso Control Legal, 2007.
- [5] Referencia al proyecto técnico rev12
- [6] KNIGHT, T. C. H. Sistema Informativo de la Dirección de Establecimientos Penitenciarios, 2005.
- [7] Ingeniería, Departamento central de. 2009. Entorno Virtual de Aprendizaje. [En línea] 2009. [Citado el: 2 de Febrero de 2010.] http://eva.uci.cu/file.php/102/Curso_2009-2010/Semana_5/Seminario_2/Materiales_Complementarios/Tecnologia_BPM.pdf
- [8] Ingeniería, Departamento central de. 2009. Entorno Virtual de Aprendizaje. [En línea] 2009. [Citado el: 20 de Enero de 2010.] <http://eva.uci.cu/mod/resource/view.php?id=21011>
- [9] Sommerville, Ian. 2005. Ingeniería de software. Madrid: s.n., 2005.
- [10] Cruz, Aracelis Reina Betancourt. 2009. Conferencia Introducción a la Ingeniería de Requisitos. Ciudad de la Habana: Universidad de las Ciencias Informáticas, 2009.
- [11] CANÓS, J. H.; P. LETELIER. Metodologías Ágiles en el Desarrollo de Software.
- [12] JACOBSON, I.; G. BOOCH. El Proceso Unificado del Desarrollo del Software. . 435 p.
- [13] Sommerville, Ian. Ingeniería de Software. Sexta edición. s.l. : Addison-Wesley, 2002. ISBN: 970-26-0206-8.
- [14] Ivar Jacobson, James Rumbaugh, Grady Booch. El proceso unificado de desarrollo de software. [ed.] Addison-Wesley. Madrid : Pearson, 2000. ISBN 84-7829-036-2.
- [15] Pressman, Roger S. Ingeniería de Software, un enfoque práctico. Quinta edición. S.l. : McGraw-Hill Companies, 2002. ISBN: 8448132149.

BIBLIOGRAFÍA

1. Schuller, Joseph. Aprendiendo UML en 24 horas, 2000
2. Ivar Jacobson, Grade Booch, James Rumbaugh. El proceso unificado de desarrollo de software, 2000.
3. Pressman, Roger. S. Ingeniería de software: Un enfoque práctico. 2002, vol. Quinta edición
4. LARMAN, C. UML Y Patrones. Introducción al análisis y diseño orientado a objeto. México 1999. p.
5. VISUAL, PARADIGM, 2009. [Disponible en: <http://www.visual-paradigm.com/product/vpuml/>
6. ÖVERGAARD, Gunnar; PALMKVIST, Karin, "Use Cases: Patterns and Blueprints". 2004. Addison Wesley.

GLOSARIO DE TÉRMINOS

C

CASE: Computer Aided Software Engineering (Ingeniería de Software Asistida por Computación).

Caso de uso (CU): Conjunto de secuencias de acciones que un sistema ejecuta y que produce un resultado observable.

Conducción: Acción de presentar bajo custodia a un interno en determinado lugar donde su permanencia es temporal y es custodiado todo el tiempo por los educadores auxiliares.

D

DEP: Dirección de establecimientos penitenciarios.

H

HTTP (Protocolo de Transferencia de Hipertexto): Define la sintaxis y la semántica que utilizan los elementos software de la arquitectura Web para comunicarse; protocolo orientado a transacciones y sigue el esquema petición-respuesta entre un cliente y un servidor.

I

Ingeniería de software: La aplicación de un enfoque sistemático disciplinado y cuantificable al desarrollo, operación y mantenimiento del software.

N

Navegador: Programa utilizado para ubicar y ver páginas Web. Por ejemplo Netscape.

P

Penado: Persona del sexo femenino o masculino que se encuentra en un centro penitenciario en cumplimiento de una sanción firme de privación de libertad.

Procesado: Persona de sexo femenino o masculino que se le señale como autor o participe de un hecho punible y que ingresa al Sistema Penitenciario en virtud de Auto de Privación Judicial de libertad dictado por el Juez de Control o de Juicio según el estado del proceso.

R

RUP: Rational Unified Process (Proceso Unificado de Desarrollo de Software).

S

SACORE: Sistema para el Control del Recluso.

SACDEP: Sistema de capacidades de la dirección de establecimientos penitenciarios.

SAIDEP: Sistema de incidencias de la dirección de establecimientos penitenciarios.

SIGEP-Cu: Sistema de Gestión Penitenciario Cubano.

Software: Conjunto de instrucciones que al se ejecutadas proporcionan las características, funciones y el grado de desempeño deseados.

T

Traslado: Movimiento de un interno o grupo de ellos, de un lugar de internamiento a otro.

U

UML: Lenguaje Unificado de Modelado.