

Universidad de las Ciencias Informáticas



Facultad 15

Título: “Diseño e Implementación del módulo Situación Legal del Sistema Penitenciario Nacional.”

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

Autor(es): José A. Gómez Llano
Yohairo B. Consuegra Peña

Tutor(es): Ing. David Vargas Hernández

Ciudad de La Habana, Julio del 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 firmamos la presente a los ____ días del mes de _____ del año 2010.

José Antonio Gómez Llano.

Yohairo Benito Consuegra Peña.

Firma del Autor

Firma del Autor

Ing. David Vargas Hernández.

Firma del Tutor

AGRADECIMIENTOS

. Pepex

- Antes que nada agradecerles a mis padres pues a ellos les debo todo lo que hoy soy. Gracias por su amor incondicional, por la educación, y sobre todo por la paciencia...
- A mi hermano, el Dariel, por su cariño y apego y por estar ahí a cada segundo mortificándome, jajaja...
- A mi Roque, decías que la UCI nunca acabaría y ya vez ahorita tienes un jevito Ingeniero. Gracias por todo amore, por la confianza, el amor y el sacrificio de tantos años. Je t'aime, 10000 y 20000.
- A mis abuelos por su eterno cariño y amor.
- A todos los tíos, tías, primos, primas, a mis hermanos, parientes. A la familia en general por la confianza que han depositado en mí durante todos estos años.
- A Yohairo por los 5 años de andanzas y tropiezos, al fin terminamos mi hermano.
- Al los hermanos y hermanas que nos ha dado la UCI y a los que ya venían de años atrás. Al Fonsek, Yusbel, Armin, Peter, los Billy, Father, Bis, y Chorno (Nestor, Ángel y José), Misael, a Maikelon, a Irita, Elide, Annies, la gente de la 5, Alvariño, el Tommy, a Fredduko. Gracias a todos por estar ahí siempre que hizo falta y por los malos y buenos momentos que pasamos, esos fueron los mejores de la vida, los quiero, pórtense bien y no se aflojen que ahora empieza esto.
- A la universidad y a todo el que de una forma u otra a puesto su grano de arena en mi formación personal y profesional.
- A la gente del proyecto, a David, Yanet, Yanay. Al L3@Corporation, a Aldo, Yasel, René, Sulay, Yudi, etc...sin su ayuda esto no hubiera sido imposible sino para Diciembre, jajaja...Gracias por todo su apoyo y confianza.
- Y a todos aquellos que se olvidan en momentos como estos pero que están siempre presentes.

AGRADECIMIENTOS

- Ante todo quiero agradecer a mi mamá y a mi abuela Cachita ya que han hecho todo lo posible para que pueda llegar a graduarme, me han apoyado en todo en la vida y me han sabido dar aliento y confianza para salir de los momentos malos que he pasado, gracias por su amor, gracias por quererme tanto...
- Agradecerle también a mi papá que a pesar de que no vivimos juntos se ha portado muy bien conmigo y me ha ayudado a salir adelante y me ha brindado gran apoyo y me ha servido de gran ejemplo...
- A mi hermana, jejeje, mi hermana que se ha portado muy bien conmigo, y no sé, si pudiera nacer de nuevo si llegara a quererla tanto como ahora, gracias tere...
- A mi novia Iris, que de verdad se ha portado muy bien conmigo y no tengo palabras para agradecer lo que hemos pasado juntos, tanto en las buenas y en las malas, es lo más grande que me ha pasado en la vida y espero que sigamos así...Y agradecerle también a su mamá Estrella por considerarme como su hijo.
- A mi familia en general, a mis tías que están todas locas, jejeje, a mis primos, a mis tíos, gracias a todos por apoyarme y brindarme confianza en todo esta carrera que ya se está terminando....
- A Pepex por lograr una amistad incalculable en estos 5 años que espero que nunca termine, fito ya terminamos. Y también a su mamá Ana, a su hermano Dariel, a su papá y a la Roque por haberse portado tan bien conmigo en estos 5 años...
- A mis hermanos que han venido desde primero y desde años de atrás y se han portado bien conmigo tanto en las buenas como en las malas, Yusbel, Armin, Misael(Mago), Ángel, Nestor, a Elide mi hermana hasta de apellido, a la gente de somalia jejeje, a Yoiner, Katia, Leo, Yuyo, Charly y al Bin que no están aquí pero se han portado en talla, a Yaremís que de verdad se ha portado conmigo tan bien que a veces pienso que es mi mamá, José Manuel que en estos últimos años hemos logrado una buena amistad y espero que siga así, a los colegas del edificio 110.
- Agradecerles también a los colegas y hermanos del barrio que siempre me han apoyado para que pueda terminar esta tesis, Yunier (piloto), a Fernadito, a Yoisel y Yobanis, a Mima y Yuri, a todos porque son muchos.
- No podría dejar de agradecerles a Neyda, Adonis, Holguita, Raiko, al Neto, a Maribel que en estos 5 años en la Habana se han portado bien conmigo y me han ayudado en todo...
- A David, Yanet, Yanay. Al L3@Corporation, a Aldo, Yasel, René, Sulay, Yudi, etc....

DEDICATORIA

Pepex

- Primeramente dedicárselas a mis padres. Se pasó trabajo para llegar pero al final aquí está el resultado, es todo suyo, se lo merecen.
- A mi Roque del alma, que es casi Informática también...
- Al Dariel, estudia para que un día me dediques tu tesis también penquito...
- A mis abuelos por lo tantos y tantos años de entrega y amor.
- Dedicada con todas las de la ley al Yoha por el código y las horas de desvelo y sueño, \$%@&=♀♪# y por lo buenos y buenos momentos que vivimos en la universidad.
- A toda la familia, al final esta tesis es de ustedes también.
- A José Manuel, eterno Billy ancestro y hermano de causa, por los malos y buenos ratos de tesis.
- A todos los hermanos y hermanas de la UCI, esto también es de ustedes.
- En fin, a todas aquellas personas que confiaron en mí, y los que no también, porque no, pues de ellos aprendí lo mejor.

Yohairo

Ante todo dedicarles este trabajo a mis padres y abuela ya que han visto hecho realidad su sueño.

A mi novia Iris por lo que ha luchado conmigo y lo bien que se ha portado.

A mi hermana Tere y a mi hermano Yonlier por lo que siento por ellos.

No podría dejar de dedicarla a mi compañero de Tesis Pepex por todos estos años luchando con las clases, las fiestas, las noches sin dormir y por los momentos buenos hemos pasado en la UCI.

A toda mi familia que es la más loca del mundo pero en los momentos difíciles siempre han estado apoyándome en todo y en lo que haya falta, ustedes se lo merecen.

A nuestro compañero de batalla José Manuel, que de verdad pasamos unas cuantas noches sin dormir por culpa del grails.

A todos los amigos que hice en la Uci, grandes amigos de verdad.

Y a todos aquellos que esperaron esto de mí y de una forma y otra me apoyaron.

RESUMEN

El Sistema Penitenciario Nacional en la actualidad cuenta con un Sistema Automatizado para el Control de Recluso (SACORE) para gestionar toda la información penitenciaria de los individuos que se encuentran cumpliendo sanción en dichos centros. Dicho sistema a pesar de que se ha ido perfeccionando a partir de los requerimientos y solicitudes de los usuarios durante sus siete años de explotación actualmente continua presentando problemas al no estar informatizados en su totalidad muchos de los procesos priorizados, entre los que se encuentran los procesos encargados de registrar y controlar la Situación Legal de los asegurados, acusados y sancionados. Esta situación además de originar un trabajo riguroso y arcaico ocasiona graves problemas legales no solo para los internos sino para todo aquel que de una forma u otra tiene el deber de controlar y gestionar todas las actividades legales en los centros penitenciarios.

El siguiente trabajo consiste en la creación del Diseño e Implementación del módulo Situación Legal del Sistema de Gestión del Interno (SIGI), para apoyar el control de los procesos jurídicos y trámites legales de los internos mediante la integración con los diferentes órganos de justicia. Dando cumplimiento durante el desarrollo del mismo a las tareas y objetivos planteados a partir del análisis de los requerimientos de software establecidos en acuerdo con el cliente y haciendo uso de las tecnologías y herramientas definidas para el proyecto.

Palabras Claves: SIGI, Situación Legal, SACORE

INDICE

| | |
|-------------------------------------------------------------------------|----|
| DECLARACIÓN DE AUTORÍA | I |
| AGRADECIMIENTOS | II |
| DEDICATORIA | IV |
| RESUMEN | V |
| INTRODUCCIÓN | 1 |
| Antecedentes | 1 |
| Situación Problemática | 2 |
| CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA | 6 |
| Introducción | 6 |
| Sistema Penitenciario | 6 |
| Expediente Legal | 6 |
| Sistemas Informáticos en Sistemas Penitenciarios | 7 |
| Sistema de Gestión Penitenciario Venezolano | 8 |
| Sistema Penitenciario Nacional | 10 |
| Herramientas y Tecnologías a utilizar. | 14 |
| Herramientas de modelado | 15 |
| Herramientas de desarrollo | 16 |
| Tecnologías | 18 |
| Conclusiones | 19 |
| CAPÍTULO 2: ARQUITECTURA DEL SISTEMA | 22 |
| Introducción | 22 |
| Arquitectura del Sistema. | 22 |
| Arquitectura en Capas según Grails | 23 |
| Arquitectura Cliente-Servidor | 26 |
| Patrones de Diseño | 26 |
| Diseño de la solución | 29 |
| Descripción de las funcionalidades | 34 |
| Conclusiones | 36 |
| CAPÍTULO 3: DISEÑO E IMPLEMENTACIÓN DE LA SOLUCIÓN | 35 |
| Introducción | 35 |
| Actividades de Diseño e Implementación del módulo Situación Legal | 35 |
| Diseño del Dominio | 35 |
| Diseño del Modelo de Datos | 37 |

| | |
|----------------------------------------------------|----|
| Diseño de la Interfaz de Usuario | 41 |
| Implementación de las clases del dominio. | 42 |
| Implementación de las vistas. | 43 |
| Implementación de los controladores. | 44 |
| Implementación de la capa de acceso a datos. | 44 |
| Implementación de la lógica en el cliente. | 46 |
| Diseño de la funcionalidad Registrar Proceso | 49 |
| Pruebas | 51 |
| Conclusiones | 51 |
| Conclusiones | 50 |
| BIBLIOGRAFÍA | 52 |
| GLOSARIO DE TÉRMINOS..... | 71 |

INTRODUCCIÓN

Antecedentes

Luego del triunfo de la revolución el 1ro de enero de 1959 el gobierno cubano toma un sin número de medidas con el objetivo de revertir muchas de las políticas y leyes que había dejado como legado la tiranía. Una de las medidas que toma en aras de hacer un ordenamiento jurídico y crear nuevas formas para enfrentar los delitos en correspondencia con los perjuicios que éstos producían fue la de renovar el sistema penitenciario existente en aquel entonces. Muchas de las antiguas prisiones heredadas de la tiranía, donde el abuso, las torturas, los tratos inhumanos a los internos, así como la vejación (los cuales constituían los métodos y procedimientos que caracterizaban la estancia de los detenidos en la prisión), fueron desactivadas y se construyó un Sistema Penitenciario totalmente humano, basado en el respeto y en una gestión de control inspirada en la regulación de normas y leyes internacionales en cuanto al tratamiento a los reclusos. Durante estos 51 años de revolución el sistema penitenciario actual ha sufrido un sin número de transformaciones que no solo han posibilitado cumplir con los fines de la sanción penal sino que también ha ayudado a situar a nuestro sistema de justicia entre los de mayor carácter garantista y humanista del mundo, por contar entre sus fortalezas con el perfeccionamiento de la legislación penitenciaria y de su base reglamentaria.

Durante más de 30 años el control en los centros penitenciarios se ha realizado de forma manual, haciendo el trabajo de la gestión legal de los internos de manera ardua y ralentizada, lo que trajo consigo un sin número de situaciones las cuales afectan en gran medida los procesos de trabajo que se ejecutan para el control, tratamiento y atención de los acusados, asegurados y sancionados en los Centros Penitenciarios. Todos estos percances sumados a la introducción de nuevas tecnologías en el país hicieron que fuera de vital importancia la informatización de todas las transformaciones funcionales y organizativas en los sistemas de trabajo para lograr un óptimo aprovechamiento de las mismas en los centros penitenciarios.

Es por eso que en el año 1989 comienza en Cuba el sistema de informatización de los centros penitenciarios, con la automatización de los datos principales del recluso y ciertos aspectos de control penal. Pero no es hasta algunos años más tarde que a raíz del cumplimiento de la orden 43/99 del Vice Ministro Primero se crea un Sistema Automatizado para el Control del Recluso, SACORE por sus siglas, el cual comenzó a dar respuesta en gran medida a muchos de los problemas descritos anteriormente.

Dicho sistema culminó su desarrollo a finales del 2002, poniéndose en marcha a principios del 2003 y cuenta en la actualidad con tres módulos principales: Control Penal, Reeducción Penal y el Orden Interior, los cuales aumentan las especificaciones de la automatización de los datos principales del recluso y los aspectos de control penal existentes, además de adicionar un mayor número de facilidades. En los 7 años de explotación del sistema, a pesar de sus facilidades y ayuda brindada a dichos centros aún cuenta con requisitos incompletos o pendientes de dichos módulos, por no contar la institución con el tiempo para su análisis, diseño y programación, y por carecer de las condiciones tecnológicas para su implantación posterior.

Situación Problemática

Uno de estos requisitos incompletos o aún no modernizados en su totalidad es el referente a la Situación Legal de los internos(as) que se encuentran bajo la custodia del Sistema Penitenciario Nacional. El mismo incluye por su importancia información sobre los procesos judiciales del individuo(a), los trámites legales pendientes a los que puede estar sujeto así como el régimen por el cual transita.

Muchas de estas informaciones, las cuales se encuentran actualmente en ejecución para la gestión y control de la Situación Legal de los internos(as), radican su principal problema en insuficiencias que presentan los procesos priorizados, al no estar estos totalmente completos o informatizados, un ejemplo de ellos son:

- Digitalización del Expediente carcelario y demás documentos originados durante el ingreso y tránsito del interno por el Sistema Penitenciario.
- 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.
- Gestiones legales con los órganos de Justicia y Sistema de Policía.

La existencia de estas insuficiencias en los procesos priorizados, ya sea por falta de organización o informatización, además de la carencia de tecnología de punta en los centros penitenciarios, así como los diferentes problemas de inter conectividad entre los tres niveles de mando no solo conlleva en muchas ocasiones a realizar un trabajo riguroso y arcaico sobre las fuerzas ejecutoras, sino que ocasiona además graves problemas en la toma de decisiones y demás actos legales donde fueran presentadas dichas informaciones como constancia de la Situación Legal de los internos o en los procesos por los cuales el mismo transita (26). Entre los principales problemas que presenta dicho módulo se encuentran:

1. Insuficiente control en la gestión a realizar o realizada por los centros penitenciarios en las soluciones a los trámites legales indefinidos, en correspondencia con los órganos afectados.
2. Digitalización del expediente y otros documentos que se emplean en el control del interno, proceso que conlleva un trabajo con el entorno.

Dadas las situaciones problemáticas anteriores que presenta el Sistema Penitenciario Nacional en lo referente a la Situación Legal en dichos centros, se plantea como **Problema a resolver** en esta investigación:

¿Cuál es el diseño e implementación del módulo Situación Legal del Sistema Penitenciario Nacional a partir de los requerimientos de software establecidos con el cliente y haciendo uso de la arquitectura y las tecnologías definidas por el proyecto?

Definiendo a su vez como **Objeto de estudio**:

Los procesos de desarrollo del módulo Situación Legal del Sistema Penitenciario Nacional.

Y quedando definido como **Campo de acción:**

Los modelos de diseño e implementación del módulo Situación Legal del Sistema Penitenciario Nacional.

Teniendo en cuenta la situación que atraviesa el Sistema Penitenciario Nacional así como las irregularidades de la Situación Legal como gestor de información para conocer la situación jurídica de los internos(as), se plantea como **Objetivo General:**

Desarrollar el módulo Situación Legal del proyecto SIGI a partir del análisis de los requisitos de software establecidos con el cliente y haciendo uso de la arquitectura y las tecnologías definidas por el proyecto.

En aras de dar cumplimiento al objetivo planteado se traza el siguiente conjunto de **Tareas Generales:**

- Estudio de las tecnologías y herramientas a utilizar en el diseño e implementación del módulo Situación Legal, definida en la arquitectura del SIGI.
- Análisis de los requisitos de software y del modelo de negocio correspondientes al módulo Situación Legal.
- Diseño de la solución de software para los requisitos relacionados con la situación legal de los acusados, sancionados o asegurados del Sistema Penitenciario Nacional.
- Implementación del diseño realizado relacionado con la situación legal de los acusados, sancionados o asegurados del Sistema Penitenciario Nacional.

El documento consta de 3 capítulos además de los Anexos y el Glosario de Términos, a continuación se brinda un resumen de los contenidos que abordarán los mismos.

Capítulo 1: Fundamentación Teórica

Abordará conceptos relacionados con los Sistemas Penitenciarios y el tratamiento de la Situación Legal en los mismos. Se analizarán las principales herramientas y tecnologías a utilizar para el desarrollo de los procesos relacionados con el diseño e implementación del módulo Situación Legal del Sistema Penitenciario Nacional.

Capítulo 2: Arquitectura del Sistema

Abordará todo lo referente al diseño propuesto para la solución del software así como la descripción de la arquitectura a utilizar. Se especifican además el conjunto de funcionalidades a desarrollar dando un breve resumen de las mismas.

Capítulo 3: Diseño e Implementación

Abordará las actividades definidas para el diseño e implementación del módulo Situación Legal las cuales darán cumplimiento a las funcionalidades propuestas en el capítulo anterior. Contendrá además los casos de prueba realizados a la funcionalidad Registrar Proceso a un Asegurado.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

Introducción

En el presente capítulo se abordan algunos conceptos relacionados con los Sistemas Penitenciarios y el tratamiento de la Situación Legal en los mismos como generalización de la situación problemática a resolver. Se analizan además las principales herramientas y tecnologías a utilizar para el desarrollo de los procesos relacionados con el diseño e implementación de la Situación Legal de los acusados, sancionados o asegurados del Sistema Penitenciario Nacional.

Sistema Penitenciario

Se define Sistema Penitenciario como el órgano 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. (37)

Cada estado estipula dichas organizaciones basándose en sus concepciones políticas, y adecuándose a las condiciones sociales y económicas de su desarrollo. En nuestro país este sistema está dirigido por la Dirección de Establecimientos Penitenciarios del Ministerio del Interior y 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. A su vez los fundamentos de la política penitenciaria están determinados en la Constitución de la República, La Ley 62, Código Penal, además la organización y las condiciones de ejecución de las sanciones privativas de libertad se corresponden con lo establecido en las "Reglas Mínimas Clásicas para el Tratamiento a los Reclusos" aprobadas el 30 de agosto de 1955 por la Organización de las Naciones Unidas (ONU) y la Declaración Universal de los Derechos del Hombre del 10 de diciembre de 1948. (22)

Expediente Legal

Un expediente se cataloga como una herramienta administrativa utilizada en organismos de gobiernos o administraciones públicas. En cada país su definición difiere ligeramente, aunque mantienen la misma finalidad en todos los casos: reunir la documentación necesaria para sustentar el acto administrativo. Un expediente sin importar su contenido tiene como principal objetivo gestionar y controlar la información de una entidad o individuo, permitiendo a su vez evaluar y administrar dicha información para la toma de decisiones, para un análisis en tiempo real o simplemente para un análisis histórico de la misma.

En los Centros Penitenciarios durante décadas este se ha caracterizado por ser parte de la dinámica generadora de información penitenciaria, la cual está compuesta por la Ficha Penitenciaria, que se genera en cada prisión por la que el interno(a) transite y recoge los datos concretos del mismo, y el Expediente Legal el cual no es más que un conjunto de documentos que se elaboran al ingresar una persona a cumplir una sanción, medida cautelar de prisión provisional y medida reeducativa de seguridad, ordenados metodológicamente con el objetivo de certificar la situación legal y avalar la trayectoria en prisión. Entre los fundamentales documentos está la remisión de ingreso, copia de sentencia, testimonio de liquidación de sanción, decas dactilares y certificaciones de las autoridades penitenciarias. (35)

Cada expediente legal es único, y su principal importancia en estos centros radica en ser el proveedor de toda la situación jurídica de los internos(as) que se encuentran bajo la custodia del Sistema Penitenciario Nacional.

Sistemas Informáticos en Sistemas Penitenciarios

En la actualidad los productos informáticos evolucionan dinámicamente para poder adaptarse a las necesidades tecnológicas del momento y los Centros Penitenciarios al igual que disímiles organizaciones han debido modernizarse para adaptarse a las necesidades tecnológicas del momento. Aspectos como la seguridad, control y evaluación se han visto día a día amenazados por la evolución constante de los sistemas informáticos y el rápido incremento de las tecnologías. Viejos y rigurosos mecanismo de trabajo, así como los arduos procesos de gestión existentes durante décadas en las prisiones hoy en día han sido sustituidos por eficientes soluciones informáticas capaces de gestionar dichos procesos con la máxima seguridad y mínimas dificultades.

El sistema penitenciario del gobierno de Panamá fue creado a principios del año 1997, como resultado de un proyecto financiado por las Naciones Unidas y el Gobierno Español en coordinación con la Dirección General de Sistemas Penitenciarios de Panamá (DGSP). La finalidad del software es mantener almacenado en una base de datos los registros de los internos que están detenidos en los centros penales a nivel nacional. Es un sistema centralizado, con una sede que se enlaza con otras unidades operativas distribuidas en una parte de los centros penitenciarios, con el resto se mantiene un enlace mediante el sistema de actualización por disco de tres y media. (4)

Desde el año 2003, el sistema penitenciario de Buenos Aires, Argentina posee un sistema informático especialmente desarrollado para mantener una base de datos de personas privadas de su libertad. Este sistema que se encuentra implantado en solo una provincia argentina constituye un punto de avance en la recogida de información a individuos puesto que recopila sus datos personales, algunas

incidencias de su expediente judicial y las novedades que lo relacionan. Toda la información recogida es emitida por fax al Registro General de Internos desde las unidades penitenciarias. (4)

Pero si bien dichos Sistemas Informáticos vienen a mejorar y perfeccionar los procesos de control y gestión en los centros penitenciarios, la carencia de los mismos o su indebido manejo pueden en ocasiones traer graves problemas. En Brasil tras una inspección realizada por el Consejo Nacional de Justicia en algunos recintos penitenciarios de cuatro estados brasileños se detectó que alrededor de mil detenidos ya habían cumplido su pena y continuaban encarcelados y otros mil 218 eran privados de beneficios a los que tenían derecho, cifras que representaban el 21 por ciento de los presos. Sólo las libertades concedidas representan economizar un millón 200 mil reales (515 mil dólares). Al final del proceso se dio como principal causa del suceso la carencia de automatización de los datos de los presos así como de su situación jurídica y demás documentos legales. (29)

Sistema de Gestión Penitenciario Venezolano.

SIGEP es un Sistema de Gestión Penitenciaria creado con el fin de automatizar los procesos penitenciarios en la República Bolivariana de Venezuela. El mismo desarrolló una solución de amplio alcance basada en la confección de un expediente carcelario para la automatización entre otros del proceso de Ejecución de la Pena y las Decisiones Judiciales.

En dicha solución un individuo(a) puede tener varios Expedientes Penitenciarios, cada uno de ellos correspondiente al tránsito de este por el Sistema Penitenciario (Figura 1). Cada Expediente Penitenciario puede tener varios procesos judiciales por los cuales el individuo(a) puede estar penado o procesado, un estado el cual puede ser abierto cuando se inicia el proceso, pudiendo cambiar a cerrado por acumulación en caso de ser el proceso por el cual se abre el expediente acumulado en otro proceso, reabierto si se desunen dichos procesos así como en ejecución una vez dictada la sentencia y cerrado cuando en alguna de las fases por las cuales transita el interno(a) se dicte una decisión conclusoria. Las detenciones, las cuales evidencian los períodos que este ha estado detenido fuera del Sistema Penitenciario y la ejecución de la pena correspondiente, la cual tiene como objetivo mantener toda la información relacionada con las penas en ejecución del individuo.

De cada proceso judicial se conocen los delitos, que se clasifican en presunto o probado, los abogados y el Órgano del Ministerio Público que lo atienden, una síntesis del hecho por el que haya sido sentenciado el individuo(a), la jurisdicción del proceso y las etapas por las que transcurre (Anexo 1).

De cada etapa del proceso judicial ya sea Control, Juicio, Apelación, Casación o Revisión se conoce el número de expediente, el Tribunal que atiende el proceso, las fechas relevantes para cada etapa, las decisiones que son emitidas al terminar cada una y el motivo por el que entró a la etapa. (Anexo 2).

Los datos de la ejecución de la pena se registran a nivel de Expediente Penitenciario, puesto que la ejecución de la pena es única para todos los procesos judiciales con sentencia definitivamente firme (Anexo 3). Pero en algunas ocasiones un interno(a) puede estar cumpliendo por más de una pena lo que se refleja en más de una ejecución, para erradicar esta situación se puede proceder a ejecutar una Acumulación de Penas, que no es más que la unificación de las dos ejecuciones en una, conducida por uno de los tribunales de ejecución o por otro. El cómputo se registra como parte de la etapa de Ejecución de la pena y se mantiene un historial de las modificaciones que este ha sufrido durante la ejecución de la pena.

La solución permite además gestionar tanto los documentos asociados a los datos registrados así como gestionar además los distintos abogados encargados de los procesos judiciales por los cuales transita el interno.

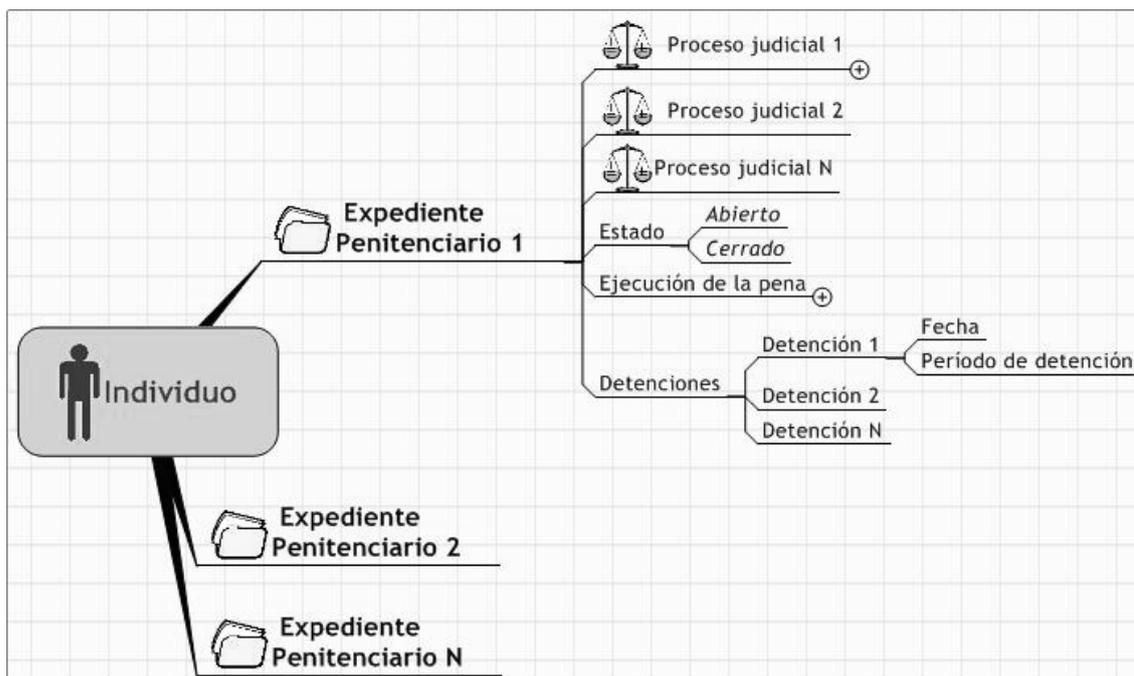


Figura: 1 Expediente Penitenciario (SIGEP).

Dicha solución en sus diferentes versiones permitió mantener la información disponible en todo momento, para los especialistas o Funcionarios de Control Penal (FCP), brindando conocimiento y control sobre las fases por las que transcurre el proceso penal de un interno, las decisiones tomadas sobre cada una y la ejecución de la pena. (10)

Sistema Penitenciario Nacional

Los primeros pasos de informatización de dicho Expediente Legal en nuestro país como gestor de toda la información referente a la Situación Legal de los internos(as) vienen dados por la creación del SACORE, el cual gestiona dicha información principalmente en el subsistema Control Penal. Un Expediente Legal está conformado por documentos legales expedidos por las autoridades competentes que amparan su internamiento, en él se recogen los aspectos más sobresalientes de la trayectoria penal y penitenciaria del interno, que incluyen, clasificación, promociones, beneficios otorgados, revocaciones, movimientos, entre otros. El mismo contiene los documentos que se remiten con el detenido, así como los que se obtienen y confeccionan a su ingreso, durante su permanencia y egreso, lo cual permite el control del acusado, sancionado o asegurado.

Entre las informaciones que el mismo registra se encuentra la referente a la Situación Legal de los internos(as), información que es de vital importancia no solo por ser la encargada de gestionar toda la situación jurídica de los individuos(as) que se encuentran bajo la custodia del Sistema Penitenciario Nacional sino además por estar estrechamente relacionada con todo lo concerniente al ingreso, egreso y traslado de los internos(as).

SACORE en los siete años de explotación con que cuenta se ha ido perfeccionando a partir de los requerimientos y solicitudes de los usuarios, y aunque se ha logrado que el Sistema Informativo existente en los tres niveles de mando sea el resultado del sistema automatizado implantado aún cuenta con funcionalidades no modernizadas en su totalidad como es el caso de dicho Expediente Legal y algunas funcionalidades que engloba como la referente a la Situación Legal, lo que trae consigo que al no contar SACORE con un sistema informatizado en su totalidad capaz de gestionar toda esta información anteriormente citada se afecta y atrasa en muchas ocasiones el análisis oportuno y la toma de decisiones prácticas por parte de los especialistas o funcionarios de control penal en los diferentes procesos o actos legales por los cuales transite el interno(a).

En consecuencia con lo anteriormente visto y teniendo en cuenta la situación que atraviesa el Sistema Penitenciario Nacional (SPN) respecto a las irregularidades existentes en la Situación Legal como gestor de información para conocer la situación jurídica de los internos(as) surge la necesidad de desarrollar un sistema informático capaz de apoyar la informatización y modernización de todos los procesos que se ejecutan en el SPN referentes a la Situación Legal de los internos (as), digitalizando a su vez el Expediente Legal y demás documentos que se emplean en el control del interno en su tránsito por el SPN.

Propuesta del proyecto SIGI

La solución informática se apoyará principalmente en el Sistema Automatizado para el Control del Recluso existente en la actualidad en el Sistema Penitenciario Nacional así como en la solución de software de amplio alcance desarrollada por el proyecto SIGEP para la República Bolivariana de Venezuela la cual a pesar de poseer diferencias con el sistema a desarrollar también tiene un sin número de similitudes además de ser un sistema ya probado y desplegado el cual en sus diferentes versiones a permitido brindar información en todo momento a los especialistas y funcionarios de control penal sobre el proceso por el cual transita un interno y la ejecución de la pena.

Esta solución desarrollada por SIGEP y abordada en epígrafes anteriores a pesar ser una solución informática encargada de controlar y gestionar principalmente los procesos legales en centros penitenciarios al igual que la propuesta que se presenta para el Sistema Penitenciario Nacional ambas cuentan con características diferentes lo que hace imposible tanto la reutilización de requisitos y del negocio como del diseño de la funcionalidades y su código fuente. Entre estas diferencias se encuentra principalmente el hecho de que ambos sistemas son soluciones a la medida, y como tal ambas están exentas de ser soluciones generalizadas para sistemas penitenciarios. Otra característica importante a destacar es el hecho de que ambos sistemas se rigen por las normas y leyes que estipula tanto el estado venezolano como el cubano para los procesos de ejecución de la sanción, las cuales están pactadas en las legislaciones de ambos países, lo que conlleva que al cada estado contar con los fundamentos de la política penitenciaria en base lo descrito en su constitución haga imposible que se puedan reutilizar los requisitos capturados para darle cumplimiento a las necesidades del cliente. Como ejemplo de lo anteriormente citado podemos encontrar la gestión a realizar a la hora de gestionar los datos legales en el expediente penitenciario a un interno una vez que este se encuentre ingresado al sistema. SIGEP para darle cumplimiento a dicha funcionalidad divide el proceso en dos funcionalidades principales: Registrar Proceso y Registrar Ejecución de la Pena. Cada una de estas funcionalidades recoge los principales datos relacionados con el proceso por el que está cumpliendo el interno, dividiendo principalmente la funcionalidad Registrar Proceso en varias fases, de las cuales se recogen datos específicos y sobre las cuales se toman decisiones que rigen en gran medida dicho proceso, de la funcionalidad Registrar Ejecución de la Pena se recoge principalmente el cómputo de la pena, así como la condena y las decisiones tomadas. De ambas funcionalidades se recogen además los documentos asociados a las mismas. En el Sistema Penitenciario Nacional dicho proceso encargado de gestionar toda la información legal de los internos es registrado a su vez también en el expediente penitenciario del mismo, pero no ocurriendo de la misma forma todo el flujo de control encargado de darle cumplimiento a la funcionalidad Registrar Proceso. Debido a que el cumplimiento de esta funcionalidad está regido principalmente por los documentos que le son asociados al interno

según el proceso por el que este transitando en el sistema y por los documentos que en un momento determinado le pueden llegar debido a decisiones o revocaciones emitidas sobre la sanción. Una vez que el interno este ingresado en el sistema se comienza a registrar el proceso según el motivo por el cual haya ingresado y los documentos que al mismo le lleguen como se explicó anteriormente. Una vez registrados dichos documentos se pasa a registrarla los datos al proceso, y se registran los delitos, la responsabilidad civil, las sanciones accesorias y los trámites legales en caso de tenerlos dependiendo todas estas funcionalidades directamente de la llegada o no de los mismos. Otro aspecto que difiere entre ambos sistemas es el referente al proceso de acumulación de causas y de acumulación de penas cuando el interno está cumpliendo por más de una sanción aspecto que es tratado en el Sistema penitenciario Nacional como un proceso de sanción conjunta. SIGEP establece para la acumulación de causas que los procesos por los que cumple se encuentren en la misma fase y que ambos proceso tengan en dicho momento igual estado. En el caso de la acumulación de penas ambos procesos se unifican dando como resultado un solo computo y una condena y quedando como órgano encargado del proceso uno de los tribunales de los anteriores procesos o un nuevo tribunal que se dicte. En el Sistema Penitenciario Nacional este aspecto es tratado como una conjunta y solo se aprueba cuando le llega el documento asociado a dicho proceso, registrando a su vez los datos del mismo.

Otros de los aspectos de menor peso pero que también difieren en la reutilización de requisitos, del negocio, del diseño de las funcionalidades así como de su código fuente puesto que en cierto momento discrepan con toda la lógica del negocio por muchos de los factores antes expuestos son las distintas clasificaciones, tanto de internos, como de procesos y delitos, las distintas clasificaciones en regímenes y los permisos o aprobaciones de cambio de los mismos así como el otorgamiento de beneficios.

La propuesta de solución que presenta SIGI estará enmarcada en el subsistema Registro Legal. El Expediente Legal tendrá como uno de sus acápite Situación Legal, el mismo contendrá el proceso o los procesos por el cual el interno transita, registrando de cada uno de estos el estado, los documentos del mismo, el delito, la sanción, los órganos encargados del proceso, así como una breve reseña del delito y las sanciones accesorias. De cada proceso se recogerán los principales datos del mismo, fechas de inicio de sanción, preventivas, de cumplimiento, así como la causa del proceso, el tribunal, el número de denuncia y el expediente de fase preparatoria (EFP) entre otros. Los estados de cada proceso van a estar dados según el cumplimiento de estos, pudiendo variar en dependencia de la cantidad de procesos por los cual el interno transite así como de las decisiones tomadas sobre cada uno de ellos. El estado de un proceso puede variar desde Por Cumplir hasta Cumplido, pasando por

Cumpliendo en caso de que el proceso este en ejecución, Interrumpido al ser detenido la ejecución del proceso y pudiendo llegar a Conjunta por la unificación de procesos.

Los documentos serán los encargados de gestionar el control de todos los procesos, al contener estas las constancias de cada aprobación y decisión tomada, los mismos van a ser los responsables de gestionar todos los procesos internos de trámites legales pendientes así como los beneficios otorgados a cada interno. Se recogerán además los Delitos de los acusados y sancionados así como los datos de cada proceso y una reseña de los mismos. Del acápite Responsabilidad Civil se recogerá el tipo de responsabilidad, el estado en que se encuentra la misma, la persona(s) afectada(s), así como la cuantía de la misma. De las Sanciones Accesorias se recogerá un listado de las mismas así como la duración de estas. (Ver Anexo 4).

Otra información importante tratado en acápite Situación Legal será el de los Trámites Legales Pendientes, los mismos se generarán según los documentos que le lleguen a cada proceso por el cual el interno ingresa o reingresa al sistema, así como excepcionalmente cuando se crea una sanción conjunta por más de un proceso. De cada Trámite Legal se recoge además la fecha en que se generó y la fecha de culminación del trámite, así como otros datos específicos según el trámite que se le haya generado al interno. (Ver Anexo 5). El Régimen estará a su vez incluido como otro acápite y va a gestionar principalmente los diferentes regímenes por los cuales puede transitar un interno una vez sancionado.



Figura: 2 Expediente Penitenciario (SIGI).

El sistema una vez registrado los documentos y el proceso por el cual cumple un interno(a) y el cual coincide en este caso con el proceso por el cual ingresa a prisión debe permitir a su vez actualizar los datos de dicho proceso en caso de llegarle posteriormente algún documento al interno(a) o de registrarle otro nuevo proceso al interno(a) en caso de haber comenzado a cumplir por otro proceso diferente al que ingreso a prisión. Una vez que el interno este cumpliendo por más de un proceso el sistema debe permitir a su vez unificar esos procesos en uno, cambiando el estado del proceso a Conjunta y permitiendo a su vez recoger los datos del nuevo proceso que se creó.

Herramientas y Tecnologías a utilizar.

Como metodología de referencia para guiar el trabajo de implementación, se utilizó lo que propone Rational Unified Process (RUP) para las disciplinas de diseño e implementación.

RUP es un proceso de desarrollo de software, un conjunto de metodologías adaptables al contexto y necesidades de cada organización. RUP divide el proceso en cuatro fases, dentro de las cuales se realizan varias iteraciones en número variable según el proyecto, lo que pone de manifiesto una de sus características principales, la de ser iterativo e incremental, además los casos de uso describen los

requisitos funcionales del sistema desde la perspectiva del usuario lo que hace que sea guiado por los casos de uso y además es centrado en la arquitectura pues esta permite ganar control sobre el proyecto para manejar su complejidad y controlar su integridad. (28)

En la disciplina de análisis y diseño RUP propone traducir los requisitos a una especificación que describa cómo implementar el sistema, haciéndola suficiente para que el sistema pueda ser implementado sin ambigüedades. Aquí se debe refinar también la arquitectura hasta llegar a su forma definitiva. Entre los principales artefactos que deben ser generados a lo largo de esta fase encontramos el modelo de datos, la descripción de la arquitectura, el modelo de análisis y diseño así como el modelo de despliegue.

En la disciplina de implementación se busca de manera específica planificar las integraciones del sistema necesarias en cada iteración, distribuir el sistema asignando componentes ejecutables a nodos en el diagrama de despliegue así como implementar las clases y subsistemas encontrados durante el diseño entre otros. Debe además probar los componentes individualmente como parte del flujo de prueba y luego integrar los resultados en un sistema ejecutable. Estos resultados se obtienen como parte de los artefactos, modelo de implementación, modelo de despliegue, e implementación del subsistema. Al final de esta fase se debe obtener una versión estable del software a construir capaz de satisfacer las necesidades del cliente.

Herramientas de modelado

Las herramientas de modelado de software utilizadas para generar y representar los artefactos del sistema a realizar son:

•**ER/Studio:** Es una herramienta de modelado de datos para el diseño y construcción de bases de datos a nivel físico y lógico. ER/Studio está equipado para crear y manejar diseños de bases de datos funcionales y confiables. Ofrece fuertes capacidades de diseño lógico, sincronización bidireccional de los diseños físicos y lógicos, construcción automática de bases de datos, documentación y fácil creación de reportes.

ER/Studio ofrece las siguientes funcionalidades:

- Capacidad fuerte en el diseño lógico.
- Sincronización bidireccional de los diseños lógico y físico.
- Construcción automática de Base de Datos.
- Reingeniería inversa de Base de Datos.

- Documentación basada en HTML.
- Un Repositorio para el modelado.

ER/Studio está enfocado a ayudar a tomar decisiones en cómo resolver embotellamientos de los datos, elimina redundancia y alcanza en última instancia usos de más alta calidad que entreguen datos más eficientes y exactos a la empresa. (7)

•**Visual Paradigm Suite 3.1:** Es una suite completa de herramientas CASE, para realizar el modelado de los procesos a desarrollar. Tiene como facilidades la de brindar productos de calidad al soporta aplicaciones web. Permite además generación de código para Java y exportación como HTML así como la compatibilidad entre ediciones. Permite la generación de todos los diagramas generados por la metodología RUP en sus diferentes fases. Su suite comprende además el Visual Paradigm For UML 6.1, es una herramienta de diseño que soporta todos los diagramas UML, diagramas de SysML y el diagrama entidad-relación, además soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. El software de modelado UML ayuda a una más rápida construcción de aplicaciones de calidad, mejores y a un menor coste. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación.

El Visual Paradigm permite a su vez a los desarrolladores poder diseñar la documentación del sistema con plantillas de diseño. Y a los analistas de sistemas poder estimar las consecuencias de los cambios con los diagramas de análisis de impacto, como la matriz y el diagrama de análisis. (36)

Herramientas de desarrollo

Las herramientas de desarrollo utilizadas para llevar a cabo el desarrollo de las funcionalidades son:

NetBeans 6.8 Milestone2: El NetBeans IDE es un IDE de código abierto escrito completamente en Java usando la plataforma NetBeans. El NetBeans IDE soporta el desarrollo de todos los tipos de aplicación Java (J2SE, web y aplicaciones móviles). Entre sus características se encuentra un sistema de proyectos basado en Ant, control de versiones y refactoring. Cuenta con un compilador de Java interno y un modelo completo de los archivos fuente de Java. Esto permite técnicas avanzadas de refactorización y análisis de código. El IDE también hace uso de un espacio de trabajo, en este caso un grupo de metadata en un espacio para archivos plano, permitiendo modificaciones externas a los archivos en tanto se refresque el espacio de trabajo correspondiente, emplea módulos (en inglés plugin) para proporcionar toda su funcionalidad. Puede extenderse usando otros lenguajes de

programación como son C/C++ y Python y trae incluido entre sus novedades más destacadas el uso de Groovy. (25)

- **Contenedor Web (Apache Tomcat 6.0):** Apache Tomcat es un contenedor Web escrito en Java, por lo que funciona en cualquier sistema operativo que disponga de una máquina virtual Java y desarrollado en un ambiente participativo y abierto. Apache Tomcat es usado en numerosas aplicaciones web de gran escala y críticas en diversas industrias y organizaciones que se referencian en su sitio oficial. (24)

Subversión: Subversión es un software de sistema de control de versiones. Está desarrollado sobre software libre bajo una licencia de tipo Apache/BSD y se le conoce también como svn por ser ese el nombre de la herramienta de línea de comandos. Una característica importante de Subversión es que los archivos versionados no tienen cada uno un número de revisión independiente. En cambio, todo el repositorio tiene un único número de versión que identifica un estado común de todos los archivos del repositorio en cierto punto del tiempo. El acceso al repositorio es mediante la red, lo que le permite ser usado por personas que se encuentran en distintos ordenadores.

Su principal importancia radica en que varias personas pueden modificar y administrar el mismo conjunto de datos desde sus respectivas ubicaciones fomentando así la colaboración. Permite además integrar varias interfaces a entornos de desarrollo, un ejemplo de esto son el TortoiseSVN el cual provee integración con el explorador de Windows. Y como carencia o desventaja tenemos que el manejo de cambio de nombres de archivos no es completo, pues este es manejado como la suma de una operación de copia y una de borrado. (23)

Sistema Gestor de Base de Datos

- **Oracle Database 11g:** Oracle es un sistema de gestión de base de datos relacional (o RDBMS por el acrónimo en inglés de Relational Data Base Management System), desarrollado por Oracle Corporation. Está considerado como uno de los sistemas de bases de datos más completos, destacando entre sus principales fortalezas el soporte de transacciones, la estabilidad, escalabilidad y el soporte multiplataforma.

Como sistema gestor de datos relacional está orientado al acceso remoto y redes (internet). Hoy por hoy Oracle se puede implementar en diferentes plataformas: Familia de Microsoft, Unix, Linux, entre otras. Las arquitecturas en las que se asienta pueden ser: Intel, Alpha, Sparc,... a nivel de procesadores y es perfectamente configurable en entornos "OLTP", paralelos, Cluster, e incluso resulta

una genial solución a nivel de Datawarehouse y CRM. Oracle, se puede implementar en cliente/servidor con muchas arquitecturas de red, como TCP e IPX/SPX. Presenta una interface de última tecnología basado en Java y XML, así como un servidor adjunto de aplicaciones para internet, email, seguridad de datos, etc. Tiene además un conjunto de soluciones para la seguridad y rechazo a usuarios "no deseados" en el sistema, así como una estabilidad y escalabilidad a prueba de bombas: "Cluster". Su instalación administración, configuración e inclusive la implementación de herramientas visuales, en un entorno Windows se ha conseguido simplificar al máximo para conseguir dar un servicio óptimo a los usuarios. Entre sus principales desventajas se encuentran que necesita de mucha memoria RAM debido al interface gráfico desarrollado en Java así como el alto costo de su licencia en el mercado. (22)

Tecnologías

Las tecnologías utilizadas para el desarrollo de las funcionalidades son:

- **JSON: (Javascript Object Notation)** formato ligero para intercambio de datos. JSON, acrónimo de JavaScript Object Notation, es un formato ligero para el intercambio de datos. JSON es un subconjunto de la notación literal de objetos de JavaScript que no requiere el uso de XML. La simplicidad de JSON ha dado lugar a la generalización de su uso, especialmente como alternativa a XML en AJAX. Una de las supuestas ventajas de JSON sobre XML como formato de intercambio de datos en este contexto es que es mucho más sencillo escribir un analizador semántico de JSON. (13)

- **Javascript:** Lenguaje script utilizado para unir el conjunto de tecnologías usados en la web. JavaScript es un lenguaje de scripting basado en objetos, utilizado para acceder a objetos en aplicaciones. Principalmente, se utiliza integrado en un navegador web permitiendo el desarrollo de interfaces de usuario mejoradas y páginas web dinámicas. JavaScript es un dialecto de ECMAScript y se caracteriza por ser un lenguaje basado en prototipos, con entrada dinámica y con funciones de primera clase. Todos los navegadores modernos interpretan el código JavaScript integrado dentro de las páginas web. Para interactuar con una página web se provee al lenguaje JavaScript de una implementación del DOM. Su principal importancia radica en que tradicionalmente, se venía utilizando en páginas web HTML, para realizar operaciones y en el marco de la aplicación cliente, sin acceso a funciones del servidor. JavaScript se ejecuta en el agente de usuario, al mismo tiempo que las sentencias van descargándose junto con el código HTML. (32)

- **Dojo Toolkit:** Librería de clases Javascript. Dojo es un frameworks que contiene Apis y widgets (controles) para facilitar el desarrollo de aplicaciones Web que utilicen tecnología AJAX. Contiene un

sistema de empaquetado inteligente, los efectos de UI, drag and drop Apis, widget Apis, abstracción de eventos, almacenamiento de Apis en el cliente, e interacción de Apis con AJAX. Resuelve asuntos de usabilidad comunes como pueden ser la navegación y detección del navegador, soportar cambios de URL en la barra de URLs para luego regresar a ellas, y la habilidad de degradar cuando AJAX/JavaScript no es completamente soportado en el cliente. Es conocido como "la navaja suiza del ejército de las bibliotecas Javascript". Proporciona una gama más amplia de opciones en una sola biblioteca JavaScript y es compatible con navegadores antiguos. (33)

- **Gant** es una herramienta para las tareas de secuencias de comandos Ant con Groovy en vez de XML para especificar la lógica. Una especificación de Gant es un script Groovy y así puede traer todo el poder de Groovy que soportar directamente, algo que no es posible con scripts Ant. Si bien puede ser visto como un competidor de la hormiga, Gant utiliza tareas Ant para muchas de las acciones, pero Gant es realmente una forma alternativa de hacer las cosas usando Ant, pero usando un lenguaje de programación en lugar de XML para especificar las reglas. (1)

- **GRAG** es un generador de objetos de dominio para el frameworks de desarrollo Grails. GRAG permite la ingeniería inversa una vez definida y establecida una base de datos así como generar el dominio de objetos de Grails. El mismo genera los objetos de dominios conteniendo estos todas las restricciones y relaciones existentes. (1)

Conclusiones

En este capítulo se realizó un estudio más detallado de todo lo concerniente con el Sistema Penitenciario Nacional lo que permitió comprobar y analizar el estado actual del mismo en todo lo referente a la Situación Legal de los asegurados, acusados y sancionados que transitan por el Sistema Penitenciario Nacional. Se estudiaron también algunos sistemas penitenciarios extranjeros relacionados con los procesos existentes para el control y gestión de los procesos legales de un interno, entre el que se destaca SIGEP por su cercanía y experiencia en tal rama, permitiendo comprobar el flujo de trabajo y el control y gestión de los mismos en relación con los procesos judiciales que registran. Se abordaron además todas las metodologías, herramientas y tecnologías propuestas por el proyecto SIGI para la creación de la solución del módulo Situación Legal.

CAPÍTULO 2: ARQUITECTURA DEL SISTEMA

Introducción

El presente capítulo abordará principalmente el diseño propuesto para la solución del software así como la descripción de la arquitectura a usar en el proyecto SIGI, sobre la cual se va a enmarcar a su vez el desarrollo del módulo Situación Legal. Se dará una breve descripción de sus componentes así como un análisis de los mismos.

Arquitectura del Sistema.

La arquitectura de software se define como “la organización fundamental de un sistema encarnada en sus componentes, las relaciones entre ellos y el ambiente y los principios que orientan su diseño y evolución” (IEEE 2000). O como la organización o estructura (en un punto dado en el tiempo) de los componentes importantes que interactúan a través de interfaces, los componentes que se compone de sucesivos componentes más pequeños y las interfaces. Una arquitectura puede ser descompuesta recursivamente en partes que interactúan a través de interfaces, las relaciones que conectan a las partes, y las limitaciones para el montaje de piezas. Partes que interactúan a través de interfaces incluyen clases, componentes y subsistemas. (27)

La arquitectura definida para la construcción del Sistema de Gestión Penitenciaria de la República de Cuba (SIGI) está basada principalmente en las funcionalidades y facilidades que propone el frameworks de desarrollo Grails, este más que un frameworks, es una plataforma para desarrollo de aplicaciones web basada en el lenguaje de tipado dinámico Groovy, se compila y ejecuta en una JVM (Java Virtual Machine). Se integra con el lenguaje Java en cualquier nivel e incluye componentes que van desde los flujos webs hasta la capa de acceso a datos. Grails está basado principalmente en herramientas poderosas como: Spring, Hibernate, Sitemesh, Gant, 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” y que han mejorado la manera de desarrollar en Java, por tanto representa el siguiente nivel de abstracción para JEE. (1) La arquitectura del sistema se basará a su vez en el estilo arquitectónico Cliente–Servidor y la propuesta de Arquitectura en Capas que propone Grails.



Figura: 3 Arquitectura basada en Grails.

Arquitectura en Capas según Grails.

La arquitectura de Grails está conformada por 3 capas lógicas principales: Web Layer, Service Layer, y Data Layer, donde cada capa está separada de la siguiente e interactúan mediante interfaces que definen funcionalidades que la misma debe brindar o también llamadas fachadas las cuales aseguran que el acoplamiento sea el más bajo posible y la abstracción del funcionamiento de la capa inferior, sea casi total. Cada capa proporciona servicios a la capa inmediatamente superior y se sirve de las prestaciones que le brinda la inmediatamente inferior. Esta arquitectura en capas por su diseño proporciona la facilidad de modificar cada capa todo lo posible sin infligir daños o alteraciones a la capa inmediata, a continuación se brinda una descripción de las mismas:

Web Layer: En esta capa se encuentran las Vistas y la Lógica de Presentación, las cuales según la arquitectura que propone Grails estarán enmarcadas sus clases en los paquetes Controllers, el cual contendrá las clases controladoras y View and Layouts donde se encontrarán las Groovy Server Pages o más conocidas como GSP. En la Lógica de Presentación se manejará todo el flujo web utilizando la implementación del patrón Modelo Vista Controlador que nos brinda Grails mediante Spring MVC. Grails implementa el patrón MVC, en el que la lógica empresarial se separa de la presentación de la aplicación. Esto permite cambiar fácilmente el aspecto de la aplicación, sin modificar su comportamiento. La capa de presentación se compone principalmente de: modelo, vistas, controladores.

Controlador: Un controlador de Grails es una clase responsable por el manejo de los pedidos provenientes de la aplicación. El controlador recibe la petición, realiza algún trabajo potencial con la misma y finalmente decide que sucederá a continuación, lo cual puede incluir alguna de los siguientes flujos.

- Ejecutar otra función de controlador.
- Mostrar una vista.
- Mostrar información directamente con la respuesta de la petición.

Un controlador es prototipado, lo cual significa que una nueva instancia es creada por cada petición, por tanto los desarrolladores no necesitan manejarlos en modo “singleton”. Proveen la entrada principal para cualquier aplicación de Grails, coordinando los pedidos entrantes, delegando hacia los servicios o clases de dominio para la lógica de negocios y renderizando las vistas.

Modelo: Una de las actividades fundamentales llevadas a cabo por los controladores es obtener los datos que serán mostrados en la vista. El controlador puede recoger esta información directamente, delegarla a algún servicio u otro componente comprendido en la capa de acceso a datos esta información es pasada a la vista en forma de un mapa u objeto de información. Dicho objeto representa el modelo.

Vista: Grails utiliza para la interacción con el usuario, la tecnología más popular y robusta que ofrece la empresa Sun Microsystems en contextos de presentación web, es decir JSP. Pero basada en una implementación mediante GSP, que es una extensión de JSP y puede incluir Groovy. El mismo permite a los desarrolladores mezclar etiquetas de lenguajes de marcas tradicionales como HTML con código Java para producir vistas dinámicas. 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. Las misma están representadas en el paquete de clase View and Layouts.

- **Service Layer:** En esta capa se 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. Sus clases radicarán según la arquitectura propuesta por Grails en el paquete Services. 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.
- **Data Layer:** Maneja los objetos de acceso a datos abstrayéndolos del mecanismo de persistencia usado; a través de interfaces que exponen las operaciones de persistencia. Grails para evitar trabajar directamente con un gestor de base de datos y sus tablas y permitir trabajar con objetos en su lugar utiliza Hibernate 3 como una herramienta ORM pero esta vez dada la naturaleza dinámica de Grails y la adopción del convenio sobre la configuración crea sobre una versión superior de una nueva implementación de Hibernate llamado Grails objeto mapeo relacional (GORM) que simplifica el trabajo con Hibernate y elimina cualquier configuración externa.

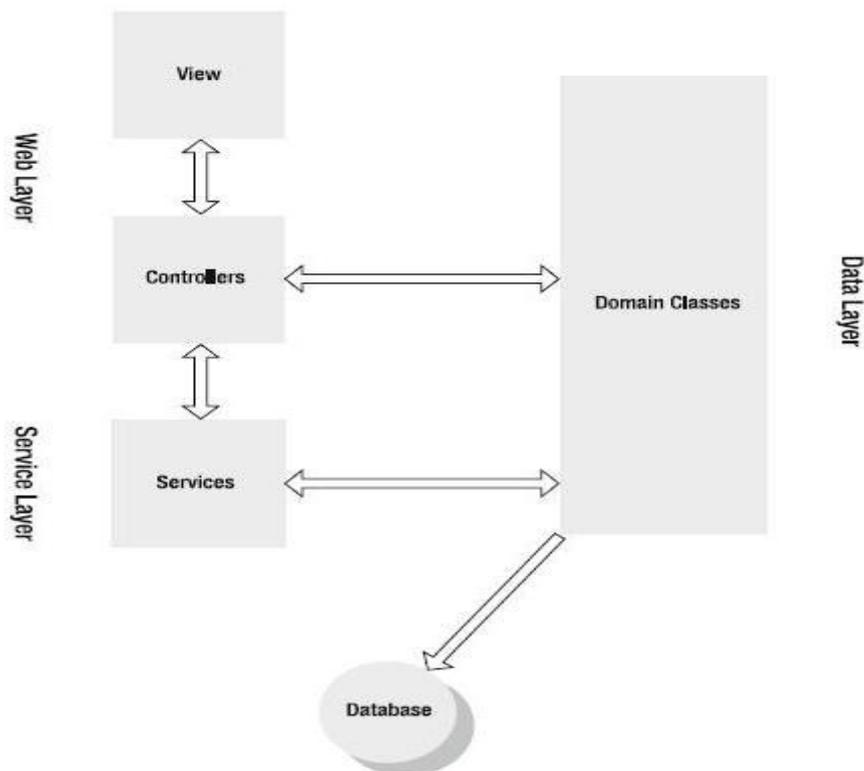


Figura: 4 Arquitectura en Capas según Grails.

Arquitectura Cliente-Servidor

La arquitectura cliente servidor se encuentra dentro de la clasificación de estilo de llamada y retorno, donde dos tipos de aplicaciones se encuentran ejecutándose de forma independiente, una de estas aplicaciones se ejecuta como cliente y la otra como servidor. El cliente y el servidor generalmente están localizados en diferentes sistemas, sin embargo pueden encontrarse en el mismo. El cliente es la entidad que hace la petición por un servicio. El servidor es la entidad que provee el servicio correspondiente a la petición. El servicio debe procurar el resultado, el cual es retornado. Esta arquitectura posibilita no solo una mayor escalabilidad y una manera fácil de hacerla sino que además proporciona un reparto de carga adecuado para dichos sistemas. Por parte de la aplicación cliente hay una menor complejidad en los procesos así como una menor ocupación de la memoria y la aplicación Servidor permite a su vez un menor tráfico en la red.

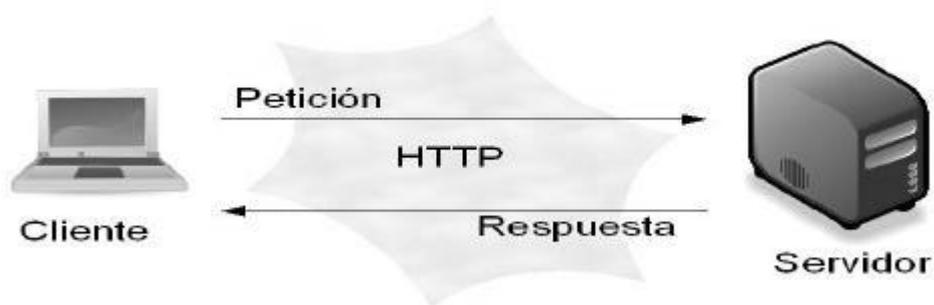


Figura: 5 Arquitectura Cliente-Servidor

Patrones de Diseño

Los patrones de diseño son el esqueleto de las soluciones a problemas comunes en el desarrollo de software, brindan una solución ya probada y documentada a problemas de desarrollo de software que están sujetos a contextos similares. Estos se caracterizan por estar conformados por un conjunto de elementos, como: su nombre, el problema (cuando aplicar un patrón), la solución (descripción abstracta del problema) y las consecuencias (costos y beneficios). (17)

En el diseño y la implementación realizada del módulo Situación Legal se utilizaron diferentes patrones de diseño para brindar y ayudar a desarrollar una solución más robusta y escalable.

De los patrones que propone el grupo de los patrones GoF o también conocidos Gang of Four se utilizó de forma general el Singleton y quedando como propuesta a su vez el uso de patrón Facade. Este último está comprendido entre los patrones de creación y se propone utilizar para separar las clases relacionadas con la lógica de negocio con las clases de la lógica de presentación, evitando así las constantes peticiones al servidor y permitiendo a su vez estructurar las llamadas o peticiones que se realizaran desde el cliente. En el caso de que se decida cambiar la implementación de las clases no es necesario que la capa superior se entere de tal situación, sencillamente la fachada no permitiría que esto ocurra, al no existir una dependencia directa entre las capas.

De los patrones Grasp se utilizaron de forma general los patrones Alta Cohesión, Bajo Acoplamiento, Controller y Front-Controller.

El patrón Controller propone asignar la responsabilidad de controlar el flujo de eventos de un sistema, a clases específicas llamadas controladores. Los controladores no ejecutan las tareas sino que las delegan en otras clases, con las que mantiene un modelo de alta cohesión. Cada clase de Grails promueve el uso de este patrón como parte de los convenios del frameworks. A su vez el patrón Front-Controllers propone utilizar un controlador como el punto inicial de contacto para manejar las peticiones del usuario en una aplicación.

El controlador maneja el control de peticiones, incluyendo la invocación de los servicios de seguridad como la autenticación y autorización, la elección de una vista apropiada, el manejo de errores, y el control de la selección de estrategias de creación de contenido. La clase **procesoControllers** es la encargada de gestionar dicho controlador en el sistema.

Grails, como frameworks de aplicaciones web y con la visión de convertirse en un marco de trabajo altamente productivo no puede estar ajeno al uso de patrones por lo que utiliza como principales paradigmas en este sector dos patrones fundamentales: Convención sobre Configuración o también conocidos por sus siglas en Ingles como Convention over Configuration y DRY, más conocido también como Don't Repeat Yourself.

Convention over Configuration no es más que un paradigma de programación de software que busca decrementar el número de decisiones que un desarrollador necesita hacer, ganando así en simplicidad pero no perdiendo flexibilidad por ello y DRY es una filosofía de definición de procesos que promueve la reducción de la duplicación. Ambos patrones en general proporcionan un entorno de desarrollo estandarizado y ocultan en gran parte detalles de configuración. (9)

Con la utilización del frameworks Spring como componente de Grails se utilizan diferentes patrones de dicho frameworks que aportan un mejor diseño e implementación al sistema. Un ejemplo de esto es la utilización del patrón Modelo Vista Controlador. Spring-MVC es uno de los módulos del Framework de Spring, y como su nombre indica implementa una arquitectura Modelo - Vista - Controlador, explicada anteriormente, que se utilizará como base para desarrollar la capa de presentación de la aplicación. (13)

Diseño de la solución

El Sistema de Gestión Penitenciario Cubano (SIGI) estará constituido por un conjunto de sistemas y subsistemas estructurados según las áreas que conforman la actividad penitenciaria. En los subsistemas se agruparan todo los módulos relacionados con un objetivo a fin, permitiendo un mejor aprovechamiento, reutilización y acomodamiento de los mismos. Los módulos serán por su parte los encargados de agrupar todas las funcionalidades comunes, dando cumplimiento en ellos a los requisitos planteados. La Situación Legal estará comprendida en un módulo del mismo nombre: "módulo Situación Legal" y estará comprendido en el subsistema Registro Legal el cual permitirá administrar los aspectos relacionados con la situación legal de internos y el control del cumplimiento de la sanción o medida de seguridad reeducativa de internamiento durante su tránsito por el sistema penitenciario. Dicho módulo estará estrechamente relacionado con el módulo Ingreso (Figura 6) por ser este último la principal entrada a los procesos de la Situación legal de los internos(as).

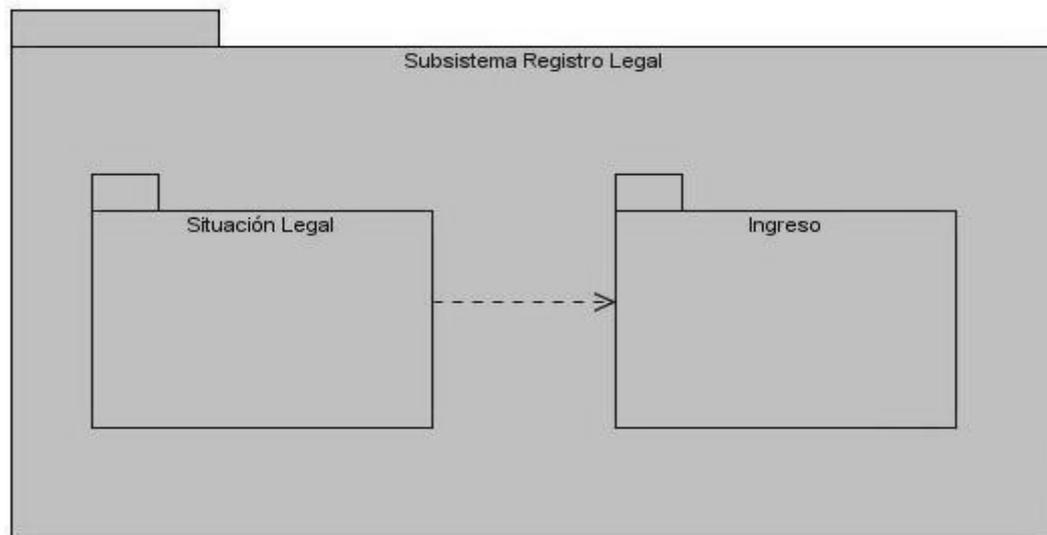


Figura: 6 Diagrama de Módulos

Para una mejor comprensión de la arquitectura propuesta y de la implementación el módulo Situación Legal se dividió en varios paquetes los cuales organizan los diferentes casos de uso según las funcionalidades propuestas para lograr un correcto funcionamiento del sistema como se muestra en la **Figura 7**. Dicha estructura por las características de Grails es común para la jerarquía de paquetes que el mismo propone, teniendo en cada paquete de Grails, dígame: Controllers, Views and Layouts, y Domain Class, etc. una estructura semejante. Entre los paquetes definidos se encuentran:

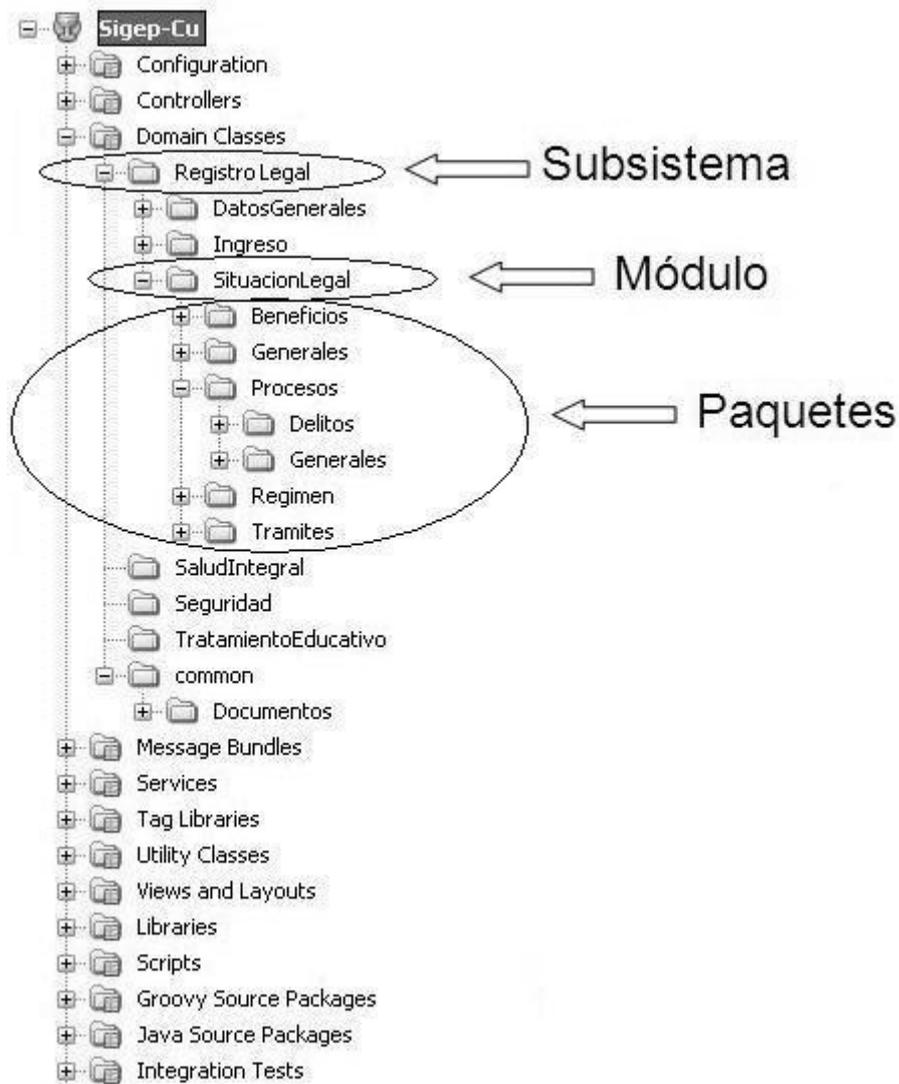


Figura: 7 Estructura Módulo Situación Legal

El paquete **Generales** contará con las clases encargadas de gestionar los datos principales con que cuenta un interno(a) y que están estrechamente relacionados al módulo Ingreso. Estos datos son la principal entrada en el control de los procesos de la Situación Legal de los internos(as) así como los encargados de contener las principales evaluaciones y valoraciones de los mismos.

El paquete **Documentos** contendrá todas las funcionalidades referentes a los documentos con que ingresa un interno(a) y los que en un momento determinado del proceso le pueden ir llegando. Dichos documentos rigen todo el proceso de la Situación Legal de los internos(as) durante su tránsito por el Sistema Penitenciario.

El paquete **Procesos** es el encargado agrupar todas las clases relacionadas con los procesos del interno, desde un proceso por el que ingresa a prisión hasta por uno que no cumple. El objetivo de las mismas en su totalidad es el de registrar mediante sus clases la información sobre los procesos judiciales que le han sido abiertos a un interno. De cada proceso se pueden registrar los delitos que le son imputados al interno, la fiscalía, tribunal y demás órganos encargados del cumplimiento del proceso. En cada una de ellas se toman decisiones judiciales que definen la continuidad o cierre del proceso penal. Cuando sobre un proceso se toma una decisión de sentencia firme, el interno pasa de acusado a sancionado / de asegurado a sancionado / de sancionado a asegurado / de asegurado a acusado y de sancionado a acusado y comienza la ejecución de la pena.

El paquete **Régimen** permitirá registrar y controlar mediante el agrupamiento de las clases la clasificación en régimen dada a cada interno, la fecha en que se le otorgó así como mantener un histórico de las clasificaciones que ha tenido y los motivos por los cuales ha cambiado de uno a otro régimen.

El paquete **Trámites Legales Pendientes** agrupara las clases encargadas de gestionar todos los trámites legales que se pueden generar a un interno una vez que este ya esté cumpliendo por un proceso o que comience a cumplir por otro proceso diferente por el que ingreso a prisión.

Se definieron a su vez las clases y componentes que estarán comprendidas en los paquetes **common** y que serán las encargadas de gestionar todas aquellas funcionalidades comunes tanto para los módulos como subsistemas lo que hace más flexible y reutilizable el código, tanto para las clases del dominio como para las de acceso a datos y las que gestionan la lógica de negocio.

Descripción de las funcionalidades

A continuación se definen las principales funcionalidades relacionadas con el módulo Situación Legal y que estarán contenidas en los paquetes anteriormente descritos. De las mismas se dará una breve descripción así como su nombre.

| Nombre | Descripción |
|-------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------|
| Registrar Proceso | Permite registrar el proceso por el que cumple un interno(a) o cualquier otro proceso que le llegue al interno, según el motivo de ingreso del mismo. |

| Nombre | Descripción |
|-----------------------------------|-----------------------------------------------------------------------------------------------|
| Registrar documentos del proceso. | Permite registrar los documentos que le van llegando a un interno por un proceso determinado. |

| Nombre | Descripción |
|-------------------------------|----------------------------------------------------------------------------------------------------------------------|
| Registrar datos de un proceso | Permite registrar los datos de un proceso determinado dependiendo de la documentación que le llegue por ese proceso. |

| Nombre | Descripción |
|-----------------------------|--------------------------------------------------------------------------------------------------------------------------------|
| Registrar datos de delitos. | Permite registrar los datos de uno o más delito en dependencia de los delitos que haya cometido el interno en caso de existir. |

| Nombre | Descripción |
|---------------------------------|-------------------------------------------------------------------------------------|
| Registrar sanciones accesorias. | Permite registrar las sanciones accesorias imputadas al interno en caso de existir. |

| Nombre | Descripción |
|-------------------------------------------|----------------------------------------------------------------------------------------------------------|
| Registrar datos de responsabilidad civil. | Permite registrarle al interno las responsabilidades civiles asociadas a cada delito en caso de existir. |

| Nombre | Descripción |
|------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Registrar trámite legal pendiente. | Permite registrar los trámites legales pendientes asociados a un interno en caso de existir. Además el sistema debe permitir notificar un trámite legal pendiente al interno en dependencia del estado en que se encuentre(n) el(os) proceso(s) del interno(a). |

| Nombre | Descripción |
|------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Asignar estatus legal. | El sistema deberá clasificar al interno en sancionado, acusado o asegurado en dependencia de los datos de su situación legal y los cambios que pueda tener el interno. |

| Nombre | Descripción |
|--------------------|---------------------------------------------------------------------------------------------|
| Asignar categoría. | El sistema deberá categorizar al interno según las categorías especiales, en caso de tener. |

| Nombre | Descripción |
|------------------------|---------------------------------------------------------------------------------|
| Asignar clasificación. | El sistema deberá clasificar al interno en: Militar, militar común, común, etc. |

| Nombre | Descripción |
|-----------------------|----------------------------------------------------------------------------------|
| Registrar preventivas | Permite registrarle al interno las fechas preventivas que pueda tener asociadas. |

Conclusiones

En este capítulo se trató de forma general la arquitectura propuesta para la creación del módulo Situación Legal enfatizando en los estilos arquitectónicos propuestos Arquitectura en Capas y Arquitectura Cliente-Servidor. Se analizó el diseño realizado para la confección de ambos estilos arquitectónicos e integrando estos al frameworks de desarrollo propuesto. Se especificó a su vez toda la estructura de paquetes, clases, modelos, etc. en relación con lo antes expuesto así como con los frameworks de trabajo descritos en el capítulo 1 lo que facilita una mejor comprensión de la solución propuesta para la realización del diseño e implementación del módulo Situación Legal del Sistema Penitenciario Nacional.

CAPÍTULO 3: DISEÑO E IMPLEMENTACIÓN DE LA SOLUCIÓN

Introducción

En el presente capítulo se abordará todo lo referente al diseño e implementación del módulo Situación Legal tomando como base los estudios realizados en capítulos anteriores. Se detallarán las actividades definidas para darle cumplimiento a las funcionalidades propuestas en el capítulo anterior.

Actividades de Diseño e Implementación del módulo Situación Legal.

A continuación se detallan el conjunto de actividades definidas por el SIGI, para la construcción de la solución. Las mismas fueron desarrolladas para el diseño e implementación del Módulo Situación Legal.

Diseño del Dominio

En el modelo de dominio se captura los tipos más importantes de objetos que existen o los eventos que suceden en el entorno donde estará el sistema. El diseño del dominio constituye la entrada principal a las restantes actividades de diseño pues en él se identifican los nomencladores y las clases del dominio, las entidades que serán gestionadas por la capa de lógica de negocio, las que persistirán en la capa de acceso a datos y las que se mostrarán en la capa de presentación. La definición del dominio, en especial de las entidades persistentes sirve como una primera aproximación al diseño definitivo del modelo de datos. Grails utiliza en este sector de persistencia potentes patrones como Table Data Gateway y Row Data Gateway los cuales aseguran que existirá una tabla en la base de datos por cada entidad del dominio que se posea y que por cada atributo de la misma que se posea va a haber una fila en la tabla de la base de datos.

El diseño de la solución se enmarca específicamente sobre el CU Registrar Proceso, el cual según su complejidad se dividió en dos CU principales, Registrar documentos del proceso por el que cumple y Registrar documentos del proceso por el que no cumple.

En la Figura 8 se muestra el modelo para el caso del proceso por el que un interno ingresa al sistema. De dicho proceso, según el motivo de ingreso, se adicionan los documentos con los cuales el interno debe haber ingresado al sistema, se recogen los datos del mismo, registrando a su vez, en caso de existir las preventivas. Se recogen además en caso de existir los datos de los delitos, las sanciones accesorias y la responsabilidad civil del interno, así como los trámites legales pendientes.

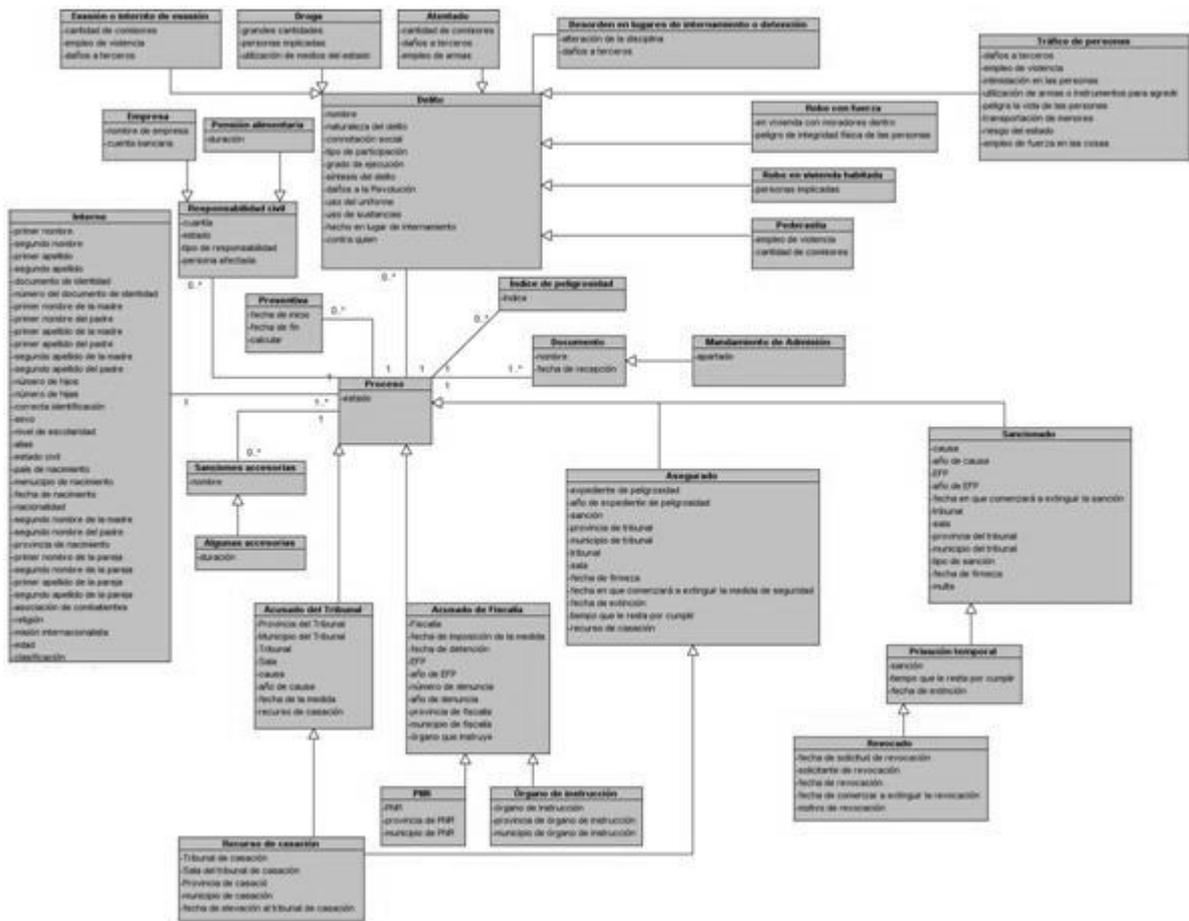


Figura: 8 Registrar Proceso por el que cumple.

Del CU global Registrar Proceso se deriva también otro CU que rige el flujo de trabajo existente en el módulo Situación Legal, el caso de uso: “Registrar documentos del proceso por el que no cumple” (Anexo 7). De este se recogen los datos del proceso por lo que un interno reingresa a prisión o por los cuales se le abre un nuevo proceso cuando ya está cumpliendo por otro, los delitos cometidos, las sanciones accesorias impuestas así como la responsabilidad civil y se recogen además los trámites legales pendientes que se pueden generar a un interno en caso de existir. A su vez el módulo se encarga de controlar además los regímenes por los cuales puede transitar el interno, los beneficios generados durante su transcurso por el sistema así como la clasificación propuesta por el sistema.

Descripción de las entidades de dominio significativas.

| Entidad | Descripción |
|----------------|------------------------------------------------------------------------------------------|
| Proceso | Almacena todos los datos referentes al proceso por el que esté cumpliendo un interno(a). |

| Atributos | Tipo | Puede ser nulo |
|--------------------|-------------|-----------------------|
| IDPROCESO | VARCHAR | NO |
| PROCESO | PROCESO | NO |
| CAUSA | INTEGER | SI |
| ANNO_CAUSA | INTEGER | SI |
| PROVINCIA_TRIBUNAL | VARCHAR | SI |
| MUNICIPIO_TRIBUNAL | VARCHAR | SI |
| TRIBUNAL | VARCHAR | SI |
| SALA | VARCHAR | SI |
| FECHA_EXTINCION | DATE | SI |
| EFP | INTEGER | SI |
| ANNO_EFP | INTEGER | SI |
| FISCALIA | VARCHAR | SI |
| PROVINCIA_FISCALIA | VARCHAR | SI |
| MUNICIPIO_FISCALIA | VARCHAR | SI |
| FECHA_MEDIDA | DATE | SI |
| SANCION | INTEGER | SI |
| MULTA | INTEGER | SI |
| TIPO_SANCION | VARCHAR | SI |

| Entidad | Descripción |
|----------------|------------------------------------------------------------------|
| DELITO | Almacena los datos de los delitos que pueda tener un interno(a). |

| Atributos | Tipo | Puede ser nulo |
|--------------------|-------------|-----------------------|
| IDDELITO | VARCHAR | NO |
| DELITO | DELITO | NO |
| NATURALEZA_DELITO | CHAR(1) | NO |
| CONNOTACION_SOCIAL | CHAR(1) | NO |
| TIPO_PARTICIPACION | CHAR(1) | NO |
| GRADO_EJECUCION | CHAR(1) | NO |
| SINTESIS_DELITO | VARCHAR | NO |
| DANNOS_REVOLUCION | CHAR(1) | NO |
| USO_UNIFORME | CHAR(1) | NO |
| USO_SUSTANCIAS | CHAR(1) | NO |
| CONTRA QUIEN | CHAR(1) | NO |
| USO_VIOLENCIA | CHAR(1) | NO |
| DANNOS_TERCEROS | CHAR(1) | NO |

| Entidad | Descripción |
|-------------------|------------------------------------------------------------------------------------------|
| SANCION_ACCESORIA | Almacena los datos de las sanciones accesorias que le pueden ser imputadas a un interno. |

| Atributos | Tipo | Puede ser nulo |
|---------------------|-------------------|-----------------------|
| IDSANCION_ACCESORIA | VARCHAR | NO |
| SANCION_ACCESORIA | SANCION_ACCESORIA | NO |
| DURACION | INTEGER | SI |

| Entidad | Descripción |
|-----------------------|----------------------------------------------------------------------------------------------------------------------|
| RESPONSABILIDAD_CIVIL | Almacena los datos de las responsabilidades civiles que puede tener un interno(a) asociadas a los delitos cometidos. |

| Atributos | Tipo | Puede ser nulo |
|-------------------------|-----------------------|-----------------------|
| IDRESPONSABILIDAD_CIVIL | VARCHAR | NO |
| RESPONSABILIDAD_CIVIL | RESPONSABILIDAD_CIVIL | NO |
| DURACION | INTEGER | NO |
| CUANTIA | INTEGER | NO |
| EMPRESA | VARCHAR | SI |
| PERSONA_AFECTADA | VARCHAR | NO |
| ESTADO | VARCHAR | NO |
| CUENTA BANCARIA | VARCHAR | SI |

| Entidad | Descripción |
|----------------|-----------------------------------------------------------------------------------|
| TRAMITE_LEGAL | Almacena los datos de los trámites legales pendientes que puede tener un interno. |

| Atributos | Tipo | Puede ser nulo |
|-------------------|---------------|----------------|
| IDTRAMITE_LEGAL | VARCHAR | NO |
| TRAMITE_LEGAL | TRAMITE_LEGAL | NO |
| ORGANO_TRAMITADOR | VARCHAR | SI |
| DOCUMENTO_ERRONEO | VARCHAR | SI |
| FECHA_GENERACION | DATE | NO |
| FECHA_RESOLUCION | DATE | SI |

Diseño de la Interfaz de Usuario

En esta actividad se definen las dos principales partes de la capa Web de Grails que van a estar presentes en el módulo: las vistas y los controladores así como el flujo de navegación de las mismas y su relación con los controladores. Las vistas son implementadas mediante GSP o JSP, y por el diseño y arquitectura de Grails estarán compuestas por los Tag Libs y la Templates que propone SiteMesh. Las mismas van a ser las encargadas de mostrar los datos obtenidos en las peticiones del usuario a los controladores así como de enviarles a estos los parámetros obtenidos. Los controladores por su parte gestionan la aplicación mediante la recepción de las acciones emitidas por el usuario a los GSP, interactuando con las mismas o simplemente con las clases de Dominio o delegando las acciones a otro controlador o capa diferente.

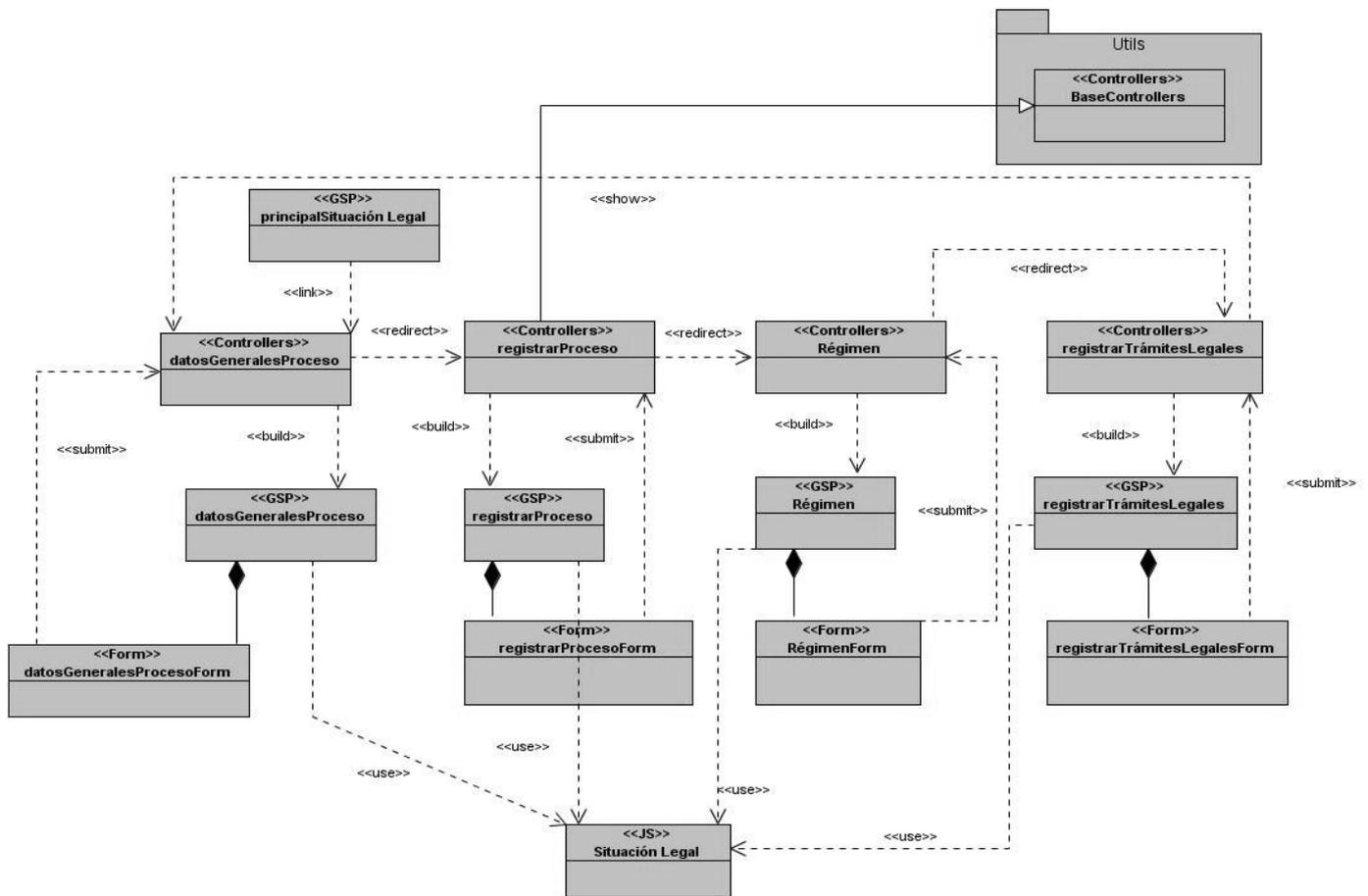


Figura: 10 Diseño de la Capa de Presentación

Implementación de las clases del dominio.

Las clases del dominio generalmente no presentan un gran comportamiento en las aplicaciones. Grails por su parte fortalece su estructura haciendo mayor la dependencia de controladores y vistas. Las clases de dominio contendrán la validación de sus atributos mediante expresiones y restricciones propias de Grails, las que facilitan en gran medida el comportamiento de dichos datos tanto en las vistas como en los métodos a los que puedan estar vinculados en los controladores. Las clases de dominio, las cuales estarán contenidas en **grails-app/domain-class**, contendrán a su vez las relaciones existentes entre ellas siguiendo la estructura del modelo de relaciones que propone Grails.

Para la relación que se establece entre las clases de dominio con las vistas y controladores Grails presenta una técnica llamada Scaffolding que una vez creada las clases de dominio le solicita a Grails que se creen las clases controladoras y las vistas necesarias para realizar las operaciones. A continuación se muestra la implementación de una clase de dominio con la validación de sus atributos así como las relaciones existentes con otras clases.

```
Class Generales {
    String edadCometerDelito
    String edadComextSancion
    String fechaAsignacion
    Boolean repatriarse
    String fechaRepatriacion

    Static hasMany = [proceso: Proceso]

    static constraints = {

        edadCometerDelito(size: 0..10)
        edadComextSancion(size: 0..10)
        fechaAsignacion(size: 0..10)
        repatriarse(size: 0..10)(nullable:true)
        fechaRepatriacion(size: 0..10)

    }
}
```

Implementación de las vistas.

Las vistas son responsables de hacer la interfaz de usuario y se implementan mediante GSP, que es la tecnología de vistas que utiliza Grails y es una extensión de JSP que puede incluir código Groovy. Las GSP tienen extensión .gsp y están dentro del directorio grails-app/views. Las GSP para introducir el código Groovy dentro de una página GSP utilizan <% %>, así como <%= %> para evaluar las expresiones y los valores de salida. No es necesario importar bibliotecas de etiquetas para construirlas pues vienen incluidas en Grails.

Las vistas van a ser las responsables de contener todas las funciones de Dojo, las cuales se crean y se definen en cada GSP, las mismas son las encargadas a su vez de invocar las llamadas a las funciones de Javascript las cuales van a estar contenidas en los archivos JS, los cuales se definieron según los métodos y funcionalidades comunes que agrupan. Su relación con los controladores se representa de forma predeterminada empleando las definiciones de Grails al cada GSP poder contener una clase controladora que la respalde.

Implementación de los controladores.

Luego de la implementación y creación de las clases de dominio se implementan los controladores mediante el uso de Scaffolding. El Scaffolding no es más que la creación de artefactos como los controladores y vistas que satisfacen una serie de requisitos, como por ejemplo la realización de operaciones CRUD a una clase de dominio, la autenticación, búsqueda y las pruebas unitarias. Los controladores son los encargados de gestionar y coordinar el flujo lógico de la aplicación mediante la recepción de las acciones del usuario. Reciben peticiones de los usuarios y actúan sobre ellas. Pueden interactuar directamente con una clase de dominio para realizar una operación de CRUD, re direccionar al usuario a un GSP diferente, delegar una acción a un actor diferente (otro controlador o una clase de servicio), y preparar y enviar la respuesta de nuevo a la vista, siempre creando un nuevo controlador para cada solicitud. Su creación va a estar definida por la sintaxis ***grails create-controller*** y su ubicación radicará en ***grails-app/Controllers*** manejando todo el flujo de trabajo de la capa Web Layers junto a las vistas.

Implementación de la capa de acceso a datos.

La implementación de la capa de acceso a datos en Grails se basa principalmente en la configuración del archivo ***DataSource.groovy***, en el cual se define el gestor de base de datos a usar así como su configuración. El uso de GORM, como una herramienta ORM evita el trabajo directo con el gestor de base de datos y sus entidades, trabajando a su vez con objetos en su lugar, lo que simplifica en gran medida el trabajo con Hibernate y elimina cualquier configuración externa. Grails es compatible con tres entornos de forma predeterminada: desarrollo, prueba y producción. Dichos entornos no solo facilitan el manejo de los datos sino que además posibilitan tratar a cada entorno de manera diferente: create-drop, create, update, blank, posibilitando hacer el trabajo de forma controlada y segura.

```

dataSource {
    pooled = true
    driverClassName = "oracle.jdbc.driver.OracleDriver"
    dialect = org.hibernate.dialect.OracleDialect
    username = "proyecto"
    password = "proyecto"
}
hibernate {

    cache.use_second_level_cache=true
    cache.use_query_cache=true
    cache.provider_class='net.sf.ehcache.hibernate.EhCacheProvider'
    //config.location = "file:/hibernate.cfg.xml"
}
// environment specific settings
environments {
    development {
        dataSource {
            dbCreate = "update" // one of 'create', 'create-drop', 'update'
            url = "jdbc:oracle:thin:@10.36.13.154:1521:SIGIDB"
            //jdbc:oracle:thin:@<host>:<port>:<sid>
        }
    }
    test {
        dataSource {
            dbCreate = "update"
            url = "jdbc:oracle:thin:@10.36.13.154:1521:SIGIDB"
        }
    }
}

```

Figura: 11 Archivo de configuración de la base de datos

GORM trae implementado a su vez métodos dinámicos como *save ()*, *delete ()*, *refresh ()* e *ident ()* así como métodos estáticos como *count ()*, *exist ()*, *find ()*, *findAll ()*, *findBy*()* y *findWhere ()*, los cuales facilitan en gran medida el trabajo con la base de datos.

Implementación de la lógica en el cliente.

Para esta actividad se implementan en los archivos JS contenidos en **Web Application/Js/SituacionLegal/SituacionLegal.js** las validaciones existentes del lado del cliente, así como el comportamiento de cada componente Dojo para los diálogos de información o error. El lanzamiento de peticiones desde el cliente al servidor también está incluido en esta actividad mediante el uso de tecnología AJAX, JSON y JSON-RPC y para el uso de componentes de bibliotecas Javascript Dojo Toolkit. Un ejemplo de validaciones hechas del lado del cliente contenidas en el fichero

js es la correspondiente a la funcionalidad Registrar Datos Legales de un proceso como se muestra a continuación:

```

save:function(){

    var antec = dijit.byId('selecantecedentes').attr("value");
    var clasif = dijit.byId('selecclasif').attr("value");
    if(dijit.byId('estatusLegald').attr("value")=="){
        document.getElementById('mensaje').innerHTML = "Debe insertar un proceso".bold();
        dijit.byId('error').show();
        return;
    }
    if(antec == ""){
        document.getElementById('mensaje').innerHTML = "Debe seleccionar un Antecedente".bold();
        dijit.byId('error').show();
    }else{
        var from = dojo.byId('formGenerales');
        var arregloBM = new Array();
        dijit.byId('tabla').store.fetch({
            query : { estado : "*" },
            onComplete : function(items, ds)
            {
                for ( var i = 0; i < items.length; i++)
                {
                    var objeto = new Object();
                    objeto.estado = items[i].estado;
                    arregloBM.push(objeto);
                }
            }
        });
        dojo.xhrPost({
            url : "/Prisiones-Cuba/generales/save",
            sync : true,
            handleAs:"json",
            form:from,
            content : {
                list:dojo.toJson(arregloBM),
                estatus:dijit.byId('estatusLegald').attr("value")
            },
            load : function(data) };
        pruebajs.a.myjs.mensaje(); ]}]
    
```

Diseño de la funcionalidad Registrar Proceso

La funcionalidad Registrar Proceso va a ser la encargada de registrar todo el flujo de trabajo del Módulo Situación Legal. La solución permitirá inicialmente registrar los principales datos legales del interno, entre los que se encuentran, la clasificación, la categoría, fechas delictivas y el estatus legal, el cual se registra una vez terminado de registrarse el proceso. Posteriormente se registran los documentos asociados a cada proceso como se muestra en la Figura 13 y los datos del proceso Figura 14 registrando a su vez las fechas preventivas en caso de existir.

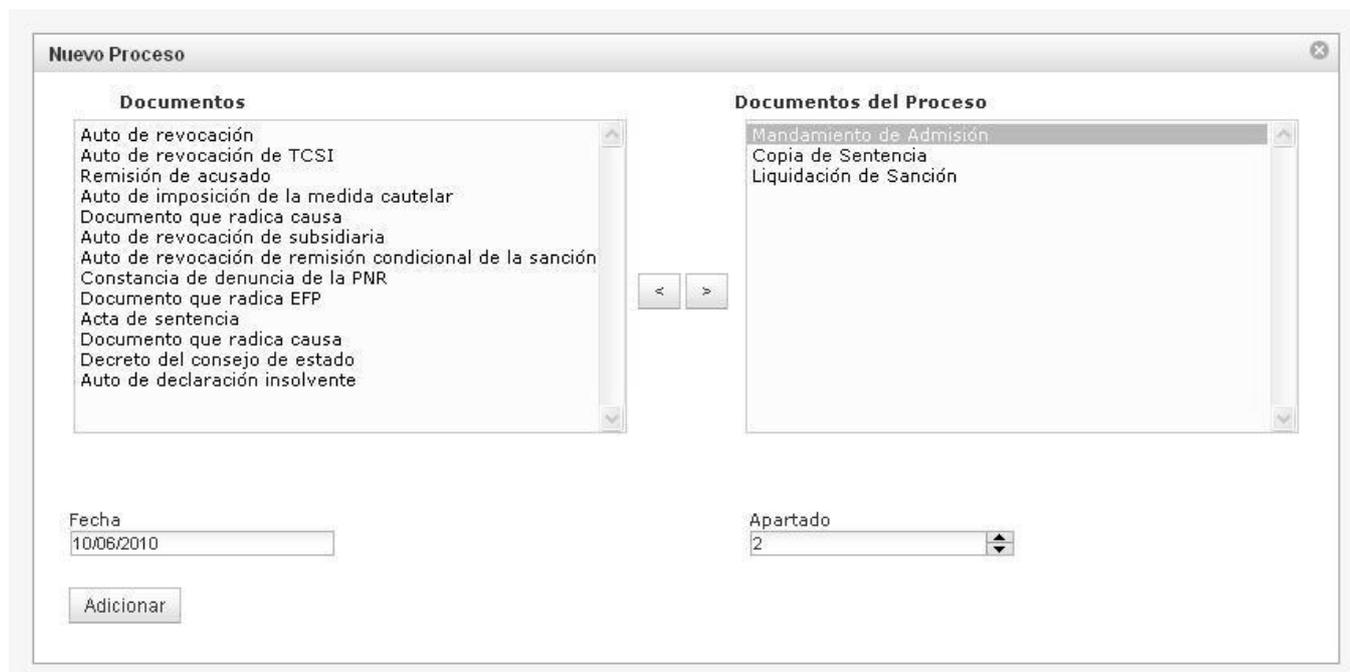


Figura 13: Interfaz Registrar Documentos del Proceso.

Dichos procesos pueden ser de diferentes tipos recogiendo a su vez diferentes datos según el motivo de ingreso que lo arraigue. Los tipos de procesos por los que puede cumplir un interno no deben variar según concesiones que se han hecho con el cliente, y aunque las mismas pueden estar sujetas a cambios en un futuro tanto la flexibilidad del sistema como la arquitectura de Grails aseguran que dichos cambios no afecten de forma directa la ejecución de la solución. La existencia de algún nuevo proceso en la capa de acceso a datos solo conllevará la inserción de la misma en el gestor así como leves cambios en las clases de dominio al Grails tenerlas estrechamente relacionadas en la capa Domain Layers. De los procesos se pueden registrar a su vez en caso de permitirlo el proceso y según los documentos que el mismo contenga tanto los Delitos, como las Sanciones Accesorias, la Responsabilidad Civil y los Trámites Legales Pendientes. Para cada uno de estos datos antes mencionados se gestionará toda la lógica de negocio así como el flujo de navegación y las

operaciones de acceso a datos en sus respectivas clases encargadas de controlar la correcta operatividad de dichas funcionalidades.

Los cambios en la capa Web Layers serían a su vez mínimos, pues el uso de **Scaffolding** manejaría todos los cambios propuestos por la capa Domain Layers.

Figura 14: Interfaz Registrar Datos Proceso

Las vistas encargadas de todo el flujo de navegación al registrar un proceso estarán contenidas en los **Tag Container** y **Content Page** y las mismas contarán de cinco páginas las cuales mostrarán en el siguiente orden, los documentos a registrar de un proceso, los datos de proceso, los delitos asociados al mismo, así como las sanciones accesorias y la responsabilidad civil que le pudo ser imputada a un interno(a). Cada GSP tendrá asociada un controlador el cual será el encargado de controlar todo el flujo de la funcionalidad, creando las páginas, validando los datos recogidos en las mismas y redireccionando a otros controladores en caso necesario.

En el controlador de proceso, el método **buscarTipoProceso** es el encargado de gestionar en el módulo Ingreso correspondiente al Subsistema Área de ingreso, observación, evaluación y diagnóstico el motivo de ingreso por el cual será registrado el proceso al interno(a), y mostrar las vistas

correspondientes a dicho proceso. El controlador **BaseControllers** a su vez gestionará el flujo de navegación existente entre los controladores y sus vistas.

Pruebas

Luego de diseñar e implementar las funcionalidades correspondientes al Módulo Situación Legal es necesario contemplar como parte de los mismos la preparación y ejecución de pruebas, así como revisiones del código e integración de las partes que componen el software. Para darle cumplimiento a lo antes expuesto se trazaron un conjunto de pruebas las cuales se detallan más adelante en el Anexo 12, donde se muestra un ejemplo para el caso de prueba Registrar Proceso. Al ejecutar dicho caso de prueba se obtuvo como resultado un conjunto de no conformidades las cuales fueron corregidas posteriormente.

No conformidades detectadas al caso de prueba Registrar Proceso a un Asegurado:

- En la interfaz Registrar documentos del proceso en ocasiones al adicionar el Mandamiento de Admisión y luego quitarlo el campo Apartado queda activo.
- El apartado recogido para el caso del Registrar Proceso a un asegurado debe de ser solo de 1, 2 o 4.
- En caso de llegarle al interno los documentos Mandamiento de Admisión y Liquidación de la Sanción el sistema debe brindar la posibilidad de registrarle las fechas preventivas al interno.
- El sistema debe brindar la posibilidad de recoger la Responsabilidad Civil y las Sanciones Accesorias a los internos que como Asegurado le llegue el documento de Copia de Sentencia.

Conclusiones

Al concluir el presente capítulo se detallaron de forma más específica todas las actividades definidas por el SIGI para el diseño e implementación del Módulo Situación Legal del Sistema Penitenciario Nacional, mediante el uso de diagramas y fragmentos de códigos. Se definieron a su vez los posibles casos de prueba a los que estará sometido el módulo, dando como resultado un conjunto de no conformidades a las cuales se le dio cumplimiento durante el transcurso del desarrollo.

Conclusiones

Como resultado del trabajo realizado se diseñó e implementó el módulo Situación Legal del Sistema Penitenciario Nacional a partir de los requerimientos de software establecidos con el cliente y haciendo uso de la arquitectura y las tecnologías definidas por el proyecto.

Se realizó un estudio de las tecnologías y herramientas a utilizar en el diseño e implementación del módulo así como del análisis de los requisitos de software y del modelo de negocio correspondientes.

Las actividades de diseño e implementación fueron ejemplificadas a través de diagramas de clase y fragmentos de código fuente cumpliendo así con los objetivos propuestos para este trabajo.

Se obtuvo un conjunto de herramientas capaces de realizar una gestión ágil de la información sobre los procesos legales del individuo(a), los trámites legales pendientes a los que puede estar sujeto así como el régimen por el cual transita.

Se logró erradicar en gran medida las insuficiencias que presentaban las herramientas predecesoras en este sentido.

RECOMENDACIONES

Comprobar el rendimiento del módulo frente a bases de datos de grandes volúmenes de información para validar que el diseño propuesto satisface tales condiciones, puesto que actualmente el sistema está probado para una cantidad limitada de datos que no ponen al límite la solución propuesta

Revisar las funcionalidades del módulo Situación Legal con el objetivo de agregar nuevas funcionalidades.

Realizar iteraciones posteriores en aras de satisfacer en mayor grado los requisitos funcionales propuestos por el cliente con el objetivo de contar con un diseño de mayor robustez y flexibilidad.

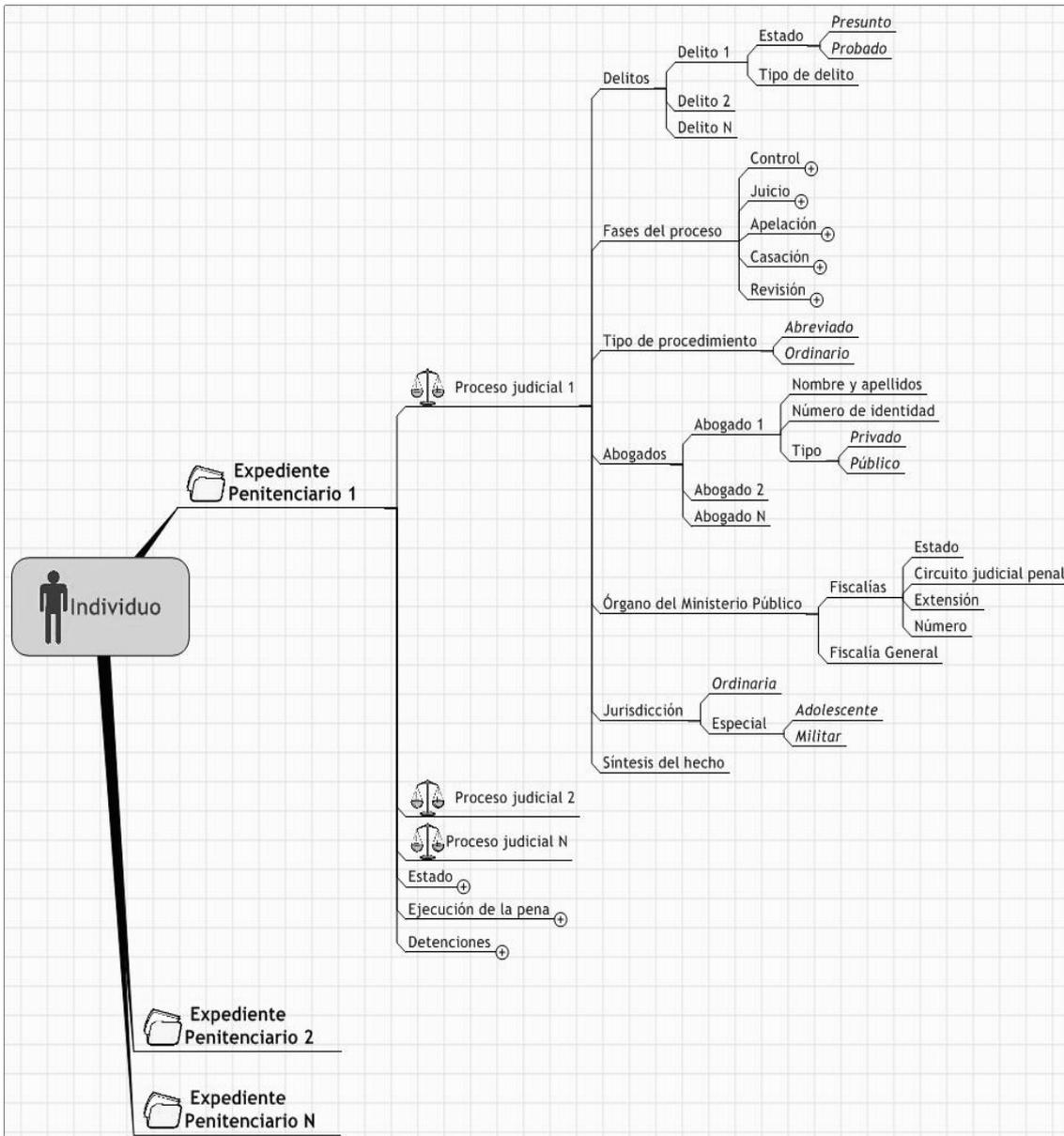
BIBLIOGRAFÍA

1. **Embarcadero Technologies Inc.** Embarcadero. [En línea] <http://www.embarcadero.com/products/er-studio>.
2. **Sun Microsystems Inc.** OpenSymphony. [En línea] <http://www.opensymphony.com/sitemesh/>.
3. **Apache Software Foundation.** Apache. [En línea] <http://tomcat.apache.org/>.
4. **Json.org.** JSON. *Java Script Object Notation*. [En línea] <http://www.json.org/json-es.html>.
5. **Open Source Software Engineering Tools.** Subversion. [En línea] <http://subversion.tigris.org/>.
6. **Oracle Corporation and/or its affiliates.** NetBeans. [En línea] <http://netbeans.org/index.html>.
7. **Sun Microsystems Inc.** Javascript. [En línea] <http://www.javascript.com/>.
8. **The Dojo Foundation.** DojoToolkit. [En línea] <http://www.dojotoolkit.org/>.
9. **Código Penal Cubano.** *Ley nro. 62*. República de Cuba : s.n.
10. **Juventud Rebelde.** *Informe Nacional de la República de Cuba al Examen Periódico Universal del Consejo de Derechos Humanos*. Ciudad de la Habana : s.n., 2009.
11. **Abdul-Jawad, Bashar.** *Groovy and Grails Recipes*. United States of America : Sun Microsystems Inc, 2009. ISBN-13 (pbk): 978-1-4302-1600-1.
12. **Ministerio de Justicia.** Gaceta Oficial de la República de Cuba. [En línea] <http://www.gacetaoficial.cu..>
13. **Ministerio del Interior.** *Informe de Factibilidad del proyectode desarrollo estratégico del Sistema Penitenciario Nacional*. Ciudad de la Habana : s.n., 2008-2009.
14. **Ministerios de Relaciones Exteriores.** CubaMinRex. [En línea] <http://www.cubaminrex.cu..>
15. **Naciones Unidas.** *Reglas Minimas para el Tratamiento de los Reclusos*. 1955.
16. **Rational Software Corporation.** *RUP. "Rational Unified Process"*. 2003.
17. **República Federativa de Brasil.** Brasil. [En línea] <http://www.brasil.gov.br..>
18. **SpringSource.** Grails. *Grails*. [En línea] SpringSource, 2009. <http://www.grails.org>.
19. **Visual Paradigm International.** Visual Paradigm. [En línea] [Citado el: 10 de febrero de 2010.] <http://www.visual-paradigm.com..>
20. **Albiol, Francesc Rosés.** *Introducción a Hibernate*. 2003.

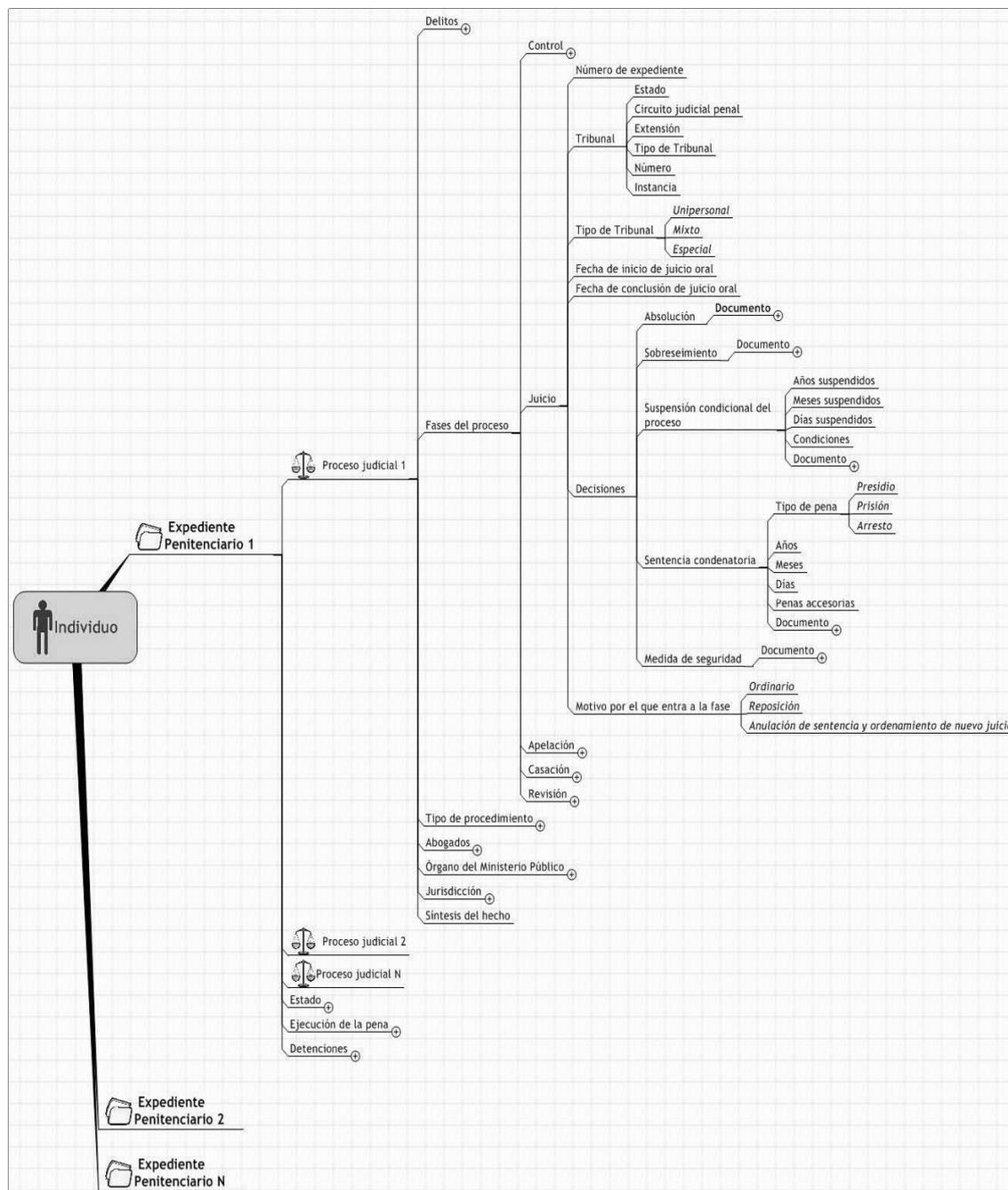
21. **Benavides Zaila, Yadira y Gómez Correa, Juan Carlos.** *Diseño e implementación de los módulos Decisiones y Egresos del Sistema de Gestión Penitenciaria de la República Bolivariana de Venezuela.* Ciudad de la Habana : s.n., 2008.
22. **Bodoff, Stephanie.** *The J2EE Tutorial.* s.l. : Addison-Wesley., 2002.
23. **Gaceta Oficial de la República de Cuba.** *Resolución No-76/2000.* Ciudad de La Habana : s.n., 2000.
24. **Glen Smith, Peter Ledbrook.** *Grails in Action.* United States of America : Manning Publications Co., 2009. ISBN 978-1-933988-93-1.
25. **Gómez., Joaquina Naranjo.** *El Sistema Penitenciario Cubano.* . 2006.
26. **Gongora Rodríguez, Alien y Pupo Diéguez, Yordanys.** *Diseño e implementación de la Capa de Presentación de las funcionalidades relacionadas con el registro de decisiones y la ejecución de la pena del módulo Situación Jurídica.* Ciudad de la Habana : s.n., 2008.
27. **Jacobson, Ivar, Booch, Grady y Rumbauch, James. 2000.** *El Proceso Unificado de Desarrollo del Software.* . Madrid : s.n., 2000.
28. **King, Gavin y Bauer, Christian.** *Hibernate in Action.* 2004.
29. **LADD., SETH.** *Expert Spring MVC and Web Flow.* . 2006. .
30. **Larman., Craig.** *UML Y Patrones. Introducción al análisis y diseño orientado a objeto.* . Mexico : s.n.
31. **ORACLE, CORPORATION.** *Oracle Database Documentation Library.* 2005.
32. **Peiso Hernández, TC. Filiberto, Curbelo Mederos, Tte. Coronel Raúl y Fernández Borrego, 1-Tte. Jaen.** *INFORME DE FACTIBILIDAD E IDEAS PRELIMINARES SOBRE LA INFORMAIZACION DEL SISTEMA PENITENCIARIO CUBANO.* Ciudad de la Habana : s.n., 2009.
33. **Pressman., Roger.** *Ingeniería de Software. Un enfoque práctico.* . 2002.
34. **Santillán., Lesbia Pérez.** *Sistema Bancario en México.* 2006. 2006.
35. **Tormo, Carmen Cantero.** *Software de reconocimiento y verificación de firmas manuscritas digitalizadas.* Barcelona : Universidad Ramon Llull.
36. **Walls, Craig y Breidenbach, Ryan.** *Spring in Action.* Estados Unidos de América : Manning Publications Co., 2005.
37. **Zequeira Peña, Dr. Tte. Cnel. Alfonso y Céspedes Quesada, Lic.1er Tte. Aimeé.** *Vocabulario Jurídico Penitenciario.* Ciudad de la Habana : MININT, 2005.

ANEXOS

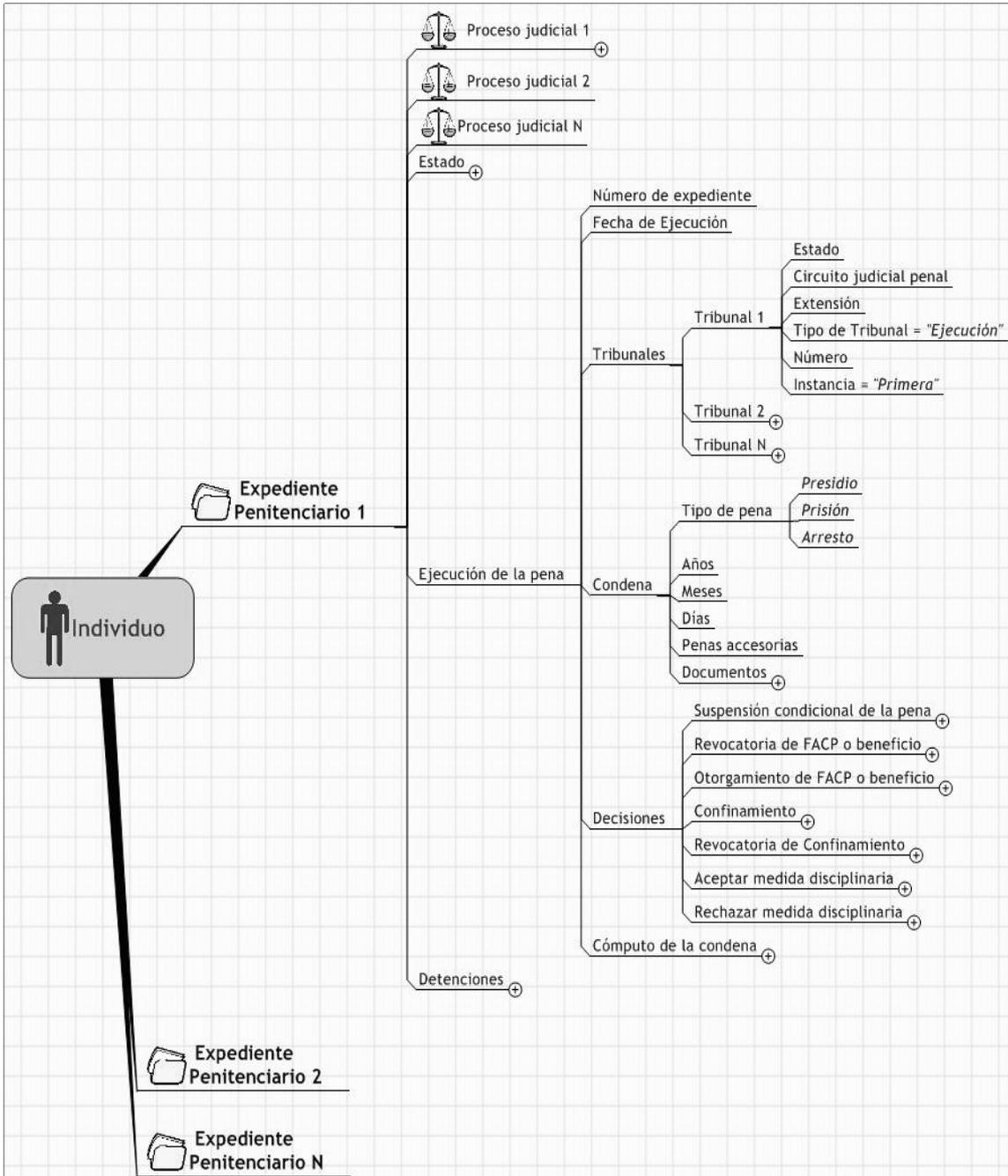
Anexo 1: Proceso Judicial (SIGEP)



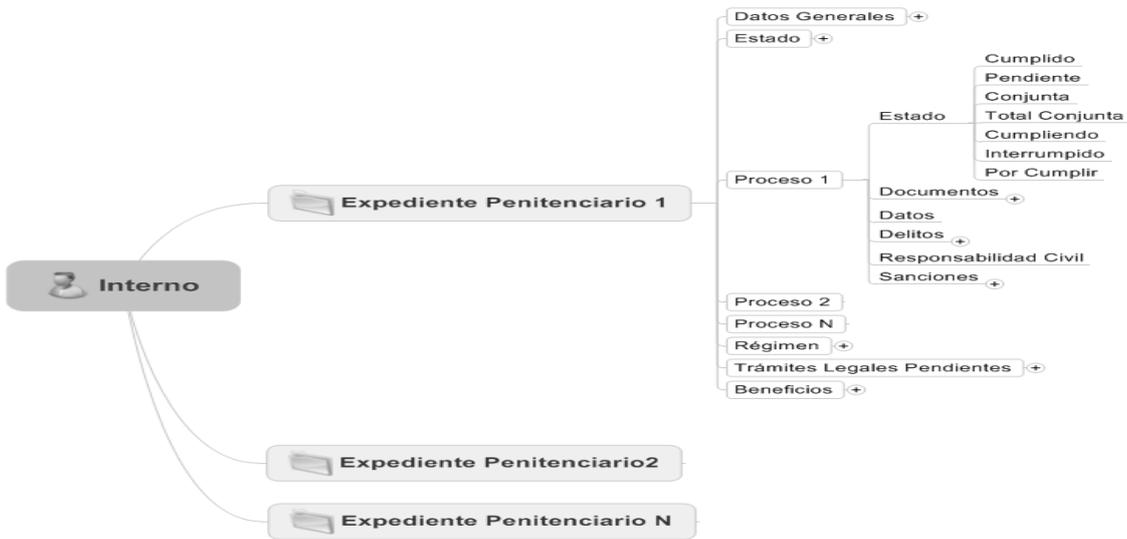
Anexo 2: Fases y Decisiones Judiciales (SIGEP)



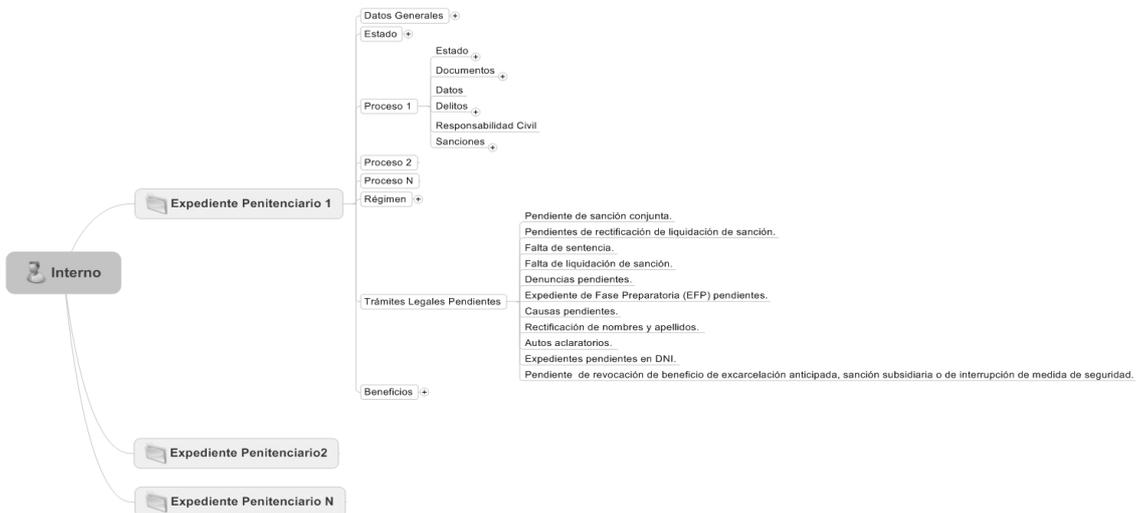
Anexo 3: Ejecución de la Pena (SIGEP)



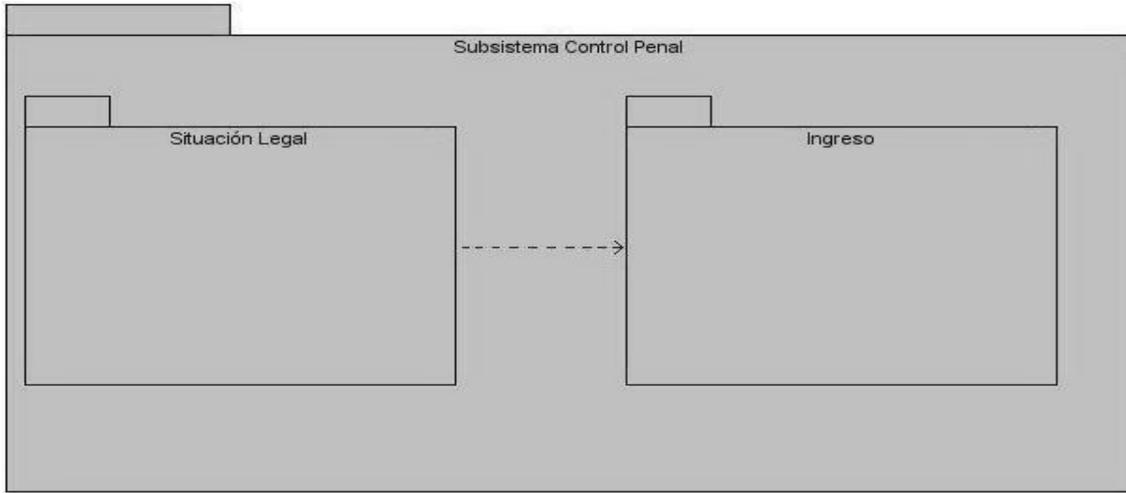
Anexo 4: Procesos Legales (SIGI)



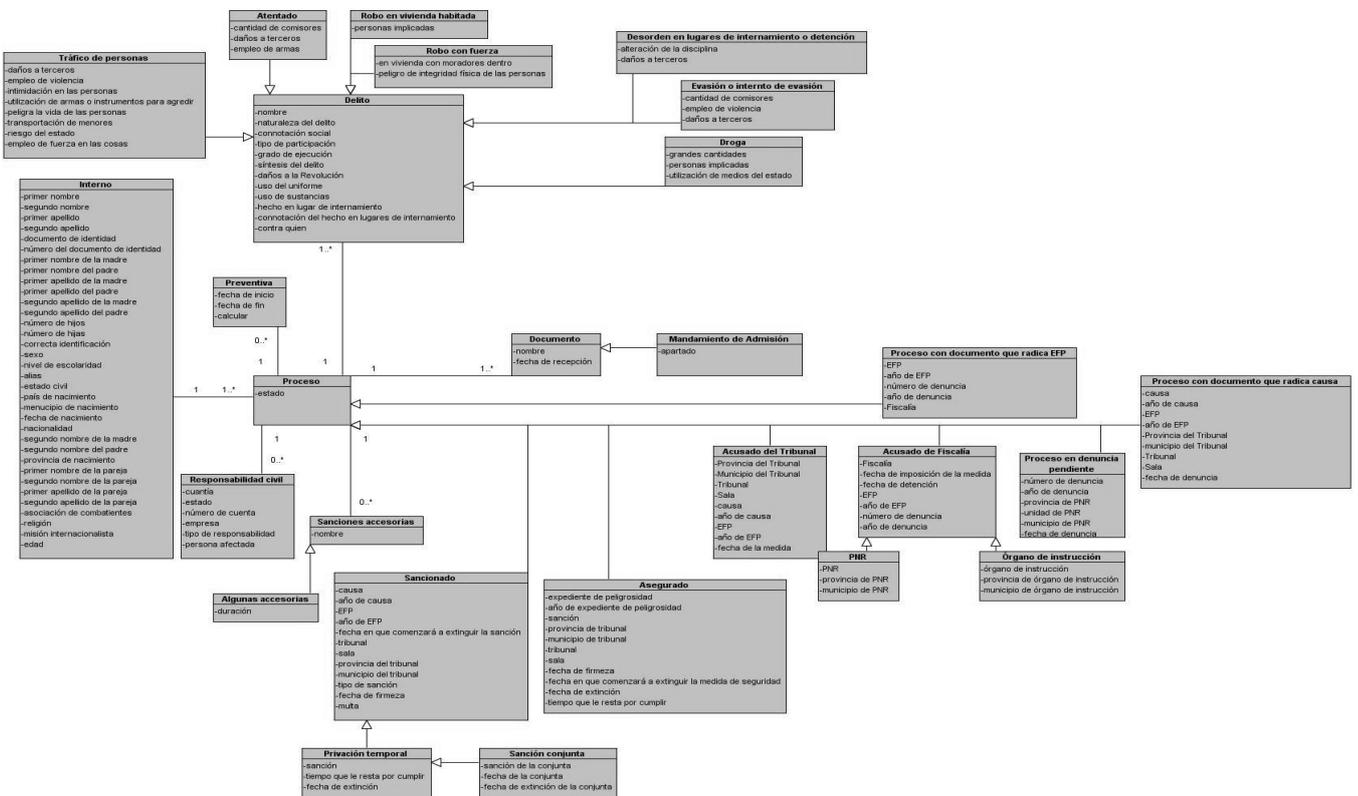
Anexo 5: Trámites Legales Pendientes (SIGI)



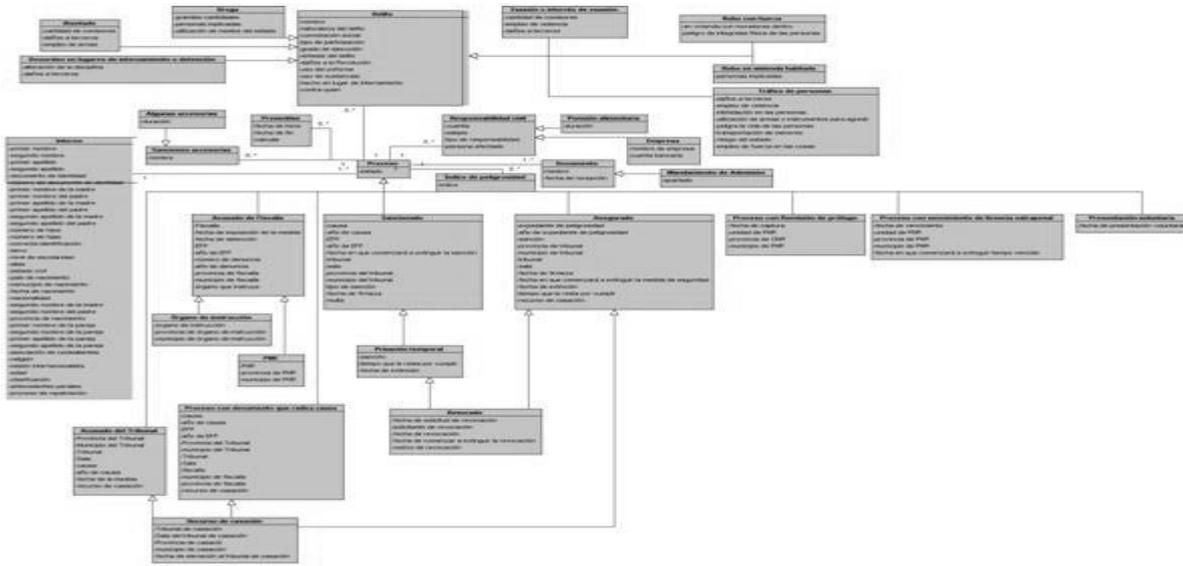
Anexo 6: Diseño de paquetes



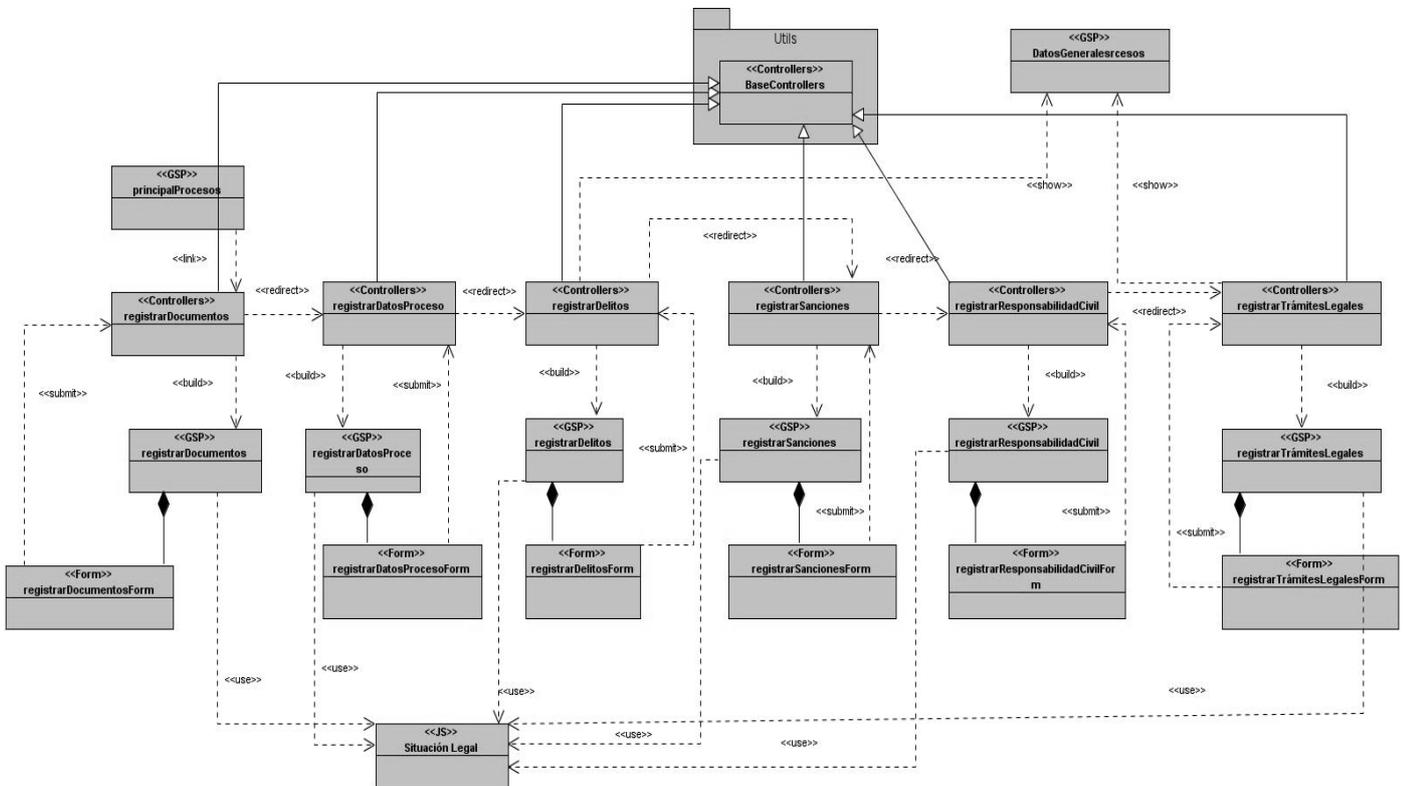
Anexo 7: Registrar Proceso por el que no cumple



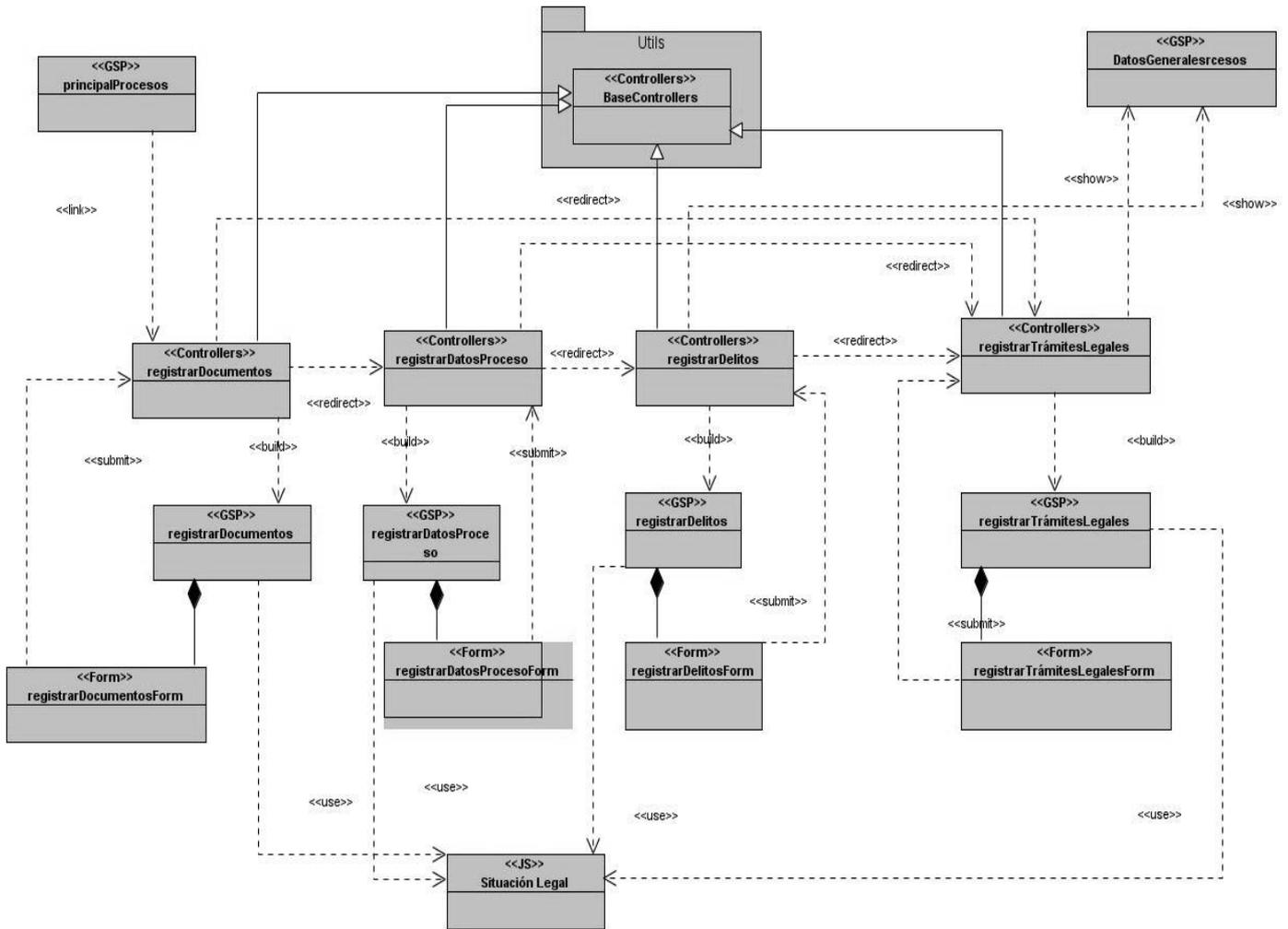
Anexo 8: Actualizar Proceso



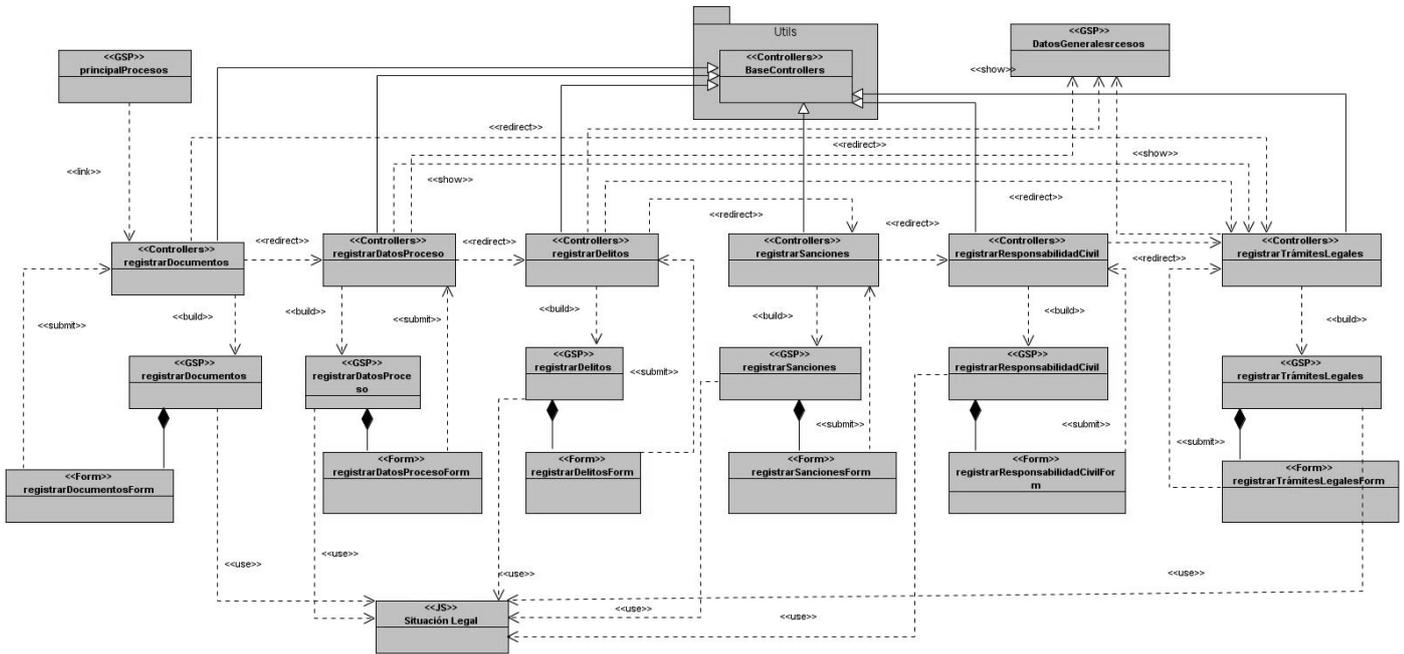
Anexo 9: Diseño CU-Registrar Proceso por el que Cumple.



Anexo 10: Diseño CU-Registrar Proceso por el que no Cumple.



Anexo 11: Diseño CU-Actualizar Proceso.



Anexo 12: Caso de prueba Registrar Proceso

Esta funcionalidad permite registrar el proceso por el que un interno ingresa a prisión. Recogiendo del mismo los datos generales del proceso, los documentos, los delitos, las sanciones accesorias, la responsabilidad civil así como los trámites legales pendientes que se le pudieron haber generado. Por la complejidad de la funcionalidad y por el mismo contener diversos motivos de ingreso se especificó el siguiente caso de prueba para el caso del Registrar Proceso a un Asegurado.

Clases Válidas

| No. | Nombre del Campo | Clasificación | Puede ser Nulo | Descripción |
|-----|-----------------------------------------|-------------------|----------------|--------------------------------------------------------------------------------------------------------------------------|
| 1 | Antecedentes Penales | Lista Desplegable | No | Todas las letras del abecedario español, vocales acentuadas y el espacio (Primario, Reincidente, Multireincidente, etc.) |
| 2 | Estatus Legal | Lista Desplegable | No | Todas las letras del abecedario español, vocales acentuadas y el espacio (Acusado, Asegurado, Sancionado, etc.) |
| 3 | Clasificación | Lista Desplegable | No | Todas las letras del abecedario español, vocales acentuadas y el espacio (Común, Militar Común, Tránsito, CSE, etc.) |
| 4 | Edad al cometer el delito | Cuadro de Texto | No | Fechas menores o iguales 15 años y que cumplan con el formato dd/mm/aaaa. |
| 5 | Edad al comenzar a extinguir la sanción | Cuadro de Texto | No | Fechas menores o iguales que la actual que cumplan con el formato dd/mm/aaaa. |
| 6 | Categoría | Lista Desplegable | No | Todas las letras del abecedario español, vocales acentuadas y el espacio (Apelación, Ofensivas, CSE, etc.) |
| 7 | Fecha Asignación | Cuadro de Texto | No | Fechas menores o iguales que la actual que cumplan con el formato dd/mm/aaaa. |
| 8 | Repatriarse | Radio Button | Sí | |
| 9 | Fecha de Repatriación | Cuadro de Texto | No | Fechas menores o iguales que la actual que cumplan con el formato dd/mm/aaaa. |
| 10 | Apartado | Cuadro de Texto | No | Todas las letras del abecedario español, vocales acentuadas y el espacio (1, 2, 3, 4, 5, etc.) |

| | | | | |
|----|------------------------------------|-------------------|----|--------------------------------------------------------------------------------|
| 11 | Expediente de Peligrosidad | Cuadro de Texto | No | Todas las letras del abecedario español, vocales acentuadas y el espacio, etc. |
| 12 | Año del Expediente de Peligrosidad | Cuadro de Texto | No | Todas las letras del abecedario español, vocales acentuadas y el espacio, etc. |
| 13 | Provincia del Tribunal | Lista Desplegable | No | Todas las letras del abecedario español, vocales acentuadas y el espacio, etc. |
| 14 | Municipio del Tribunal | Lista Desplegable | No | Todas las letras del abecedario español, vocales acentuadas y el espacio, etc. |
| 15 | Tribunal | Lista Desplegable | No | Todas las letras del abecedario español, vocales acentuadas y el espacio, etc. |
| 16 | Sala | Lista Desplegable | No | Todas las letras del abecedario español, vocales acentuadas y el espacio, etc. |
| 17 | Fecha de la Medida | Cuadro de Texto | No | Fechas menores o iguales que la actual que cumplan con el formato dd/mm/aaaa. |
| 18 | Sanción | Cuadro de Texto | No | Todas las letras del abecedario español, vocales acentuadas y el espacio, etc. |
| 19 | Fecha de Generación | Cuadro de Texto | No | Fechas menores o iguales que la actual que cumplan con el formato dd/mm/aaaa. |

Clases Inválidas

Todas las posibles combinaciones de datos insertados por el probador

Escenarios de Uso

- **Escenario 1:** Registrar proceso correctamente
 1. Seleccionar pestaña Generales
 2. Insertar los Datos Generales del Proceso
 3. Seleccionar Adicionar
 4. Insertar Datos del Proceso
 5. Seleccionar Aceptar
 6. Insertar Datos de los Trámites Legales
 7. Seleccionar Adicionar

- **Escenario 2:** Registrar proceso insertando datos erróneos
 - 1 Seleccionar pestaña Generales
 - 2 Insertar los Datos Generales del Proceso incorrectamente
 - 3 Seleccionar Adicionar
 - 4 Insertar Datos del Proceso incorrectamente
 - 5 Seleccionar Aceptar
 - 6 Insertar Datos de los Trámites Legales incorrectamente
 - 7 Seleccionar Adicionar

- **Escenario 3:** Registrar proceso insertando datos en blanco
 - 1 Seleccionar pestaña Generales
 - 2 Insertar los Datos Generales del Proceso dejando al menos un campo en blanco
 - 3 Seleccionar Adicionar
 - 4 Insertar Datos del Proceso dejando al menos un campo en blanco
 - 5 Seleccionar Aceptar
 - 6 Insertar Datos de los Trámites Legales dejando al menos un campo en blanco
 - 7 Seleccionar Adicionar

Juegos de Datos

| ID | Escenarios | Clases Válidas | Clases Inválidas | Resultado esperado | Resultado obtenido |
|----|-----------------------------------------------------------------|----------------------------------------|----------------------------------|---------------------------------------------------------------------------------------------------------------------------------|--------------------|
| 1 | Registrar Estatus Legal correctamente | Estatus Legal: "Asegurado" | | El sistema asigna el Estatus Legal al interno. | |
| 2 | Adicionar Categoría correctamente. | Categoría: "CSE" | | El sistema adiciona una categoría correctamente. | |
| 3 | Registrar los documentos de un proceso. | Documento: "Mandamiento de admisión" | | El sistema adiciona el documento seleccionado al proceso. | |
| 4 | Insertar los Datos Generales del Proceso dejando campos vacíos. | | Antecedentes Penales: "Primario" | El sistema muestra un mensaje de error indicando que el campo vacío debe ser llenado. | |
| 5 | Insertar datos erróneos al proceso. | | Causa: "[1, 2, 3...]" | El sistema muestra un error indicando que los datos entrados son incorrectos. | |
| 6 | Filtrar información | Provincia de Tribunal: "Ciudad Habana" | | El sistema realiza una búsqueda de filtrado permitiendo seleccionar en el campo Municipio solo los municipios de Ciudad Habana. | |

| | | | | | |
|---|--------------------------------------------------------------------|--|--------------------------------|-----------------------------------------------------------------------------------------|--|
| 7 | Registrar fecha de generación de un Trámite Legal incorrectamente. | | Fecha Generación: "dd/mm/aaaa" | El sistema muestra un mensaje de error indicando que el formato de fecha es incorrecto. | |
|---|--------------------------------------------------------------------|--|--------------------------------|-----------------------------------------------------------------------------------------|--|

Anexo 13: Escenario de prueba 1

Buscar

Ingreso Datos Generales Decomisos y Ocupaciones Situación Legal Entrevistas Desig. y Ubicación

Generales Procesos Régimen Trámites

Acusado Tribunal

Causa: Año de Causa:

Provincia Tribuna: Municipio Tribunal: Tribunal: Sala:

Fecha Medida:

Repatriarse: Si No Fecha de Elevación:

Provincia Tribunal: Municipio Tribunal: Tribunal: Sala:

Datos insertados correctamente

Anexo 14: Escenario de prueba 3.

The screenshot displays a web application interface with a search bar at the top right labeled 'Buscar'. Below it are several tabs: 'Ingreso', 'Datos Generales', 'Decomisos y Ocupaciones', 'Situación Legal', 'Entrevistas', and 'Desig. y Ubicación'. The 'Datos Generales' tab is selected, showing sub-tabs for 'Generales', 'Procesos', 'Régimen', and 'Trámites'. The main form area contains the following fields and controls:

- Antecedentes Penales:** A dropdown menu.
- Estatus Legal:** A text field containing 'Acusado Pendiente a Juicio'.
- Clasificación:** A dropdown menu.
- Edad al cometer el delito:** A text field.
- Edad al comenzar a extinguir la sancion:** A text field.
- Repatriarse:** Radio buttons for 'Si' and 'No' (selected).
- Categoria:** A dropdown menu.
- Fecha de Asignacion:** A text field.
- Buttons:** 'Adicionar', 'Actualizar', 'Eliminar', and 'Adicionar Generales'.

An error dialog box is overlaid on the form, titled 'ERROR'. It contains a warning icon and the text: 'Ha ocurrido un error. Debe seleccionar un Antecedente'.

Anexo 15: Registrar Responsabilidad Civil.

Buscar

Ingreso Datos Generales Decomisos y Ocupaciones Situación Legal Entrevistas Desig. y Ubicación

Generales Procesos Régimen Trámites

Responsabilidad Civil

| Tipo Responsabilidad | Persona Afectada | Cuantía | Estado |
|--------------------------------|------------------|----------|--------------|
| Pensión Alimenticia | Persona Jurídica | 12323.50 | Liquidándola |
| Indemnización de perjuicios | Persona Jurídica | 47823.90 | Liquidándola |
| Reparación de daños Materiales | Persona Natural | 452.10 | En espera |
| Indemnización de perjuicios | Persona Natural | 897.45 | Liquidada |

Adicionar Eliminar

Tipo Responsabilidad
Indemnización de perjuicio

Cuantía
897.45

Persona Afectada
Persona Natural

Estado
Liquidada

Siguiete

Anexo 16: Registrar Delitos.

Buscar

Ingreso Datos Generales Decomisos y Ocupaciones Situación Legal Entrevistas Desig. y Ubicación

Generales Procesos Régimen Trámites

Delitos

| Delito | Fecha |
|---------------------------|------------|
| Robo en vivienda habitada | 02/06/2010 |
| Pederastía | 18/06/2010 |
| Droga | 12/06/2010 |

Tipo Delito
Droga

Fecha Delito
12/06/2010

Adicionar Actualizar Eliminar

Siguiete

Anexo 17: Registrar Sanciones Accesorias.

Buscar

◀ Ingreso Datos Generales Decomisos y Ocupaciones Situación Legal Entrevistas Desig. y Ubicación ▶▶

Generales Procesos Régimen Trámites

Sanciones Accesorias

| Sanción | Duración |
|-----------------------|----------|
| Destierro | 1254 |
| Comiso | 254 |
| Privación de derechos | 10245 |

Tipo Sanción Duración

GLOSARIO DE TÉRMINOS

SIGI: Sistema de Gestión del Interno(a).

SACORE: Sistema Automatizado para el Control del Recluso.

Interno: En Derecho Penitenciario y con las diferencias de cada país, se utiliza para referirse a las personas que cumplen sanción y medida de seguridad reeducativa, en un centro penitenciario o especializado de trabajo.

Acusado: Persona a quien se le haya decretado la medida cautelar de prisión provisional.

Sancionado: Persona ejecutoriamente sancionada a privación de libertad o a trabajo correccional con internamiento.

Asegurado: Persona a quien se le hubiere impuesto una medida de seguridad reeducativa de internamiento.

Clasificación: Es la división y agrupación de la población carcelaria en grupos de más o menos homogéneos bajo diferentes criterios jurídicos-criminológicos.

Expediente de Fase Preparatoria: Sumario en que obran todas las acciones policiales, tanto favorables como adversas al acusado.

Generales: En Derecho Penitenciario, son los datos que identifican al recluso y su Situación Legal (nombres y apellidos, padres, fecha y lugar de nacimiento, dirección del domicilio, número del carné de identidad, ocupación, oficio, causa, tribunal, delito, sanción, fecha de ingreso y de extinción).

Delito: Acción u omisión socialmente peligrosa prohibida por la Ley bajo conminación de una sanción.

Responsabilidad civil: Es la obligación del sancionado de resarcir y reparar el daño civil proveniente de la comisión del delito.

Sanción accesoria: Es la que se impone conjuntamente con una principal. Ellas son, según el Código Penal: privación de derechos; pérdida o suspensión de derechos paternos, filiales y de tutela; prohibición de ejercer una profesión, etc.

Régimen Penitenciario: Conjunto de normas que establecen el Reglamento y la Ley que deben observar los reclusos durante el cumplimiento de la sanción o medida.

Framework: Estructura de soporte definida en la cual otro proyecto de software puede ser organizado y desarrollado.

API: Interfaz de Programación de Aplicaciones.

Plugins: Un plugin o plug-in, es una aplicación que interactúa con otra para agregarle una funcionalidad específica y es ejecutada por la aplicación principal. En el caso particular de Eclipse no son más que un conjunto de clases que permiten hacerlo más extensible.

IDE: Integrated Development Environment (Ambiente Integrado de Desarrollo).

JVM: Java Virtual Machine (Maquina Virtual de Java).

CU: Caso de Uso.

Módulo: Encapsula un conjunto de funciones que debe realizar el sistema, las cuales son agrupadas por tener características muy similares y se definen en la etapa de diseño.

Subsistema: Un subsistema se refiere a un conjunto de módulos que por razones de similitud o de perseguir objetivos comunes, son agrupados.

Código o código fuente: Conjunto de instrucciones que componen un programa, escrito en cualquier lenguaje.