

Universidad de las Ciencias Informáticas

Facultad 6



Título: "SACCEM: Módulo de gestión de la información del Departamento de Supervisión del CCEEM versión 1.0."

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

Autores:

Maidelys Machado Díaz

Yader Luis Coca Ribas

Tutores:

Ing. Denny Hernández Peña

Ing. Raniel Rivero Sevilla

Ing. Yanet Parra Infante

Ing. Juan Manuel Ruiz Godoy

La Habana, 17 de junio de 2009.

“Lo importante es no dejar de hacerse preguntas.”

Albert Einstein

DECLARACIÓN DE AUTORÍA

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

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

Maidelys Machado Díaz

Firma del autor

Yader Luis Coca Ribas

Firma del autor

Ing. Dennys Hernández Peña

Firma del tutor

Ing. Ranniel Rivero Sevilla

Firma del tutor

Ing. Yanet Parra Infante

Firma del Autor

Ing. Juan Manuel Ruiz Godoy

Firma del Autor

DATOS DE CONTACTO

Tutores:

Ing. Dennys Hernández Peña

Universidad de las Ciencias Informáticas,
Ciudad de La Habana, Cuba.

dhernandezp@uci.cu

Ing. Ranniel Rivero Sevilla

Universidad de las Ciencias Informáticas,
Ciudad de La Habana, Cuba.

rrivero@uci.cu

Ing. Yanet Parra Infante

Universidad de las Ciencias Informáticas,
Ciudad de La Habana, Cuba.

yinfante@uci.cu

Ing. Juan Manuel Ruiz Godoy

Universidad de las Ciencias Informáticas,
Ciudad de La Habana, Cuba.

jmruiz@uci.cu

Autores:

Maidelys Machado Díaz

Universidad de las Ciencias Informáticas,
Ciudad de La Habana, Cuba.

mdiaz@estudiantes.uci.cu

Yader Luis Coca Ribas

Universidad de las Ciencias Informáticas,
Ciudad de La Habana, Cuba.

ylcoca@estudiantes.uci.cu

AGRADECIMIENTOS

De Maidelys:

A Victoria, Mercedes y Marbin, porque en mí hay pedacitos de ellos.

A mis amigos, a los que no les hace falta que mencione sus nombres. A las Súper Nenas.

A los profes sabios y nobles, entre ellos mis tutores.

Al que me hizo la presentación, porque sin saberlo ha hecho más que eso.

A los que me han cuidado, aguantado y entendido.

De Yader:

Primero que todo quiero agradecerles a mis padres por haberme guiado siempre, por tanto sacrificio, por haber sabido siempre el camino correcto. A mi novia. A mis amigos y amigas que lo han sido sin condiciones, que han sido tan importantes en mi vida y lo seguirán siendo. A mis tutores y a los que no los son pero que me ayudaron como si lo fueran. A mis familiares, a los que están y a los que no.

DEDICATORIA

Maidelys:

A mis abuelas

Yader:

A mis padres

RESUMEN

El Departamento de Supervisión del Centro de Control Estatal de Equipos Médicos (CCEEM), encargado de la regulación de equipos médicos, necesita agilizar un gran número de procesos de los que llevan a cabo. Para ello es necesario automatizar buena parte de las actividades que hoy se realizan de forma manual. Para esto se requiere de un sistema informático capaz de gestionar la información de dicho Departamento.

Este trabajo tiene como objetivo fundamental: el análisis, diseño e implementación de una aplicación Web para la gestión de la información del Departamento de Supervisión del CCEEM que cumpla con los requerimientos funcionales y no funcionales exigidos por el usuario. Con el desarrollo de la aplicación se propone mejorar el desempeño en el control de auditorías o registros mediante la automatización de la gestión de la información de cada uno de los procesos del Departamento de Supervisión.

Para el desarrollo de la solución se emplea el framework Symfony siguiendo la estructura del patrón arquitectónico Modelo Vista Controlador (MVC).

PALABRAS CLAVE

Gestión de la información, Symfony, patrón Modelo Vista Controlador, regulación de equipos médicos, aplicación Web.

TABLA DE CONTENIDO

INTRODUCCIÓN	1
CAPÍTULO 1 FUNDAMENTACIÓN TEÓRICA	7
1.1. Conceptos básicos relacionados con la gestión de información.	7
1.2. Actualidad y tendencia a nivel mundial de los sistemas de gestión de información.	10
1.2.1. Herramientas empleadas en la gestión de información.....	11
1.3. Actualidad y tendencias de los sistemas de gestión de información en Cuba.....	13
1.3.1. Perspectivas de desarrollo para los sistemas de gestión de información en el país	16
1.4. Necesidad de crear un sistema para la gestión de la información del Departamento de Supervisión del CCEEM.....	17
1.5. Metodología, tecnologías y herramientas empleadas en la solución	18
1.5.1. Acerca de la Arquitectura de Software	18
1.5.2. Roles involucrados en el desarrollo	23
1.5.3. Tecnologías y herramientas.....	28
1.6. Conclusiones.....	37
CAPÍTULO 2 CARACTERÍSTICAS DEL SISTEMA	38
2.1. Objeto de estudio	38
2.2. Modelo del negocio	40
2.2.1. Actores del negocio	40
2.2.2. Trabajadores del negocio	41
2.2.3. Diagrama de casos de uso del negocio	42
2.2.4. Realización de los casos de uso del negocio.....	42
2.2.5. Diagrama de clases del Modelo de objetos	51
2.2.6. Reglas del negocio	51
2.3. Requerimientos. Modelo del sistema.....	52
2.3.1. Especificación de requisitos	53
2.3.2. Justificación de los actores del sistema.	60
2.3.3. Diagrama casos de uso del sistema	61
2.3.4. Listado de los casos de uso del sistema.....	63

2.4. Conclusiones.....	68
CAPÍTULO 3 DISEÑO DEL SISTEMA	69
3.1. Descripción de estilos arquitectónicos y patrones de diseño.	69
3.2. Modelo del Diseño	71
3.2.1. Realización de casos de uso del diseño	71
3.2.2. Modelo de Datos	79
3.2.3. Modelo de Despliegue	80
3.3. Tratamiento de errores	81
3.4. Conclusiones.....	84
CAPÍTULO 4 IMPLEMENTACIÓN.....	85
4.1. Modelo de Implementación	85
4.1.1. Diagrama de componentes.....	85
4.2. Código fuente.....	88
4.2.1. Estándar de codificación	88
4.2.2. Ejemplo de código fuente	89
4.2.3. Pantallas principales de la aplicación	90
4.3. Validación a nivel de desarrollador	92
4.3.1. Resultados de pruebas exploratorias.....	94
4.4. Conclusiones.....	97
CONCLUSIONES	98
RECOMENDACIONES	99
REFERENCIAS BIBLIOGRÁFICAS	100
BIBLIOGRAFÍA.....	103
ANEXOS	107
GLOSARIO DE TÉRMINOS	160

ÍNDICE DE TABLAS

Tabla 1 Fases de RUP y los hitos correspondientes a cada una.....	23
Tabla 2 Roles y artefactos por cada flujo de trabajo de RUP.....	26
Tabla 3 Justificación de los actores del negocio.....	41
Tabla 4 Justificación de los trabajadores del negocio.....	42
Tabla 5 Descripción textual resumida del caso de uso del negocio Programar Auditoría.....	43
Tabla 6 Descripción textual resumida del caso de uso del negocio Realizar Auditoría.....	43
Tabla 7 Descripción textual resumida del caso de uso del negocio Atender Demanda.....	44
Tabla 8 Descripción textual resumida del caso de uso del negocio Aprobar Inscripción/Reinscripción del Fabricante.....	45
Tabla 9 Descripción textual resumida del caso de uso del negocio Brindar Información.....	45
Tabla 10 Justificación de los actores del sistema.....	60
Tabla 11 Descripción textual resumida del caso de uso del sistema Visualizar Expediente de Fabricante.....	63
Tabla 12 Descripción textual resumida del caso de uso del sistema Gestionar RAPAC.....	63
Tabla 13 Descripción textual resumida del caso de uso del sistema Gestionar RPA.....	64
Tabla 14 Descripción textual resumida del caso de uso del sistema Gestionar RPC.....	65
Tabla 15 Descripción textual resumida del caso de uso del sistema Gestionar planilla LCCF.....	65
Tabla 16 Descripción textual resumida del caso de uso del sistema Gestionar planilla Plan de Auditoría.....	66
Tabla 17 Descripción textual resumida del caso de uso del sistema Gestionar planilla IA.....	67
Tabla 18 Descripción textual resumida del caso de uso del sistema Gestionar planilla Registro de Asistencia.....	67
Tabla 19 Descripción textual resumida del caso de uso del sistema Generar aviso.....	68
Tabla 20 Resumen de los resultados de las pruebas a nivel de desarrollador.....	96
Tabla 21 Descripción textual del caso de uso del negocio Programar Auditoría.....	108
Tabla 22 Descripción textual del caso de uso del negocio Realizar Auditoría.....	112
Tabla 23 Descripción textual del caso de uso del negocio Atender Demanda.....	113
Tabla 24 Descripción textual del caso de uso del negocio Aprobar Inscripción/Reinscripción del Fabricante.....	115
Tabla 25 Descripción textual del caso de uso del negocio Brindar Información.....	115
Tabla 26 Descripción textual CU Visualizar Expediente de Fabricante.....	119
Tabla 27 Descripción textual CU Gestionar RAPAC.....	124

Tabla 28 Descripción textual CU Gestionar RPA	128
Tabla 29 Descripción textual CU Gestionar RPC.....	132
Tabla 30 Descripción textual CU Gestionar planilla LCCF.....	137
Tabla 31 Descripción textual CU Gestionar planilla Plan de Auditoría	141
Tabla 32 Descripción textual CU Gestionar planilla IA.....	146
Tabla 33 Descripción textual CU Gestionar planilla Registro de Asistencia	151
Tabla 34 Descripción textual CU Generar aviso	151

ÍNDICE DE FIGURAS

Figura 1 Figura de RUP en dos dimensiones. Se representa el proceso en el que se grafican los flujos de trabajo y las fases y muestra la dinámica expresada en iteraciones y puntos de control.	22
Figura 2 Diagrama de casos de uso del negocio.....	42
Figura 3 Diagrama de actividades CUN Programar Auditoría.....	46
Figura 4 Diagrama de actividades CUN Realizar Auditoría	47
Figura 5 Diagrama de actividades CUN Atender Demanda.....	48
Figura 6 Diagrama de actividades CUN Aprobar Inscripción/Reinscripción del Fabricante	49
Figura 7 Diagrama de actividades CUN Brindar Información	50
Figura 8 Diagrama de clases del Modelo de objetos para el trabajador del negocio Auditor Líder	51
Figura 9 Diagrama de casos de uso del sistema.....	62
Figura 10 Diagrama de clases del diseño CU Gestionar RAPAC	73
Figura 11 Diagrama de clases del diseño CU Gestionar planilla LCCF	73
Figura 12 Diagrama de clases del diseño CU Gestionar planilla IA.....	74
Figura 13 Diagrama de clases del diseño CU Generar aviso	74
Figura 14 Diagrama de clases persistentes	75
Figura 15 Diagrama de secuencia CU Gestionar RPA. Escenario Modificar.	76
Figura 16 Diagrama de secuencia CU Gestionar planilla LCCF. Escenario Insertar.	77
Figura 17 Diagrama de secuencia CU Gestionar aviso. Escenario Eliminar.	78
Figura 18 Diagrama de secuencia CU Gestionar RAPAC. Escenario Imprimir.	78
Figura 19 Modelo de Datos.....	79
Figura 20 Modelo de Despliegue	80
Figura 21 Validación de formularios.	82
Figura 22 Interfaz de la aplicación. Muestra un menú de selección.....	83
Figura 23 Mensajes de confirmación mostrados al usuario.	83
Figura 24 Diagrama de componentes CU Gestionar RAPAC	86
Figura 25 Diagrama de componentes CU Gestionar planilla LCCF	86
Figura 26 Diagrama de componentes CU Gestionar planilla IA.....	87
Figura 27 Diagrama de componentes CU Generar aviso.	87
Figura 28 Ejemplo de código fuente. Método executeAviso.	90
Figura 29 Interfaz de la aplicación. Muestra el menú de inicio y los accesos directos.	91
Figura 30 Interfaz de la aplicación. Muestra el paginado y los filtros.....	92

Figura 31 Ejemplo de código fuente que implementa una prueba unitaria.	93
Figura 32 Ejemplo de código fuente que implementa una prueba funcional.	94
Figura 33 Resultado de una prueba de unidad que indica error.	94
Figura 34 Resultado de una prueba de unidad que indica éxito.	95
Figura 35 Resultado de una prueba funcional que indica errores.	95
Figura 36 Resultado de una prueba funcional que indica éxito.	95
Figura 37 Resultado de ejecutar el comando test:all que muestra el resultado de todas las pruebas definidas.	96
Figura 38 Estructura y comportamiento de un sistema informático	107
Figura 39 Diagrama de clases del Modelo de objetos para el trabajador del negocio Departamento de Documentación	116
Figura 40 Diagrama de clases del Modelo de objetos para el trabajador del negocio Jefe DS.....	116
Figura 41 Diagrama de clases del Modelo de objetos para el trabajador del negocio Especialista DS.....	117
Figura 42 Patrón CRUD	117
Figura 43 Patrón Login.....	117
Figura 44 Patrón Múltiples actores.....	118

INTRODUCCIÓN

Al triunfo de la Revolución Cubana en 1959, el gobierno revolucionario inició el desmantelamiento del sistema político neocolonial. En la salud se creó un sistema integral que va desde el médico de la familia y los policlínicos hasta hospitales especializados y centros de investigación. Así, la asistencia médica gratuita forma una red que cubre la atención a toda la población. A lo largo de estas décadas se han obtenido extraordinarios logros, si tenemos en cuenta las duras condiciones y restricciones por las que ha pasado el país a raíz del bloqueo económico impuesto por Estados Unidos, pero que sin embargo no constituyeron obstáculos suficientes para impedir que Cuba se colocara como punto de referencia de este sector a nivel mundial.

El Ministerio de Salud Pública (MINSAP) es el Organismo rector del Sistema Nacional de Salud. Encargado de dirigir, ejecutar y controlar la aplicación de la política del Estado y del Gobierno en cuanto a la Salud Pública, el desarrollo de las Ciencias Médicas y la Industria Médico Farmacéutica. Una de las funciones rectoras del MINSAP es: Ejercer la evaluación, el registro, la regulación y el control de los medicamentos de producción nacional y de importación, equipos médicos y material gastable y otros de uso médico. (1)

Algunos de los aspectos que han evidenciado la necesidad de establecer programas de regulación de equipos y dispositivos médicos son: la complejidad tecnológica en los equipos y dispositivos, un mercado más global y competitivo, el incremento de oferta de equipos usados y re-manufacturados, la donación de equipos y dispositivos, el re-uso de dispositivos de un solo uso, el uso cada vez mayor de equipos y dispositivos en consultorios y a nivel domiciliario, el débil soporte de servicio técnico posterior a la venta, la necesidad de reportar los efectos adversos y la notificación de problemas con equipos y dispositivos.

El primer Programa Regulador de Equipos Médicos data de los años setenta, cuando el Congreso de Estados Unidos en 1976 lleva a la Administración de Medicamentos y Alimentos más conocida por las siglas en inglés FDA (Food and Drug Administration), por medio de la Enmienda de Equipos Médicos de 1976, al uso de los controles necesarios para garantizar la seguridad y efectividad de los productos.

De esta forma, en este período toman carácter propio las regulaciones para equipos médicos, marcando diferencias respecto a otras tecnologías sanitarias, como los medicamentos. Luego en esta misma década, Canadá acredita sus regulaciones para equipos médicos con la Ley de Alimentos y

Medicamentos, y de esta forma, inicia un análisis de su Programa Regulador en 1991. En 1994, la Organización Panamericana de la Salud (OPS) como parte de la táctica de la Reforma Sanitaria, ha venido aconsejando a los países de América Latina y el Caribe el establecimiento y fortalecimiento de los sistemas reguladores de equipos médicos, con el propósito de garantizar la seguridad de los equipos que están a disposición de la población.

La Comisión desarrolló sus tareas hasta noviembre de 1991, cuando se concluyó con la realización del documento base del Programa Regulatorio Cubano, al que se llamó “Reglamento para la Evaluación Estatal y Registro de Equipos Médicos”, así como la propuesta que a instancias del MINSAP se hacía necesario instaurar el Órgano Regulador responsable del Registro en este nivel. El documento base fue difundido nacionalmente desde 1991 hasta mayo de 1992. Se aprobó y se puso en práctica durante junio – diciembre de 1992. (2)

Para darle cumplimiento a lo antes referido y como consecuencia del desarrollo alcanzado y de las nuevas tecnologías adquiridas por el Sistema Nacional de Salud (SNS) se funda el 30 de enero de 1992 el Centro de Control Estatal de Equipos Médicos (CCEEM), institución regulatoria de la Salud Pública Cubana que garantiza la efectividad y seguridad de los equipos médicos utilizados en el SNS. Se hace necesario a su vez el establecimiento de un Registro Sanitario de Equipos Médicos sustentado en el Reglamento de la Evaluación Estatal y del Registro de Equipos Médicos y mediante la Resolución Ministerial No. 110 del 18 de junio de este mismo año se pone en vigor el Reglamento de Evaluación Estatal y se faculta al CCEEM.

La función fundamental del CCEEM es velar por que la tecnología biomédica que se utiliza en los centros que pertenecen al SNS cumpla con los requisitos de seguridad y eficacia establecidos en los documentos del Programa Regulatorio Cubano de Equipos Médicos y con las normas internacionales que existen para este tipo de tecnología y de esta forma garantizar la seguridad, eficacia y efectividad de los equipos médicos que son utilizados en el SNS, ya sean importados o de factura nacional.

Por esta razón la labor del CCEEM alcanza a todas las instituciones del SNS, fabricantes e importadores de equipos médicos y a cualquier institución que cuente con ellos, y se dispone a brindarles una amplia gama de servicios científico-técnicos. Estos servicios están vinculados a la prevención de riesgos, daños y fallos en los equipos médicos a través de trabajos de evaluación y registro, inspección, vigilancia, normalización y otras formas de investigación y control.

Desde el año 1999 Cuba se ha visto envuelta en el quehacer constructivo de la Batalla de Ideas, con más de 200 programas en marcha, muchos de ellos encaminados a mejorar en específico el área de salud. El programa de reequipamiento tecnológico beneficia a la totalidad de los servicios primarios y secundarios con que cuenta el país, con la gran ventaja adicional de acercar los servicios médicos más importantes y de calidad a los hogares y lugares de residencia de la población.

Es por esto que el país ha realizado una masiva inversión en la salud y actualmente constituye una actividad importante la introducción de equipos médicos al SNS, tanto de índole de importación como de carácter nacional, para lo cual resulta imprescindible el otorgamiento del registro sanitario emitido por el CCEEM.

Dentro de este centro está enmarcado el Departamento de Supervisión dedicado a contribuir indirectamente a la salud pública y el bienestar del pueblo, mediante el desarrollo de normativas y documentos relacionados con el campo de los equipos médicos. Su actividad constituye el soporte científico técnico fundamental del Programa Regulador de Equipos Médicos. Su trabajo fomenta la adopción de normas internacionales y el desarrollo de normas cubanas.

El Departamento de Supervisión es el encargado de las auditorías de calidad a los fabricantes de equipos médicos con vistas a garantizar el cumplimiento de los requisitos regulatorios establecidos en la Regulación sobre Requisitos de Calidad para los fabricantes de equipos médicos y de la verificación de los certificados de calidad entregados por fabricantes extranjeros. Es aplicable para todos los fabricantes inscritos o que soliciten la inscripción en el CCEEM. Las auditorías también pueden efectuarse a las instalaciones del productor cuando el fabricante de equipos médicos no produzca directamente el equipo.

De estas actividades se deriva una función importante: el trabajo con la documentación de cada uno de los fabricantes inscritos en el CCEEM así como la actualización de los documentos generales del Departamento de Supervisión.

En los últimos años, se ha visto un incremento del número de fabricantes que solicitan inscripción en el CCEEM y dado que este centro es el único de su tipo en el país el Departamento de Supervisión se ve obligado a manejar gran cantidad de documentos que contienen información relacionada a los fabricantes e instituciones inscritas en el CCEEM. Además cada vez es mayor el cúmulo de información obtenida de las actividades que realiza el departamento. Dicha información se almacena en copia dura y constantemente se necesita consultar y actualizar, funciones que se realizan

manualmente, lo que resulta engorroso para los especialistas del departamento y ocasiona la pérdida de datos, la pérdida de tiempo y que se archive información errónea. Estos factores impiden un mejor desempeño de estas actividades que son de vital importancia para el SNS. La situación descrita conlleva a plantear el **problema científico**: ¿Cómo contribuir a la mejora de la gestión de la información del Departamento de Supervisión del CCEEM?

La investigación tiene como **objeto de estudio** el proceso de gestión de la información enmarcado en el **campo de acción** el proceso de gestión de la información del Departamento de Supervisión del CCEEM.

Objetivo General:

Desarrollar el análisis, diseño e implementación de una aplicación informática para la gestión de la información del Departamento de Supervisión del CCEEM.

Objetivos Específicos:

1. Definir funcionalidades del módulo de “Gestión de la información del Departamento de Supervisión del CCEEM versión 1.0.”
2. Diseñar el módulo de “Gestión de la información del Departamento de Supervisión del CCEEM versión 1.0.”
3. Implementar los componentes definidos.

Posibles resultados:

1. Entrega de la versión funcional del módulo de “Gestión de la información del Departamento de Supervisión de CCEEM versión 1.0.”
2. Entrega de toda la documentación de acuerdo con los lineamientos mínimos de calidad establecidos por la universidad y los artefactos generados a partir de la metodología empleada.

Tareas a realizar:

Para el cumplimiento de estos objetivos se trazaron las siguientes tareas:

1. Investigación de las tendencias y tecnologías actuales para la gestión de la información.
2. Estudio de los procesos e información relacionados con el Departamento de Supervisión del CCEEM posibles a automatizar a través de entrevistas al personal.
3. Definición de los requisitos funcionales y no funcionales de la aplicación a implementar.
4. Selección de los artefactos que se obtendrán según la metodología de desarrollo que se seleccione.
5. Diseño del módulo de “Gestión de la Información del Departamento de Supervisión de CCEEM versión 1.0.”
6. Diseño de una base de datos para la gestión de la información.
7. Implementación del módulo de “Gestión de la Información del Departamento de Supervisión de CCEEM versión 1.0.”

La investigación sigue la siguiente estructura:

Capítulo 1: Fundamentación Teórica

Presentación el estado del arte. Mención de la metodología que se seguirá para alcanzar el objetivo del trabajo. Presentación de las herramientas y tecnologías que se utilizarán en el desarrollo del trabajo.

Capítulo 2: Características del sistema

Breve descripción del problema. Estudio de los principales procesos que se llevan a cabo en el Departamento de Supervisión a través del Modelo del negocio, principalmente realizaciones de casos de uso del negocio. Definición de requisitos funcionales y no funcionales descritos a través de diagramas de casos de uso del sistema y descripciones textuales de casos de uso del sistema.

Capítulo 3: Diseño del sistema

Representación de la realización de casos de uso del sistema mediante diagramas de clases del diseño e interacción, así como del modelo de datos, la descripción general de cada una de sus tablas y el modelo de despliegue.

Capítulo 4: Implementación

Representación de la implementación del sistema informático en cuestión mediante la representación del Modelo de Implementación que incluye, componentes, paquetes de componentes, subsistemas de implementación y sus respectivas relaciones representadas en el diagrama de componentes. Validación del sistema a través de pruebas exploratorias.

CAPÍTULO 1 FUNDAMENTACIÓN TEÓRICA

En el presente capítulo se realiza el estudio teórico que fundamenta la investigación. Se realiza un estudio de la evolución hasta la actualidad de los sistemas de gestión de la información en el mundo y las tendencias que han surgido durante su desarrollo. Este estudio es aterrizado en el ámbito mundial y nacional. Se incluye además el análisis de las tecnologías empleadas en la solución y los roles involucrados, así como los artefactos desarrollados por cada uno de ellos.

1.1. Conceptos básicos relacionados con la gestión de información.

El término **información** es ampliamente empleado en este capítulo por su importancia en la investigación. Su concepto, aunque pueda parecer similar, no es el mismo que el de datos y es importante que se conozca la diferencia. Para esto se enuncia **datos** como realidades concretas en su estado primario que representan hechos reales que poseen escaso valor más allá de su sola existencia. (3 pág. 5) A partir de este concepto inicial se afirma que pueden establecerse relaciones entre los datos para obtener una valiosa información. Entonces **información** se puede definir como el conjunto de datos organizados de forma tal que adquieren un valor adicional más allá del propio. (3 pág. 5) El concepto “información” se aplica generalmente con tres acepciones:

a) Información como Proceso: asociado fundamentalmente al proceso de informarse. Generalmente a través de los medios de difusión o la conversación con otras personas se produce el proceso de informarse, que ocurre intencionalmente o no, pero que está indisolublemente asociado al proceso de comunicación. La información es el contenido del proceso de comunicación.

b) Información como Conocimiento: principalmente concebida como el contenido del proceso de información. Se refiere a la expresión del conocimiento y que se comunica verbalmente, o en algún soporte escrito, ya sea en papel o en medios electrónicos, o en medios audiovisuales. La información aumenta o reduce la incertidumbre.

c) Información como Objeto: comúnmente se emplea para referirse a objetos, generalmente documentos. Esta licencia se aplica como extensión a la posesión de la cualidad de transmitir conocimientos o ser instructivos e informativos. (4)

La información es objeto de tratamiento en forma permanente, y al igual que la materia no se destruye, sino que se transforma a través de todo un ciclo de vida. Cuando se hace referencia al “ciclo de vida de la información”, generalmente se está haciendo referencia a la “información como objeto”

ya que son los documentos en formato tradicional o electrónico los que pueden llegar a desaparecer o ser destruidos como fin de su ciclo de vida. (4)

Otros conceptos que vienen a complementar los anteriormente mencionados son **gestión** y **gestión de la información**. Aunque sus definiciones pueden ser variadas, en esta investigación se ha decidido emplear el concepto de **gestión** como el proceso mediante el cual se obtiene, despliega o utiliza una variedad de recursos básicos para apoyar los objetivos de la organización. Es entonces lógico que se defina **gestión de la información** como proceso mediante el cual se obtienen, despliegan o utilizan recursos básicos (económicos, físicos, humanos, materiales) para manejar información dentro y para la sociedad a la que sirve. Tiene como elemento básico la gestión del ciclo de vida de este recurso y se desarrolla en cualquier organización. (5) Se considera importante además hacer mención de las funciones que este proceso desarrolla.

Funciones de la gestión de información:

- 1) Determinar necesidades internas de información, relativas a las funciones, actividades y procesos administrativos de la organización y a su satisfacción.
- 2) Optimizar el flujo organizacional de la información y el nivel de la comunicación.
- 3) Manejar eficientemente los recursos organizacionales de información, mejorar las inversiones sucesivas en los mismos y optimizar su aprovechamiento.
- 4) Entrenar a los miembros de la organización en el manejo o la utilización de los recursos informacionales.
- 5) Contribuir a modernizar u optimizar las actividades organizativas y los procesos administrativos relacionados con los mismos.
- 6) Garantizar la calidad de los productos de la organización y asegurar su diseminación efectiva.
- 7) Determinar las necesidades de información externa de la organización y satisfacerlas. (4)

Es útil mencionar además una serie de conceptos que pueden resultar esclarecedores vinculados a la rama de la informática. El primero y más básico es el concepto de **sistema** que se define como un conjunto de elementos o componentes que interactúan entre sí para cumplir ciertas y determinadas metas. (6)

Podrían darse muchas definiciones de sistema informático. Dos de ellas son las siguientes:

- Conjunto de elementos que hacen posible el tratamiento automático de la de información.
 - Sistema compuesto de equipos y de personal que realiza funciones de entrada, proceso, almacenamiento, salida y control con el fin de llevar a cabo una secuencia de operaciones con datos.
- (6)

En términos genéricos se puede afirmar que el sistema informático es un conjunto de elementos interrelacionados entre sí y relacionados a su vez con el sistema global en que se encuentra que pretende conseguir unos fines determinados. Los elementos constitutivos de un sistema informático serán físicos, lógicos y humanos.

Estructuralmente un sistema se puede dividir en partes pero, funcionalmente es indivisible, ya que si se dividiera perdería alguna de sus propiedades esenciales. Un sistema informático está compuesto por:

- Componente físico: que constituye el hardware del sistema informático que lo conforman, básicamente, los ordenadores, los periféricos y el sistema de comunicaciones. Los componentes físicos proporcionan la capacidad y la potencia de cálculo del sistema informático.
 - Componente lógico: que constituye el software del sistema informático y lo conforman, básicamente, los programas, las estructuras de datos y la documentación asociada. El software se encuentra distribuido en el hardware y lleva a cabo el proceso lógico que requieren los datos.
 - Componente humano: constituido por todas las personas participantes en todas las fases de la vida de un sistema informático (diseño, desarrollo, implantación, explotación). Este componente humano es sumamente importante ya que los sistemas informáticos están desarrollados por humanos y para uso de humanos. El comportamiento de un sistema informático se puede representar gráficamente.
- (6) (Anexo 1: Estructura y comportamiento de un sistema informático.)

A partir de los conceptos expuestos se puede deducir que un sistema informático para la gestión de la información es aquel sistema que implementa funcionalidades para desarrollar el trabajo automático de la información, ya sea para su almacenamiento, organización, procesamiento de datos, visualización y obtención. De ahora en adelante y en toda la investigación es referido como **sistema de gestión de información**.

1.2. Actualidad y tendencia a nivel mundial de los sistemas de gestión de información.

El papel que los sistemas de gestión de información juegan en las empresas ha experimentado un cambio considerable en estos últimos años, pasando de ser simples herramientas de tratamiento de datos para convertirse en la columna vertebral de cualquier organización, tanto a nivel interno como en lo referente a las relaciones con el exterior: clientes, proveedores, administración o la sociedad en general.

A través de una adecuada gestión de la información se facilita la innovación, el desarrollo de nuevos productos o servicios; se mejora la eficiencia en el uso de los recursos, la calidad del servicio o la toma de decisiones.

Los servicios de información y documentación accesibles mediante servidores Web, están aumentando de forma exponencial. La evolución de los servicios Web desde hace más de quince años ha ido produciendo la sustitución de páginas y documentos estáticos por documentos generados dinámicamente, debido a la interacción del usuario con la lógica de procesos y flujos de trabajo definida por los creadores del servicio y a la disponibilidad de cada vez mayores repositorios de información.

Evidentemente, se ha ido pasando progresivamente de un concepto de publicación de páginas Web, bastante simple en su origen, a esquemas más complejos y diferenciados, fundamentados en procedimientos y técnicas basados en la gestión de información.

Esta evolución, que se ha acelerado durante la primera mitad de la década de 2000, ha tenido un impacto no sólo en los métodos y técnicas de gestión de información, sino también en la propia tecnología para gestión de información y, en consecuencia, en el mercado de productos y servicios.

(7)

Si bien en la segunda mitad de la década de los 90 se podía diferenciar entre productos para gestión documental, para recuperación de información, etc., desde el año 2000 se ha producido una convergencia entre todas las plataformas, de forma que en la actualidad se pueden encontrar soluciones que pretenden ser globales y ofrecer soporte a todo el proceso de gestión de información en una organización. Las herramientas para este trabajo han recibido la denominación de sistemas de

gestión de contenidos (o *Content Management Systems*, CMS), y se han integrado con los sistemas de gestión documental y con los de recuperación de información.

A pesar de sus semejanzas no es correcto afirmar que existe una igualdad directa entre los sistemas de gestión de contenidos y los sistemas de gestión de información. Se pueden encontrar sistemas de gestión de contenidos que no implementan las funcionalidades para el trabajo documental que serían deseables, y viceversa. A través del estudio de las características y funcionalidades que presenten cada herramienta se puede determinar si es la adecuada para una situación problemática dada.

La variedad de sistemas de gestión de información en el mundo actual es difícil de especificar. Cada empresa o negocio tiene características y procesos determinados dentro de sus flujos de trabajo que requieren funcionalidades diferentes en cada sistema empleado. Existen herramientas de gestión de información especializadas en las necesidades de varias ramas de la economía, sin embargo en ocasiones resultan demasiado genéricas para ser implantadas en una empresa con procesos específicos, sumándole a esto que la mayoría son consideradas software propietario por lo que habría que tener en cuenta el costo de obtención. Surge entonces la necesidad de continuar la producción de estos sistemas a ser implantados en las empresas que se animen a la automatización, para lo que se debe estudiar a fondo el flujo de los procesos que la misma sustenta y a partir de los resultados arrojados por estos estudios elaborar una propuesta para el sistema a desarrollar que se ajuste a las necesidades y expectativas del cliente.

1.2.1. Herramientas empleadas en la gestión de información

El estudio de herramientas de gestión de la información existentes facilita la decisión a la hora de seleccionar cuáles emplear en un caso determinado o cuáles se ajustan a las necesidades de un cliente en específico. Algunas de estas herramientas son empleadas actualmente por grandes empresas y organizaciones con una amplia aceptación dada, principalmente, por los servicios que brinda y las características que implementa.

InterLan System

Se trata de un sistema de gestión de la información, diseñado para las medianas y grandes empresas que necesitan clasificar, archivar, publicar, traducir y buscar la información de una manera personalizada, rápida y sencilla en diferentes idiomas.

El uso de **InterLan System**, permite realizar de manera automática una serie de labores, dentro del proceso de gestionar la información y estructurar esta en una Intranet Corporativa.

Las funciones que lleva a cabo son:

- Crea automáticamente una Intranet Corporativa o se integra en una ya existente.
- Asegura el acceso a la información a través de un perfil de usuario.
- Personaliza el acceso a todo el sistema de documentación.
- Clasifica la información organizándola de forma departamental.
- Publica la información de forma automática.
- Recupera la información de manera conceptual, independientemente del idioma en que se encuentre.
- Traduce automáticamente a varios idiomas la información requerida.
- Se integra dentro de los estándares informáticos actuales, soportando plataformas: Windows 95/98 y NT. (8)

NEMS

El NEMS (Sistema de Gestión de Noticias y Acontecimientos) es un dinámico sistema multilingüe de publicación Web, especialmente adecuado para los sitios en la Web con listas y nuevos sectores que exigen actualización constante.

Tres usuarios participan en las actividades editoriales normales del NEMS, pero este puede adaptarse a las necesidades de cualquier institución o departamento.

El primer editor incorpora en el NEMS nuevos elementos, información sobre acontecimientos y cursos, que el sistema convierte automáticamente al formato HTML. La información se envía por la interfaz de la red al traductor o directamente al editor final. El traductor elabora la versión en el idioma correspondiente y la remite al editor final. El editor final revisa la información, la devuelve al primer editor o la publica directamente en la página Web, sin necesidad de que otro especialista la convierta al formato HTML.

Otra de las características que presenta es la de ser un sistema descentralizado en el que los usuarios autorizados pueden incorporar elementos desde cualquier lugar del mundo. La página Web (dinámica) creada por el NEMS se pone automáticamente al día al publicarse nuevos elementos, información sobre acontecimientos y cursos. Contiene, además, un sistema de búsqueda que permite al usuario buscar noticias e información archivada de acontecimientos o cursos a través de palabras clave, fechas, nombres de los países o temas. El sistema es multilingüe, de modo que el contenido puede incorporarse y organizarse en árabe, chino, español, francés e inglés. (9 pág. 10)

EIMS

El EIMS (Sistema de Gestión de la Información) es un sistema integral para la gestión de las actividades de publicación electrónica y para la clasificación de palabras clave y metadatos en las publicaciones sobre agricultura, páginas Web, imágenes y vídeos. Se comenzó a utilizar el EIMS en 1999, actualmente la mayor parte de los departamentos de la Organización de las Naciones Unidas para la Agricultura y la Alimentación (FAO), así como su red de oficinas en todo el mundo, utilizan este sistema para organizar la vasta producción de información de la Organización.

Es un sistema multilingüe, de modo que los autores pueden insertar un registro en árabe, chino, español, francés o inglés. La utilización de diccionarios multilingües asegura la clasificación del objeto independiente del idioma utilizado para asignar las palabras claves y las categorías. Este software puede adaptarse a las necesidades internas y a la estructura de cualquier organización o institución y actualmente está al alcance de todas las instituciones agrarias, ministerios, ONG (Organizaciones No Gubernamentales) y asociaciones de agricultores.

El avance en las tecnologías, el incremento del uso de la Web y el apoyo cada vez mayor de estos sistemas informáticos al desarrollo común de las empresas predice que el avance de los mismos va a crecer constantemente al igual que su variedad siguiendo el comportamiento que ha venido teniendo en las últimas décadas. (9 pág. 11)

1.3. Actualidad y tendencias de los sistemas de gestión de información en Cuba

Actualmente existen varios factores que propician la implantación de los sistemas de Gestión de la Información en Cuba, en función del desarrollo social y económico que emprende el país. Se destacan los siguientes:

1. Informatización de la sociedad cubana

La informatización de la sociedad cubana es la utilización ordenada y masiva de las tecnologías de la información y las comunicaciones en todas las esferas de la sociedad cubana, en su esfuerzo por lograr cada vez más eficacia y eficiencia en todos los procesos y por consiguiente mayor generación de riqueza y aumento en la calidad de vida de los ciudadanos.

A causa de la implantación de estos métodos se han visto resultados tangibles que han mejorado la infraestructura tecnológica del país.

Principales resultados alcanzados en el país

- Creación de una infraestructura científica y técnica que permite contar con un conjunto de centros de investigación que desarrollan resultados de reconocido prestigio internacional.
- Fortalecimiento de la red de los Centros de Enseñanza Superior y el impulso dado a la investigación en ellos.
- Creación del Ministerio de Informática y las Comunicaciones.
- Sistema de salud, de amplia base técnica, considerable integralidad y acceso irrestricto para toda la población, el cual constituye una de las expresiones más nítidas de los vínculos entre el progreso científico y el progreso social en Cuba.
- Logro de una capacidad de respuesta rápida del sistema de ciencia y técnica.
- Creación de diversos centros de investigación en el campo de las ciencias sociales y de difusión de la cultura nacional y esclarecimiento de la problemática histórica, económica, política y filosófica de la sociedad.
- Utilización de las nuevas tecnologías y conocimientos que devienen rasgos fundamentales de la revolución científico-técnica. (10)

2. Sistema de Ciencia e Innovación Tecnológica

El Sistema de Ciencia e Innovación Tecnológica es la forma organizativa que permite la implantación en forma participativa de la política científica y tecnológica que el estado cubano y su sistema de instituciones establecen para un período determinado, de conformidad con la estrategia de desarrollo

económico y social del país y de la estrategia de ciencia y tecnología que es parte consustancial de ésta.

Este sistema cubre un amplio espacio que va desde la asimilación, generación y acumulación de conocimientos hasta la producción de bienes y servicios y su comercialización, pasando, entre otras, por actividades tales como: las investigaciones básicas, las investigaciones aplicadas, los trabajos de desarrollo tecnológico, desarrollo social y de gestión, así como el empleo de modernas técnicas gerenciales.

Un objetivo central del Sistema de Ciencia e Innovación Tecnológica es contribuir de forma determinante al desarrollo sostenible de la economía cubana y al alcance por ésta de un espacio cada vez mayor en el mercado internacional, para lo cual se deberán generar nuevos conocimientos, desarrollar la tecnología, y transformar los avances científicos y los logros tecnológicos en productos competitivos con éxitos comerciales. (11)

3. El Modelo de Gestión Empresarial denominado Perfeccionamiento Empresarial

El Perfeccionamiento Empresarial es un proceso encaminado a la búsqueda de la máxima eficiencia y competitividad de la empresa estatal cubana, sobre la base de otorgarle facultades y establecer principios y procedimientos que estimulen la iniciativa y la creatividad de dirigentes y trabajadores en las organizaciones. (12)

Este proceso que se experimenta en la empresa cubana, asimila los procedimientos y técnicas más modernas de la gerencia empresarial en el mundo, ajustado al contexto cubano, trayendo consigo un nuevo sistema de gestión.

4. Política Nacional de Información de Cuba

Cuba se ha esforzado en formular e implementar su Política Nacional de Información, entendiendo esta como la concertación de la acción nacional en torno a la generación, recolección, organización, asimilación y distribución de la información que el país requiera para modernizar y afianzar el proceso de desarrollo social y económico.

Aprobada en marzo de 1999 la Política Nacional de Información, debe contribuir a:

a) Elevar la eficiencia en la producción de bienes y servicios de información para satisfacer la demanda nacional e insertar al país en la industria internacional.

b) Facilitar que el proceso de descentralización de la economía se realice exitosamente, mejorando la toma de decisiones mediante el suministro de información pertinente y oportuna.

c) Favorecer la gestión de todos los sectores mediante la elevación de su competitividad. (13)

5. Decreto Ley 199 / 99. Decreto Ley sobre la seguridad y protección de la información oficial

Esta ley tiene como objetivo establecer y regir el sistema para la Seguridad y Protección de la Información Oficial.

El Sistema para la Seguridad y Protección de la Información Oficial comprende la clasificación y desclasificación de las informaciones, las medidas de seguridad con los documentos clasificados, la Seguridad Informática, la Protección Electromagnética, la Protección Criptográfica, el Servicio Cifrado y el conjunto de regulaciones, medidas, medios y fuerzas que eviten el conocimiento o divulgación no autorizados de esta información. (14)

Aunque cada uno de los factores citados incide de modo diferente sobre las organizaciones, su acción tiene implicaciones positivas, incluso a corto plazo. Por otro lado, la acción coordinada y armónica de estos cinco elementos determinan que la perspectiva de las organizaciones cubana para la utilización exitosa de la gestión de la información sea real y beneficiosa. A diferencia de otros países, esto no depende únicamente de las fuerzas propias del mercado porque la sociedad cubana actúa conscientemente y toma las medidas necesarias, en busca del objetivo de la eficiencia y la competitividad de la economía nacional.

1.3.1. Perspectivas de desarrollo para los sistemas de gestión de información en el país

En el país, aún existe falta de claridad en la definición de gestión de la información. Sin embargo la mayoría de las empresas tienen previsto o implementado, algún sistema de gestión de la información.

La automatización es, en sentido general, el rasgo más importante de las actuales transformaciones de la técnica y aunque al país le falta un largo trecho por recorrer, no cabe la menor duda de que las tareas futuras serán cada vez más complejas donde no se puede dejar de lado el cuidado a las consecuencias sociales del progreso científico-técnico, en particular de la utilización de las

TIC(Tecnologías de Información y Comunicación) que garanticen la máxima armonía entre el progreso técnico y el social.

Las perspectivas para la gestión de la información son alentadoras ya que el enfoque gerencial centrado en el conocimiento es aplicable a cualquier tipo de organización y puede funcionar desde el individuo hasta la sociedad lo que brindará resultados mucho más sostenibles para las organizaciones.

Si se tiene en cuenta el análisis realizado, se puede tomar como cierto que la gestión de la información es una excelente herramienta de trabajo para cualquier empresa actual, por el valor que tiene para una organización, el uso adecuado de la información y la insustituible importancia de utilizar y preservar los conocimientos adquiridos por los trabajadores en el transcurso del tiempo, que son a su vez parte estratégica de su patrimonio.

1.4. Necesidad de crear un sistema para la gestión de la información del Departamento de Supervisión del CCEEM.

El estudio presentado hasta el momento ha dejado ver que actualmente en el mundo existen varios sistemas de gestión de información que cuentan con una amplia gama de funcionalidades y servicios. Además se ha podido concluir que por su gran variedad pueden ser empleados en diversas ramas de la economía incluso pueden ser modificados de acuerdo a las necesidades de la empresa. Se adicionan los resultados del estudio de las condiciones satisfactorias existentes en el país para la implantación de sistemas de gestión de la información.

Entonces cabe la interrogante: ¿por qué se plantea la necesidad de desarrollar un sistema de gestión de información para el Departamento de Supervisión del CCEEM? La respuesta está dada por diversos factores. Primeramente se debe mencionar que el flujo de procesos que se llevan a cabo actualmente en el Departamento de Supervisión posee características específicas que no son satisfechas por ninguno de los sistemas de gestión de información existentes. Esto, en gran medida, se debe al uso de planillas y registros definidos por la empresa para el desarrollo de las actividades y procesos del departamento que se han venido definiendo en base a las necesidades y experiencias que han surgido en el transcurso de los años. Estas planillas constan de un formato específico consecuente con la información que se maneja en el departamento. A esto se adiciona que el sistema a implementar deberá contar con funcionalidades de cálculo de estadísticas y gestión de avisos, así

como otras funciones que son definidas únicamente por las características y necesidades del departamento.

Lo anteriormente mencionado está relacionado con las funcionalidades que debe presentar el sistema. Sin embargo, también existen razones relacionadas con las propiedades o cualidades que debe presentar el producto final y que ya están predefinidas por la arquitectura general que rige el proyecto al cual pertenece el módulo a desarrollar. Esta arquitectura ha establecido los patrones y estilos arquitectónicos, tecnologías y herramientas, a emplear en la construcción del producto.

En la siguiente sección se mencionan las tecnologías empleadas en la solución del problema junto con sus principales características.

1.5. Metodología, tecnologías y herramientas empleadas en la solución

1.5.1. Acerca de la Arquitectura de Software

La arquitectura de un software puede concebirse como aquella estructura del programa que cohesiona las funcionalidades más críticas y relevantes (necesarias para el sistema), y que sirve de soporte al resto de funcionalidades finales (necesarias para el usuario).

Una definición más formal: “La arquitectura del software es la organización fundamental de un sistema formada por sus componentes, las relaciones entre ellos y el contexto en el que se implantarán, y los principios que orientan su diseño y evolución”. (IEEE Std 1471-2000)

Se puede incluir que aporta una visión abstracta de alto nivel, posponiendo el detalle de cada uno de los módulos definidos a pasos posteriores del diseño. Es una vista estructural de alto nivel; define un estilo o combinación de estilos para una solución; centra su atención en los requerimientos no funcionales, ya que los requerimientos funcionales se satisfacen mediante el modelado y diseño de la aplicación. Para que la arquitectura se convierta en una herramienta útil dentro del desarrollo y mantenimiento de los sistemas de software es necesario que se cuente con una manera precisa de representarla.

Para la solución se sigue la definición de la arquitectura de software para el Grupo de Desarrollo para la Gestión de Equipos Médicos al cual pertenece el proyecto SACCEM. Una definición de arquitectura, abarca la estructura del programa en términos de los estilos que guiarán el desarrollo de la aplicación, las tecnologías y herramientas en las que se apoyará el desarrollo.

Los estilos arquitectónicos guían a la organización del sistema de software. Estos incluyen reglas y líneas a seguir para la organización de un sistema. El **estilo llamada y retorno** es el más utilizado en los grandes sistemas informáticos. Esta familia de estilos enfatiza la modificabilidad y la escalabilidad. Son los estilos más generalizados en sistemas en gran escala. Miembros de la familia son las arquitecturas de programa principal y subrutina, los sistemas basados en llamadas a procedimientos remotos, los sistemas orientados a objeto y los sistemas jerárquicos en capas.

Los patrones de arquitectura, relacionados a la interacción de objetos dentro o entre los niveles arquitectónicos, se aplican durante la fase de diseño inicial para resolver problemas arquitectónicos, adaptabilidad a requerimientos cambiantes, performance, modularidad y acoplamiento. El **patrón Modelo Vista Controlador (MVC)**, perteneciente al estilo anterior, es muy recomendado para sistemas interactivos.

Es un patrón fácil de explicar y entender que reduce dependencias y potencia la reutilización. También se aprovecha la ventaja que brinda al permitir dividir el trabajo en base a distintos roles (el diseñador gráfico no necesita saber cómo se ha implementado el resto de los componentes). (15)

Proceso de Desarrollo de Software

Un proceso de desarrollo define **quién** está haciendo **qué, cuándo, y cómo** alcanzar un determinado objetivo. Se define también como el conjunto de actividades necesarias para convertir los requisitos de los usuarios en un producto final. (16 pág. XVI) Durante el período de desarrollo de un proyecto de software es necesario un proceso que sirva como guía a todos los interesados en el proyecto, clientes, usuarios y desarrolladores.

Un Proceso de Desarrollo de Software es la definición del conjunto de actividades que guían los esfuerzos de las personas implicadas en el proyecto, a modo de plantilla que explica los pasos necesarios para terminar el proyecto. (16 pág. 4)

Las piedras angulares del proceso de desarrollo del software son: el proyecto, las personas y el producto; siendo las características del cliente, el entorno de desarrollo y las condiciones del negocio, elementos que influyen en el proceso. Existe una estrecha relación entre personas, proyecto, producto y proceso. Estos términos son conocidos como las cuatro "P" en el desarrollo de software.

Personas: Los principales autores de un proyecto software son los arquitectos, desarrolladores, ingenieros de prueba, y el personal de gestión que les da soporte, además de los usuarios, clientes, y

otros interesados. Las personas son realmente seres humanos, a diferencia del término abstracto trabajadores, que se introducirá más adelante.

Proyecto: Elemento organizativo a través del cual se gestiona el desarrollo de software. El resultado de un proyecto es una versión de un producto.

Producto: Artefactos que se crean durante la vida del proyecto, como los modelos, código fuente, ejecutables, y documentación.

Proceso: Un proceso de ingeniería de software es una definición del conjunto completo de actividades necesarias para transformar los requisitos de usuario en un producto. Un proceso es una plantilla para crear proyectos.

Herramientas: Software que se utiliza para automatizar las actividades definidas en el proceso. (16 pág. 13)

Metodología de software

La metodología de software se desarrolla con el objetivo de dar solución a los problemas existentes en la producción de software, que cada vez son más complejos. Estas engloban procedimientos, técnicas, documentación y herramientas que se utilizan en la creación de un producto de software. Una metodología es un proceso.

Se debe mencionar que no existe una metodología de software universal. Las características de cada proyecto (equipo de desarrollo, recursos, etc.) exigen que el proceso sea configurable y las diversas metodologías existentes son aplicables a diferentes proyectos según sus características.

Proceso Unificado de Desarrollo (RUP)

Una de las metodologías de desarrollo más usadas actualmente es RUP. Este proceso está equilibrado por ser el producto final de más de tres décadas de desarrollo y de uso práctico. Su desarrollo como producto sigue un camino que comienza con la publicación en 1987 de su primera versión, Proceso Objectory¹, evolucionando luego al Proceso Objectory de Rational en 1997 hasta el Proceso Unificado de Rational en 1998. Su desarrollo ha recibido influencias de muchas fuentes,

1

“Objectory” es una abreviatura de “Object Factory”, compañía fundada por Ivar Jacobson en 1987

algunas desconocidas hasta para sus principales creadores (Jacobson, Booch, y Rumbaugh). (16 pág. XVIII)

Como RUP es un proceso, en su modelación define como sus principales elementos:

Trabajadores (“quién”): Define el comportamiento y responsabilidades (rol) de un individuo, grupo de individuos, sistema automatizado o máquina, que trabajan en conjunto como un equipo. Ellos realizan las actividades y son propietarios de elementos.

Actividades (“cómo”): Es una tarea que tiene un propósito claro, es realizada por un trabajador y manipula elementos.

Artefactos (“qué”): Productos tangibles del proyecto que son producidos, modificados y usados por las actividades. Pueden ser modelos, elementos dentro del modelo, código fuente y ejecutables.

Flujo de actividades (“Cuándo”): Secuencia de actividades realizadas por trabajadores y que produce un resultado de valor observable.

En RUP se han agrupado las actividades en grupos lógicos definiéndose 9 flujos de trabajo principales. Los 6 primeros son conocidos como flujos de trabajo de ingeniería y los tres últimos como flujos de trabajo de apoyo.

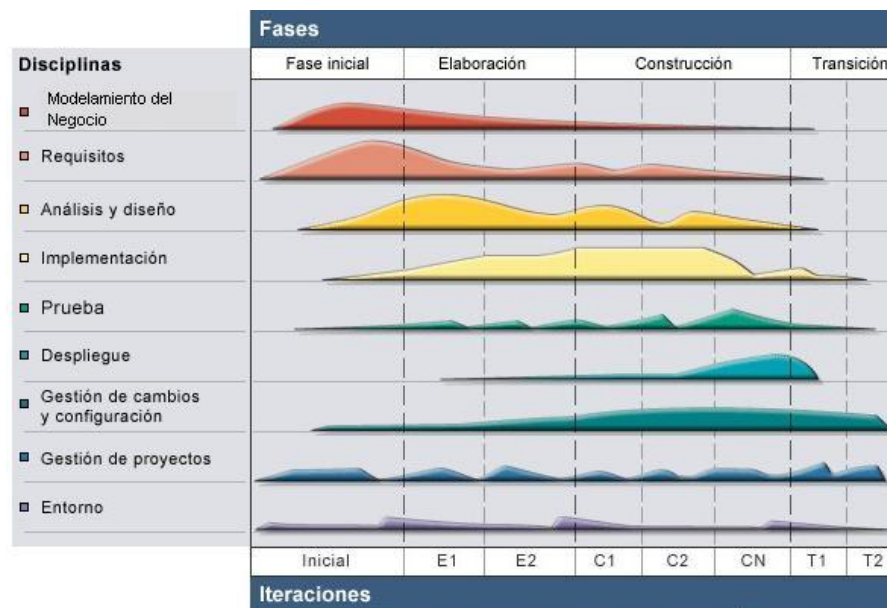


Figura 1 Figura de RUP en dos dimensiones. Se representa el proceso en el que se grafican los flujos de trabajo y las fases y muestra la dinámica expresada en iteraciones y puntos de control.

El ciclo de vida de RUP se caracteriza por ser:

Dirigido por casos de uso: Los casos de uso reflejan lo que los usuarios futuros necesitan y desean, lo cual se capta cuando se modela el negocio y se representa a través de los requerimientos. A partir de aquí los casos de uso guían el proceso de desarrollo ya que los modelos que se obtienen, como resultado de los diferentes flujos de trabajo, representan la realización de los casos de uso (cómo se llevan a cabo).

Centrado en la arquitectura: La arquitectura muestra la visión común del sistema completo en la que el equipo de proyecto y los usuarios deben estar de acuerdo, por lo que describe los elementos del modelo que son más importantes para su construcción, los cimientos del sistema que son necesarios como base para comprenderlo, desarrollarlo y producirlo económicamente. RUP se desarrolla mediante iteraciones, comenzando por los CU relevantes desde el punto de vista de la arquitectura.

Iterativo e Incremental: RUP propone que cada fase se desarrolle en iteraciones. Una iteración involucra actividades de todos los flujos de trabajo, aunque desarrolla fundamentalmente algunos más que otros. Por ejemplo, una iteración de elaboración centra su atención en el análisis y diseño, aunque refina los requerimientos y obtiene un producto con un determinado nivel, pero que irá creciendo incrementalmente en cada iteración.

Aunque cada iteración tiene que proponerse un incremento en el proceso de desarrollo, todas deben aportar al principal resultado de la fase en la que se desarrolla, por lo que los puntos de control evalúan:

Fase	Hito
Conceptualización	Objetivos
Elaboración	Arquitectura
Construcción	Funcionalidad operativa
Transición	Release del sistema

Tabla 1 Fases de RUP y los hitos correspondientes a cada una.

1.5.2. Roles involucrados en el desarrollo

En esta sesión se hace referencia a una de las cuatro “p” del proceso de desarrollo de software, las personas. Las personas involucradas en un proyecto de software pueden ser los clientes, usuarios finales, administrativos y desarrolladores. La palabra trabajador se puede emplear para denominar a los puestos a los cuales se puede asignar personas y a cuales esas personas aceptan. Un tipo de trabajador es un papel que un individuo puede desempeñar en el desarrollo de software. El término rol se emplea para hablar de los papeles que cumple determinado trabajador. (16 pág. 16) RUP define determinados roles por flujo de trabajo dependiendo de las objetivos, actividades y artefactos que se construyan en el mismo. A continuación se relacionan los roles que se desempeñarán durante el proceso de desarrollo de software vinculados a los artefactos que construyen.

Analistas:

Analista de Procesos del negocio

Dirige y coordina los requerimientos del negocio delineando y delimitando la organización que está siendo modelada.

Diseñador del negocio

Detalla la especificación de una parte de la organización.

Analista del sistema

Dirige y coordina la adquisición de requisitos esquematizando la funcionalidad del sistema y delimitándolo.

Especificador de Requerimientos

Especifica y detalla los requisitos del sistema.

Desarrolladores

Diseñador

Dirige el diseño de una parte del sistema, dentro de las restricciones de los requisitos, arquitectura y proceso de desarrollo para el proyecto.

Diseñador de interfaz de usuario

Coordina el diseño de la interfaz de usuario. Esto incluye recopilar los requisitos de utilización y los diseños de interfaz de usuario candidata a la creación de prototipos para cumplir estos requisitos.

Diseñador de BD

Dirige el diseño de la estructura de almacenamiento de datos persistentes que se utilizará en el sistema.

Implementador

Desarrolla los componentes de software y efectúa las pruebas de desarrollador para la integración en subsistemas más grandes, de acuerdo con los estándares adoptados de proyecto.

Artefactos que obtendrán

Flujo de trabajo	Roles	Artefactos
Negocio	Analista de Procesos del negocio	Actor del negocio Diagrama caso de uso del negocio.

		Reglas del negocio
	Diseñador del negocio	Trabajadores del negocio Entidades del negocio Descripciones textuales de los casos de uso del negocio Diagrama de Actividades Modelo de objetos Glosario de Términos
Requerimientos	Analista del sistema	Definición de los Requisitos funcionales y Requisitos no funcionales. Actor del sistema Caso de uso del sistema Diagrama de casos de uso del sistema
	Especificador de Requerimientos	Descripciones textuales de los casos de uso
Diseño	Diseñador	Realización de casos de uso del Diseño: diagramas de clases del diseño y diagramas de interacción Subsistemas del Diseño Paquetes del Diseño
	Diseñador de Interfaz Usuario	Mapa de navegación Prototipos de interfaz de usuario
	Diseñador de Base de Datos	Modelo de Datos

	Arquitecto	Modelo de Despliegue
Implementación	Implementador	Implementación de componentes
	Arquitecto	Modelo de Implementación

Tabla 2 Roles y artefactos por cada flujo de trabajo de RUP

Técnicas empleadas por los roles

Los desarrolladores de productos software cuentan con técnicas que los ayudan en el desarrollo de su trabajo y hacen que su desempeño sea más eficiente. La utilización de patrones es una de las técnicas más empleadas. Un patrón es una solución a un problema dado que ocurre una y otra vez. Gracias a los patrones se puede usar esa solución una infinita cantidad de veces sin necesidad de hacer lo mismo otra vez. En la solución se emplean tres tipos de patrones, arquitectónicos, de caso de uso y de diseño.

Patrones de casos de uso

Los patrones de casos de uso ayudan a identificar casos de uso a partir de las características del negocio o de los requerimientos del sistema, de una forma más eficiente. Los siguientes patrones son empleados en la solución:

CRUD (Create, Read, Update, Delete) o Gestión de Información: Modela las operaciones que pueden ser realizadas sobre una parte de la información de un tipo específico, tales como creación, lectura, actualización y eliminación.

Login: Modela las funciones de dar permiso a los usuarios del sistema para que tengan acceso al resto de los servicios. Identifica un caso de uso para gestionar la información de los usuarios.

Múltiples actores: Cuando dos actores juegan el mismo rol sobre el CU se crea un nuevo actor que representa una generalización. (17)

Patrones de Diseño

Un patrón de diseño es una descripción de clases y objetos comunicándose entre sí adaptada para resolver un problema de diseño general en un contexto particular.

Existen 3 tipos de patrones de diseño:

- De Creación: abstraen el proceso de creación de instancias.
 - De Estructura: se ocupan de cómo clases y objetos son utilizados para componer estructuras de mayor tamaño.
 - De Comportamiento: atañen a los algoritmos y a la asignación de responsabilidades entre objetos.
- (18)

Los patrones de diseño tienen como ventaja que proponen una forma de reutilizar la experiencia de los desarrolladores, para ello clasifican y describen formas de solucionar problemas que ocurren de forma frecuente en el desarrollo. Además están basados en la recopilación del conocimiento de los expertos en desarrollo de software por lo que es una experiencia real, probada, que funciona y ayuda a no cometer los mismos errores.

Entre los patrones empleados en la solución se encuentran algunos del conjunto de patrones GRASP (la siglas pertenecen a las palabras del inglés, General Responsibility Assignment Software Patterns) que comunican los principios fundamentales de la asignación de responsabilidades en el diseño orientado a objetos. (19 pág. 185)

Los patrones GRASP empleados en la solución son:

Experto: Experto es un patrón que se usa más que cualquier otro al asignar responsabilidades; expresa simplemente la "intuición" de que los objetos hacen cosas relacionadas con la información que poseen. La solución que propone es precisamente esta, asignar una responsabilidad al experto en información: la clase que cuenta con la información necesaria para cumplir la responsabilidad. Gracias a este patrón se conserva el encapsulamiento, ya que los objetos se valen de su propia información para hacer lo que se les pide.

Creador: El patrón Creador guía la asignación de responsabilidades relacionadas con la creación de objetos, tarea muy frecuente en los sistemas orientados a objetos. El propósito fundamental de este patrón es encontrar un creador que debemos conectar con el objeto producido en cualquier evento. Al escogerlo como creador, se da soporte al bajo acoplamiento.

Bajo Acoplamiento: La solución que propone este patrón es asignar una responsabilidad para mantener bajo acoplamiento. El bajo acoplamiento es un principio que se debe recordar durante las

decisiones de diseño: es la meta principal que es preciso tener presente siempre. Es un patrón evaluativo que el diseñador aplica al juzgar sus decisiones de diseño. Este patrón soporta el diseño de clases más independientes, que reducen el impacto de los cambios, y también más reutilizables, que acrecientan la oportunidad de una mayor productividad.

Alta Cohesión: La utilización de este patrón mejora la claridad y la facilidad con que se entiende el diseño. Además se simplifican el mantenimiento y las mejoras en funcionalidad. Esto se debe a la solución que propone: Asignar una responsabilidad de modo que la cohesión siga siendo alta.

Controlador: La mayor parte de los sistemas reciben eventos de entrada externa, los cuales generalmente incluyen una interfaz gráfica para el usuario operado por una persona. En este caso, si se está diseñando orientado a objetos, hay que elegir los controladores que manejen esos eventos de entrada. Este patrón ofrece una guía para tomar decisiones apropiadas que generalmente se aceptan. Esto proporciona un potencial de los elementos reutilizables. (19)

Otros patrones empleados en la solución pertenecen a la colección de patrones GOF, estos patrones son:

Factoría: Define una interfaz para crear un objeto, pero deja que sean las subclases quienes decidan qué clase instanciar. Permite que una clase delegue en sus subclases la creación de objetos.

Decorador: Añade dinámicamente nuevas responsabilidades a un objeto, proporcionando una alternativa flexible a la herencia para extender la funcionalidad. (18)

1.5.3. Tecnologías y herramientas

Como se había mencionado anteriormente para guiar el esfuerzo de los desarrolladores del proyecto se seguirá la metodología de desarrollo RUP. También se hizo alusión a que una de las principales características de esta metodología es que su ciclo de vida es **centrado en la arquitectura**. La arquitectura de software selecciona, combina e integra las tecnologías informáticas existentes para construir un software teniendo en cuenta los requisitos funcionales y no funcionales del cliente.

Las herramientas definidas por la arquitectura para la solución se describen a continuación.

UML 2.0

El Lenguaje Unificado de Modelado (UML) es un lenguaje para visualizar, especificar, construir y documentar los artefactos de un sistema. Es la especificación de OMG (Grupo de gestión de objetos, en inglés Object Management Group) y no solo es la forma mundial de representar la estructura de las aplicaciones, comportamiento y arquitectura sino que también representa procesos de negocio y estructura de datos. (20) Además permite el modelado de sistemas con tecnología orientada a objetos. Se debe mencionar que UML no es un proceso.

Este lenguaje tiene elementos que conforman su vocabulario. En los diagramas se relacionan estos conjuntos de elementos y son los encargados de visualizar un sistema desde diferentes perspectivas.

Los principales diagramas de UML son:

Diagramas de estructura estática:

- Diagrama de clases: Conjunto de clases, interfaces y colaboraciones; así como sus colaboraciones.
- Diagrama de objetos: Conjunto de objetos y sus relaciones.
- Diagrama de casos de uso: Conjunto de casos de uso, actores y sus relaciones.

Diagramas de comportamiento:

- Diagramas de interacción (secuencia y colaboración): Objetos y sus relaciones, incluyendo los mensajes que pueden ser enviados entre ellos.
- Diagrama de estados: Muestra una máquina de estado que consta de estados, transiciones, eventos y actividades.
- Diagrama de actividad: Es un tipo especial de diagrama de estados que muestra el flujo de actividades dentro de un sistema.

Diagramas de implementación:

- Diagrama de componentes: Organización y las dependencias entre un conjunto de componentes.
- Diagrama de despliegue: Configuración de nodos de procesamiento en tiempo de ejecución y los componentes que residen en ellos.

Ventajas

- Concurrencia, es un lenguaje distribuido y adecuado a las necesidades de conectividad actuales y futuras.
- Ampliamente utilizado por la industria desde su adopción por OMG.
- Reemplaza a decenas de notaciones empleadas con otros lenguajes.
- Modela estructuras complejas.
- Las estructuras más importantes que soportan tienen su fundamento en las tecnologías orientadas a objetos, tales como objetos, clase, componentes y nodos.
- Emplea operaciones abstractas como guía para variaciones futuras, añadiendo variables si es necesario. (19)

Herramientas CASE

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

Visual Paradigm for UML 6.4

Es una herramienta CASE, desarrollada por la compañía Visual Paradigm International, que utiliza UML como lenguaje de modelado.

Presenta las siguientes características:

- Soporta el ciclo de vida completo del software: análisis, diseño, implementación y despliegue.
- Permite la captura de requisitos, el dibujo de diagramas UML, la realización de ingeniería inversa y generación de código PHP.
- Se integra con las siguientes herramientas:

- Eclipse/IBM WebSphere
- Builder
- Net Beans IDE
- Oracle JDeveloper
- BEA Weblogic

- Está disponible en varias ediciones, cada una destinada a necesidades específicas: Enterprise, Professional, Community, Standard, Modeler y Personal.

La versión empleada en la solución es el Visual Paradigm for UML 6.4, lanzada el 20 de Octubre de 2008. Esta versión incluye siete nuevas mejoras:

1. Importa diagramas de Microsoft Office Visio a Visual Paradigm.
2. Soporta patrones de diseño.
3. Soporta las tres formas del Diagrama Entidad Relación, conceptual, lógica y física.
4. Mejora la trazabilidad de elementos utilizando el historial de revisiones.
5. Soporta líneas anidadas.
6. Muestra como elemento estereotipado del modelo un ícono de imagen.
7. Muestra y oculta elementos del diagrama según se desee. (21)

DB Designer Fork 4

DB Designer Fork 4 es un sistema de diseño visual de base de datos que integra el diseño, el modelamiento, la creación y el mantenimiento de bases de datos en un simple ambiente de ventanas. Genera las escrituras para diferentes gestores de base de datos como SQL Server, MySQL, Oracle y PostgreSQL (PGSQL). Posee un sistema de generación automática de clave única y una integración directa al servidor de base de datos llamado sincronización que transmite directamente el script hacia el servidor de BD.

Esta herramienta emplea licencia GPL y además es multiplataforma. (22)

Entorno de desarrollo

Zend Studio for Eclipse 6.0

Zend Studio es un entorno integrado de desarrollo para la programación en PHP. Forma parte, junto con Zend Platform de lo que la empresa Zend Technologies propone para el desarrollo Web, Zend Studio para la parte cliente y Zend Platform para la parte del servidor. Esta herramienta no requiere instalación previa de PHP, ni de ningún otro entorno de ejecución Java. Ha establecido el autocompletamiento de código, el plegado de código, inserción automática de paréntesis y corchetes, detección de errores de sintaxis en tiempo real y funciones de depuración. Posee también un manual de PHP integrado y un cliente FTP integrado.

El programa entero está escrito en el lenguaje de programación Java, lo que a veces supone que no funcione tan rápido como otras aplicaciones de uso diario. Sin embargo, esto ha permitido a Zend lanzar con relativa facilidad y rapidez versiones del producto para sistemas operativos como Windows, Linux y Macintosh. (23)

Tecnologías del lado del cliente

JavaScript

JavaScript se implementó por primera vez en la versión beta de Netscape Navigator 2.0 en junio de 1995, por la empresa Netscape Communications Corporation. Es un lenguaje de programación interpretado utilizado en la realización de páginas Web. Una de sus principales características es que es orientado a objetos.

La verdadera fuerza de JavaScript es la compatibilidad con los distintos navegadores, incluso con los más anticuados. Además de que es interpretado por todos los navegadores modernos. (24)

Frameworks

Un framework es una estructura de soporte definida en la cual otro proyecto de software puede ser organizado y desarrollado. Proporciona una estructura que fuerza al desarrollo de código más legible. Representa una arquitectura de software que modela las relaciones generales de las entidades del dominio y permite separar en capas la aplicación.

Symfony 1.2

Symfony 1.2 es un framework de PHP que simplifica el trabajo con páginas Web dado a la automatización de algunos patrones para resolver tareas frecuentes. Este framework tiene una estructura de código que fuerza al desarrollo de código más legible. También encapsula operaciones complejas en instrucciones sencillas. Este framework trabaja con el patrón de arquitectura Modelo Vista Controlador y la herramienta Propel para el mapeo de objetos a bases de datos. Es un framework sencillo para cualquier programador independientemente de que sea principiante o un experto en PHP. Symfony está programado en PHP5 y enfocado al desarrollo en el mismo lenguaje. Este framework hace un uso completo de los mecanismos de POO (Programación Orientada a Objetos) disponibles en PHP5. Permite descargar, instalar, actualizar y desinstalar script PHP. Symfony es un completo framework diseñado para optimizar, gracias a sus características, el desarrollo de las aplicaciones Web. (25)

ExtJS 2.0

ExtJS 2.0 es un framework de presentación completamente OO (Orientado a Objetos) de JavaScript, es multiplataforma y hace uso de la tecnología AJAX. Este framework brinda múltiples posibilidades para el trabajo con las validaciones y manejo de errores en el cliente. Además permite la personalización de temas de estilos y provee el trabajo con una amplia configuración e intenso trabajo con las hojas de estilo CSS. Basa toda su funcionalidad en JavaScript a través de librerías YUI, jQuery, o haciendo uso de la librería nativa, así en tiempo de ejecución carga y crea todos los objetos HTML a través del uso intenso de DOM. Cuenta con dos licencias, una comercial y otra Open Source. (26)

Plataforma

LAMP

LAMP es la plataforma que integra Linux, Apache, MySQL y PHP. Es ampliamente utilizada en el mundo del software libre para soluciones de aplicaciones Web, pues permite la creación de un servidor sin la necesidad de conectar sus 4 elementos por separado. También provee una facilidad de administración con el phpMyAdmin que contiene integrado. Posee las siguientes ventajas:

- Es rápido y confiable.
- Distribuido bajo licencia GPL.

- Se puede modificar todo su código fuente.
- Estos 4 componentes pueden trabajar con una gran cantidad de hardware con requerimientos mínimos.

Sistema Operativo

Ubuntu 8.10 (Intrepid Ibex)

Ubuntu 8.10 es una distribución GNU/Linux que ofrece un sistema operativo predominantemente enfocado a computadores personales, aunque también proporciona soporte para servidores. Es una de las más importantes distribuciones de GNU/Linux a nivel mundial. Se basa en la distribución Debian GNU/Linux y concentra su objetivo en la facilidad y libertad de uso, la fluida instalación y los lanzamientos regulares.

Este sistema operativo cuenta con:

- Tema de escritorio alternativo rediseñado.
- Habilidad para aprovechar el ancho de banda donde sea que se encuentre el usuario (Internet omnipresente).
- Mejoras en el soporte para todo tipo de máquina de escritorio: Desde una máquina de trabajo actual (doble núcleo, chipsets avanzados, gigas en memoria y disco duro) hasta escritorios con Pentium III.
- Cifrado de carpetas. (27)

Servidor Web

Apache 2.0

El proyecto Apache comenzó en 1995 como un proyecto para la creación de un servidor Web de código abierto. Con los años se le añadieron muchas funcionalidades como la habilidad de un solo servidor de mantener múltiples sitios Web virtuales y esquemas de autenticación. Actualmente es uno de los servidores más usados para la creación de sitios Web comerciales. Está disponible para una gran cantidad de plataformas como GNU/Linux, Mac OS, Mac OS X Server, Netware, Open Step/Match, UNIX, Solaris, SunOS, UnixWare, Windows entre otras.

Sistema Gestor de Bases de Datos

MySQL 5.0

MySQL es uno de los Sistemas Gestores de Bases de Datos (SQL) más populares desarrolladas bajo la filosofía de código abierto. Fue desarrollado por la empresa MySQLAB (fundada en 1995). Este producto se distribuye bajo licencia GPL de MySQL que obliga a cualquier producto derivado a ser distribuido bajo esa misma licencia. Entre sus características más relevantes se encuentra: amplia disponibilidad en gran cantidad de plataformas y sistemas, transacciones y llaves foráneas, replicación, conectividad, búsqueda e indexación de campos de texto. Además se debe agregar que es un sistema multiplataforma y que realiza la gestión de usuarios y contraseñas manteniendo un alto nivel de seguridad.

Lenguaje de Programación

PHP5

PHP es un acrónimo recursivo que significa **PHP Hypertext Pre-processores**. Es un lenguaje interpretado de alto nivel embebido en páginas HTML y ejecutado en lado del servidor. Puede ser desplegado en la mayoría de los servidores Web y en casi todos los sistemas operativos y plataformas sin costo alguno. Este lenguaje posee amplias ventajas como:

- Es un lenguaje multiplataforma.
- Capacidad de conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL.
- Capacidad de expandir su potencial utilizando la enorme cantidad de módulos (llamados extensiones).
- Posee una amplia documentación en su página oficial (www.php.net), entre la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda.
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Permite las técnicas de Programación Orientada a Objetos.
- Contiene una biblioteca nativa de funciones sumamente amplia e incluida.

- No requiere definición de tipos de variables.
- Tiene manejo de excepciones (desde PHP5). (25)

Gestión de Proyecto-Gestión Documental-Control y seguimiento de errores

DotProject

DotProject es una herramienta para la gestión de proyectos basada en tecnología Web. Incluye módulos para compañías, tareas (con gráficas Gantt), foros, permite subir archivos, calendarios y soporte multilinguaje, que lo hacen una herramienta muy completa. DotProject fue creado por dotmarketing.org en el año 2000, con el fin de construir una herramienta para la Gestión de Proyectos. Está construido por aplicaciones de código abierto y es mantenida por un dedicado grupo de voluntarios.

Está programado en PHP, y utiliza inicialmente MySQL como base de datos (aunque otros motores como PostgreSQL también pueden ser utilizados). Es muy recomendado para usar junto a la plataforma LAMP. El grupo que desarrolla DotProject se basa en los siguientes puntos:

- Proveer a los usuarios de funcionalidad orientada a la Gestión de Proyectos.
- Construir una herramienta con una interfaz de usuario simple, claro y consistente.
- Ser de código abierto, libre acceso y utilización.

Sistema de control de versiones

Subversion

Subversion es un sistema de control de versiones de código abierto (Open Source). Mediante Subversion, se puede registrar la historia de archivos de código fuente y documentos. Gestiona archivos y directorios a través del tiempo. Para realizar esta gestión se coloca un árbol de archivos en un repositorio central. El repositorio es como un servidor de archivos ordinario, excepto en que recuerda todos los cambios que se realizan sobre los archivos y los directorios que contiene.

Sistema de Salvas Automáticas

Bacula 2.4.2

Bacula ofrece importantes características avanzadas de almacenamiento de datos que hacen posible encontrar y recuperar archivos dañados. Está conformado por varios servicios importantes: director de

la consola, archivo, almacenamiento, y servicios de monitoreo. Es un conjunto de paquetes informáticos que permiten administrar los respaldos, recuperaciones y verificaciones de los datos de una máquina a través de una red de diferentes computadoras. Puede correr sobre una sola computadora y puede respaldar hacia varios tipos de medio de almacenamiento, incluyendo casetes y discos duros.

1.6. Conclusiones

- ✓ En medio del desarrollo de la informática en el país y el surgimiento o fortalecimiento de factores que favorecen el establecimiento de sistemas de gestión de información, se hace necesario la utilización de una herramienta que automatice aquellos procesos claves dentro del Departamento de Supervisión del CCEEM.
- ✓ El estudio de herramientas de gestión de la información existentes dejó ver que ninguna cumple con las características requeridas, por lo que se hace necesaria la creación de un nuevo sistema para la gestión de la información del Departamento de Supervisión.
- ✓ Del estudio de las herramientas que se emplean en la solución se concluye que todas cuentan con un gran potencial que tributa a que el producto final sea estable. Estas herramientas apoyan el desarrollo general de la aplicación, brindando facilidades a los desarrolladores en cuanto a rapidez, facilidad de uso y variedad de funciones, sin dejar de mencionar que son herramientas libres.

CAPÍTULO 2 CARACTERÍSTICAS DEL SISTEMA

En este capítulo se hace un estudio del negocio y se mencionan las características del sistema. Para ilustrar esto se presentan los artefactos obtenidos en el flujo de trabajo de RUP Modelamiento del negocio, los objetivos de este flujo son estudiar y entender la estructura y la dinámica de la organización y encontrar posibles automatizaciones. Se desarrolla además el flujo de trabajo de Requerimientos, en el cual se obtienen los requerimientos del sistema, los casos de uso y sus descripciones para un mejor entendimiento de los implicados de lo que el sistema debe hacer.

2.1. Objeto de estudio

Objetivos estratégicos de la organización

Los especialistas del Departamento de Supervisión del CCEEM son los encargados de garantizar el cumplimiento de los requisitos regulatorios establecidos en la Regulación sobre Requisitos de Calidad para los fabricantes de Equipos Médicos, para esto realizan las auditorías de calidad al fabricante y la verificación de los certificados de calidad entregados por fabricantes extranjeros.

Flujo actual de los procesos

Al iniciar el proceso de inscripción/reinscripción de fabricante el Departamento de Documentación entrega los documentos presentados por el fabricante al Departamento de Supervisión para su revisión. Los especialistas de este departamento realizan la revisión de la documentación de acuerdo a lo regido por la Regulación ER-11a “Regulación sobre Requisitos para Sistemas de Gestión de la Calidad para la Fabricación de Equipos Médicos”. A partir de las revisiones se emiten las conclusiones de la revisión. Estas son entregadas de forma oficial al Departamento de Documentación a través del documento “Lista de Chequeo de calidad para la inscripción del fabricante” que plasma los detalles de la revisión. Los documentos empleados en el proceso son archivados para su posterior consulta.

Las conclusiones de la revisión pudieran ser:

- Supervisar al fabricante. Esta función se lleva a cabo a través de la realización de una Auditoría de Calidad al fabricante, cuando presenta una solicitud de Inscripción o Certificación de Buenas Prácticas de Fabricación de un equipo médico, ante una queja recibida y también durante el período en que los equipos médicos se encuentran registrados, de acuerdo al Programa de auditorías del cuatrimestre en curso.

- Documentación del Sistema de Calidad con alguna dificultad que no invalida el proceso que solicita el fabricante. Se propone realizar una Inspección al Sistema de Calidad del fabricante.
- Documentación del Sistema de Calidad incompleta o no satisfactoria que detiene la continuidad del proceso que solicita el fabricante. Puede ser necesario la entrega de otros documentos o la realización de una auditoría al Sistema de Calidad del fabricante para obtener más evidencia.
- Documentación del Sistema de Calidad incompleta o no satisfactoria que invalida el proceso que solicita el fabricante. Se propone denegar el proceso.

Análisis crítico de la ejecución de los procesos

Realizar el documento “Lista de chequeo de calidad para inscripción de fabricante”.

- Escribir cada una de las conclusiones a las que se arribaron en el proceso de revisión de la documentación del fabricante.
- Este proceso es un poco engorroso a la hora de escribir todas las conclusiones siguiendo el formato establecido.

Notificar al fabricante la programación de una auditoría.

- Establecer una fecha para la auditoría y las características de la misma. Notificar al fabricante con antelación para que envíe su documentación y exprese además su acuerdo con la planificación.
- Este proceso tiene como principal problema la constante revisión de los registros para verificar si se acerca la fecha de alguna auditoría. Estos registros son:

- Registro Programa de Auditorías.
- Registro de auditorías y Planes de Acciones Correctivas.
- Registro Planes de Calidad del fabricante.

Conformar documentos a partir de planillas.

- Crear el documento a partir de la planilla establecida que corresponden a cada una de los procesos siguiendo además el formato establecido.

- Este proceso resulta engorroso a la hora de llenar manualmente los campos de las planillas siguiendo el formato establecido en las mismas.

Consultar documentos y brindar información.

- Buscar en el archivo del Departamento los documentos que se deseen consultar. Brindar información, dígame datos estadísticos obtenidos a partir de los registros del centro.
- Este proceso se dificulta por el gran volumen de información que se encuentra archivada, la búsqueda de un documento en específico provoca pérdida de tiempo y eficiencia. El cálculo de estadísticas es más engorroso aún al tener que contar y calcular manualmente.

Procesos objeto de automatización

El sistema informatizará la gestión de las planillas que se usan actualmente, permitiendo operaciones de inserción, modificación, impresión y en el caso que se requiera se podrá conformar planillas automáticamente. Brindará servicios para la gestión de registros incluyendo la gestión de avisos cuando sea cercana la fecha de realizar una acción contenida en los registros.

Permitirá además la visualización de la documentación incluyendo los expedientes de los fabricantes inscritos en el CCEEM y el cálculo de estadísticas a partir de los datos registrados en el sistema así como su visualización gráfica.

2.2. Modelo del negocio

El proceso de modelamiento del negocio permite obtener una visión de la organización con el objetivo de definir los procesos, roles y responsabilidades de la organización en los modelos de casos de uso del negocio y de objetos.

2.2.1. Actores del negocio

Un actor del negocio representa una persona, grupo de personas, organización o sistema externo al negocio que interactúe con él.

Justificación de los actores del negocio

Actor	Descripción
Fabricante	Persona responsable por el diseño, fabricación, empaquetamiento y etiquetado de un equipo médico.
Auditor Externo	Autoridad que se acerca al departamento para realizar una revisión de los procedimientos y documentación del mismo.
Demandante	Institución o unidad de salud que presenta una queja sobre los productos y procedimientos de algún fabricante.

Tabla 3 Justificación de los actores del negocio

2.2.2. Trabajadores del negocio

Un trabajador del negocio representa una persona, sistema o equipo de trabajo dentro del negocio que colabora para alcanzar un resultado esperado por el actor.

Justificación de los trabajadores del negocio

Trabajador	Descripción
Especialista DS	Trabajador del Departamento de Supervisión que atiende los diferentes procesos realizados en el negocio y conforman el equipo auditor.
Auditor Líder	Trabajador del Departamento de Supervisión encargado de dirigir al equipo auditor.
Jefe DS	Trabajador del Departamento de Supervisión encargado de dirigir todas las actividades.
Departamento de Documentación	Departamento del CCEEM encargado de la Inscripción/Reinscripción de fabricantes.

Tabla 4 Justificación de los trabajadores del negocio

2.2.3. Diagrama de casos de uso del negocio

El diagrama de casos de uso representa la interacción entre actores del negocio y los casos de uso del negocio, incluye las relaciones entre actores y entre casos de uso.

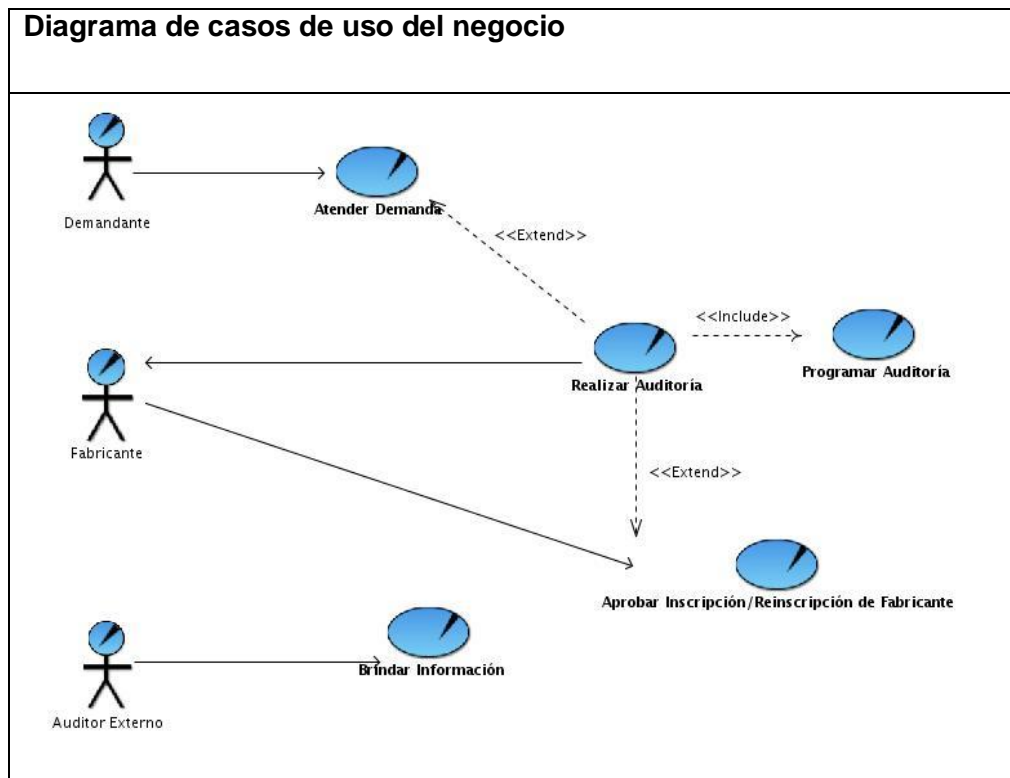


Figura 2 Diagrama de casos de uso del negocio

2.2.4. Realización de los casos de uso del negocio

Los casos de uso del negocio consisten en secuencias de actividades que, en conjunto, producen algo para el actor del negocio. La realización de los casos de uso ayuda a entender en detalle como se suceden las actividades a partir de la descripción de los flujos de procesos. Brinda una noción de cómo funcionan estos procesos dentro del negocio.

Descripción textual de los casos de uso del negocio

Una descripción textual presenta en detalles las características principales del caso de uso y describe el flujo de procesos que lo conforman empleando un vocabulario sencillo y conciso. La descripción divide el flujo en acciones del actor y la respuesta del negocio. Se incluyen los flujos alternos,

dedicados a las acciones que se desarrollan en casos excepcionales. A continuación se muestran las descripciones resumidas de los casos de uso.

1. Descripción textual resumida del caso de uso del negocio Programar Auditoría

Nombre del CU	Programar Auditoría.
Actores	Fabricante
Trabajadores	Jefe DS(Inicia)
Resumen	El CU se inicia cuando el Jefe DS debe programar una auditoría. Establece el tipo de auditoría y la fecha en que se va a realizar y se la comunica al fabricante. Este envía confirmación y el Jefe DS registra la auditoría en el Registro Programa de Auditorías, finalizando así el CU.
Posibles mejoras	Se recomienda automatizar la manipulación de datos para agilizar el proceso de actualización de documentos.

Tabla 5 Descripción textual resumida del caso de uso del negocio Programar Auditoría.

2. Descripción textual resumida del caso de uso del negocio Realizar Auditoría

Nombre del CU	Realizar Auditoría.
Actores	Fabricante (Inicia)
Trabajadores	Auditor Líder, Especialista DS, Jefe DS
Casos de uso asociados	Programar Auditoría(inclusión)
Resumen	El caso de uso se inicia cuando se programa una auditoría, se le informa al fabricante y se procede a realizar la misma. Se actualizan los documentos pertinentes finalizando así el caso de uso.
Posibles mejoras	Se propone automatizar el proceso de acceso a la información requerida para agilizar todas las actividades relacionadas así como la actualización de la misma.

Tabla 6 Descripción textual resumida del caso de uso del negocio Realizar Auditoría.

3. Descripción textual resumida del caso de uso del negocio Atender Demanda

Nombre del CU	Atender Demanda.
Actores	Demandante(Inicia)
Trabajadores	Especialista DS
Casos de uso asociados	Realizar Auditoría(extensión)
Resumen	El CU se inicia cuando el Demandante se presenta en el DS y presenta una demanda contra un producto de uno de los fabricantes inscritos. El DS valida la demanda y decide el procedimiento a seguir. Envía copia al demandante de los resultados.
Posibles mejoras	Se propone automatizar el proceso de acceso a la información requerida para agilizar todas las actividades relacionadas así como la actualización de la misma.

Tabla 7 Descripción textual resumida del caso de uso del negocio Atender Demanda.

4. Descripción textual resumida del caso de uso del negocio Aprobar Inscripción/Reinscripción del Fabricante

Nombre del CU	Aprobar Inscripción/Reinscripción del Fabricante.
Actores	Fabricante(inicia)
Trabajadores	Especialista DS
Casos de uso asociados	Realizar Auditoría(extensión)
Resumen	El CU se inicia cuando el fabricante se dirige al DD y pide inscribirse o reinscribirse en el CCEEM. El DD le traspasa la documentación al DS. Este departamento decide si la documentación presentada es suficiente o si se debe realizar una auditoría. Al finalizar emite su aprobación al DD de acuerdo a las normas establecidas a través de un informe completo finalizando así el CU.

Posibles mejoras	Se recomienda automatizar el proceso de acceso a la información requerida para agilizar todas las actividades relacionadas así como la actualización de la misma.
-------------------------	---

Tabla 8 Descripción textual resumida del caso de uso del negocio Aprobar Inscripción/Reinscripción del Fabricante.

5. Descripción textual resumida del caso de uso del negocio Brindar Información

Nombre del CU	Brindar Información
Actores	Auditor Externo(Inicia)
Trabajadores	Jefe DS
Resumen	El CU se inicia cuando el actor se presenta en el Departamento de Supervisión y pide realizar una revisión de los documentos del mismo. El Jefe DS entrega los documentos solicitados. Estos son revisados por el Auditor Externo que al terminar de revisarlos se los devuelve al Jefe DS y en caso de encontrarlos menciona sus señalamientos.
Posibles mejoras	Se recomienda informatizar el proceso de acceso a la información requerida para agilizar todas las actividades relacionadas.

Tabla 9 Descripción textual resumida del caso de uso del negocio Brindar Información.

(Para ver descripciones textuales completas de casos de uso del negocio Anexo 2: Descripciones textuales de casos de uso del negocio.)

Diagramas de actividades

El diagrama de actividades es parte de la realización de un caso de uso, representa un flujo de actividades que describen cómo colaboran trabajadores, actores y entidades del negocio para alcanzar un resultado final. El proceso consiste de un flujo básico de una o más alternativas de flujos. La estructura del flujo se describe gráficamente con la ayuda del diagrama de actividades.

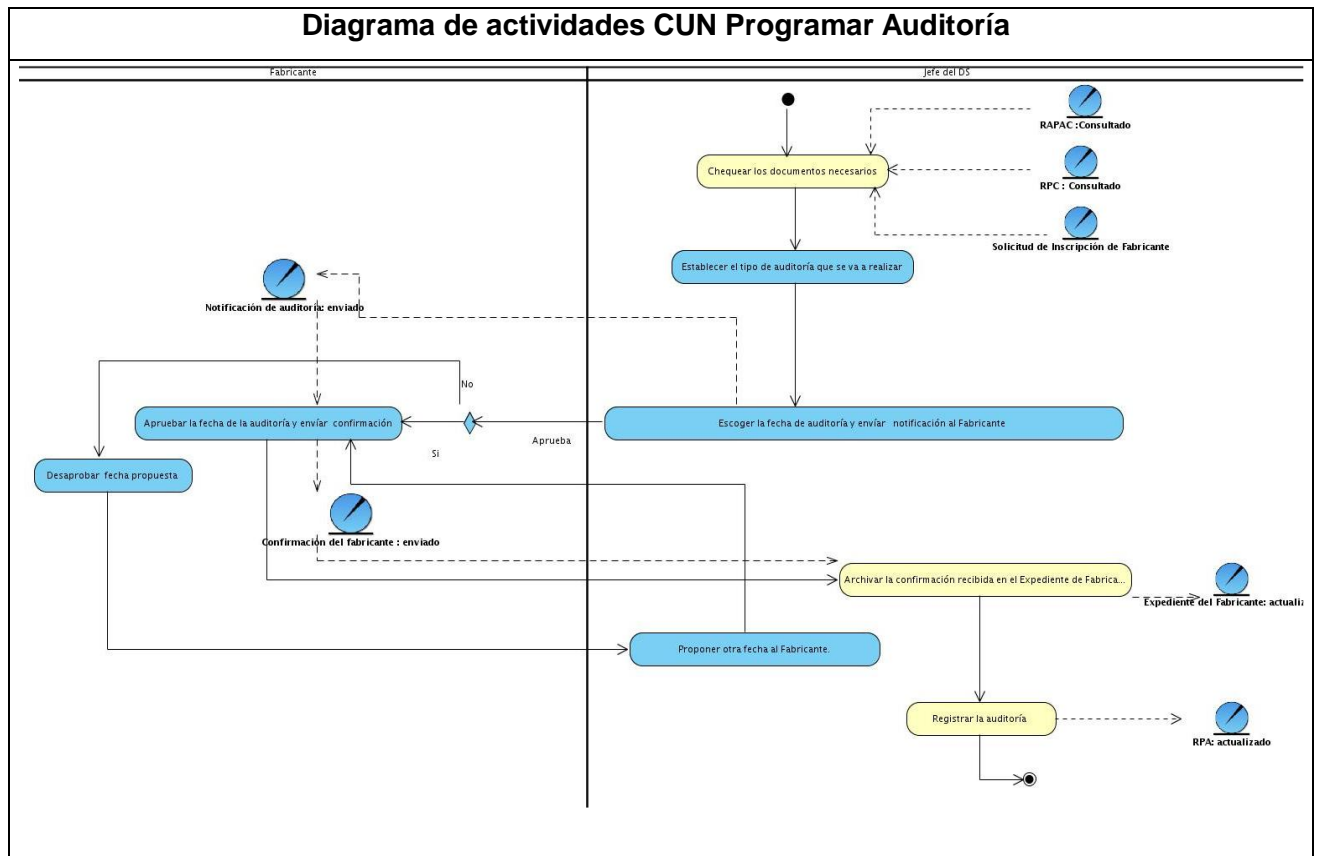


Figura 3 Diagrama de actividades CUN Programar Auditoría.

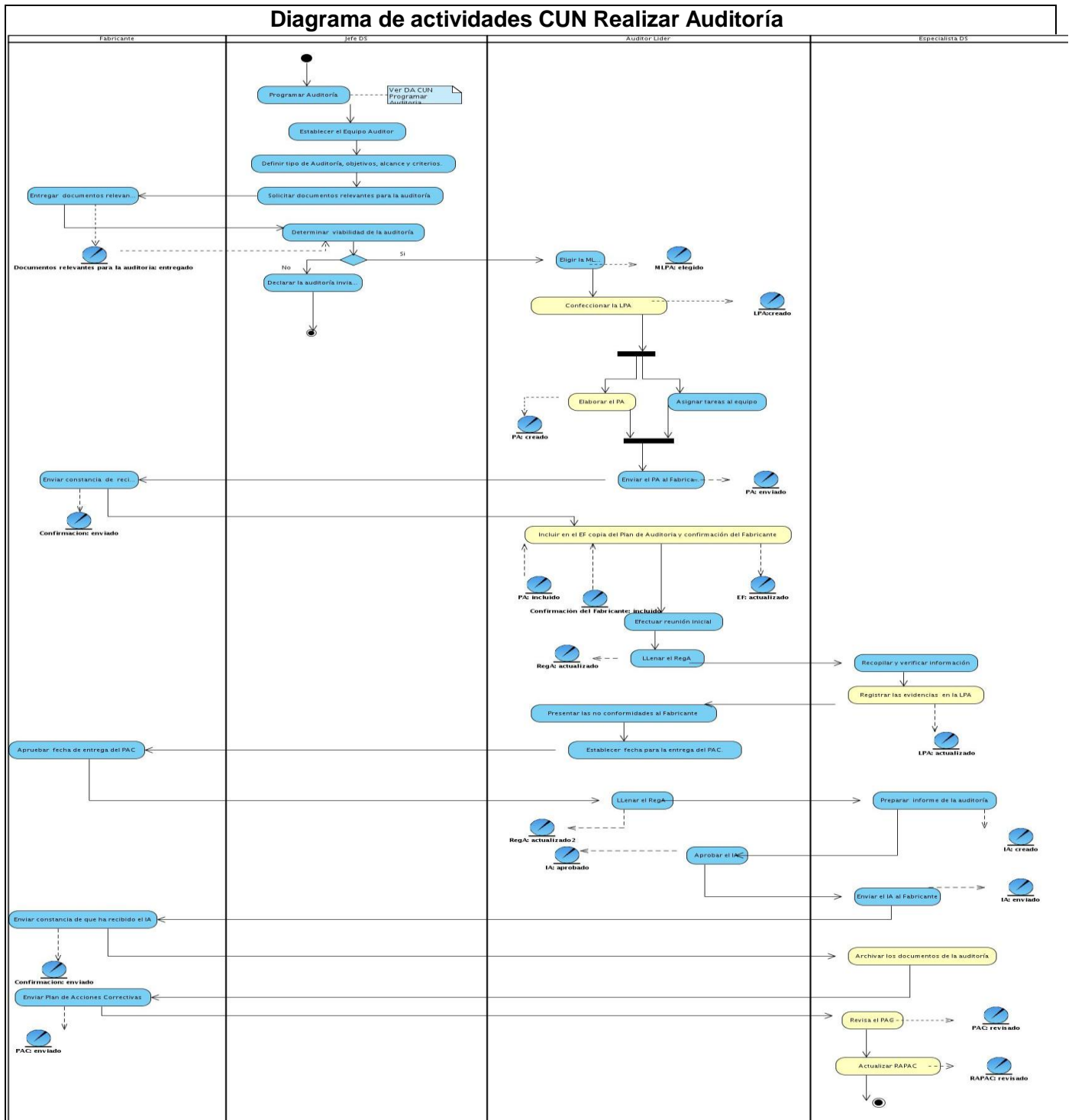


Figura 4 Diagrama de actividades CUN Realizar Auditoría.

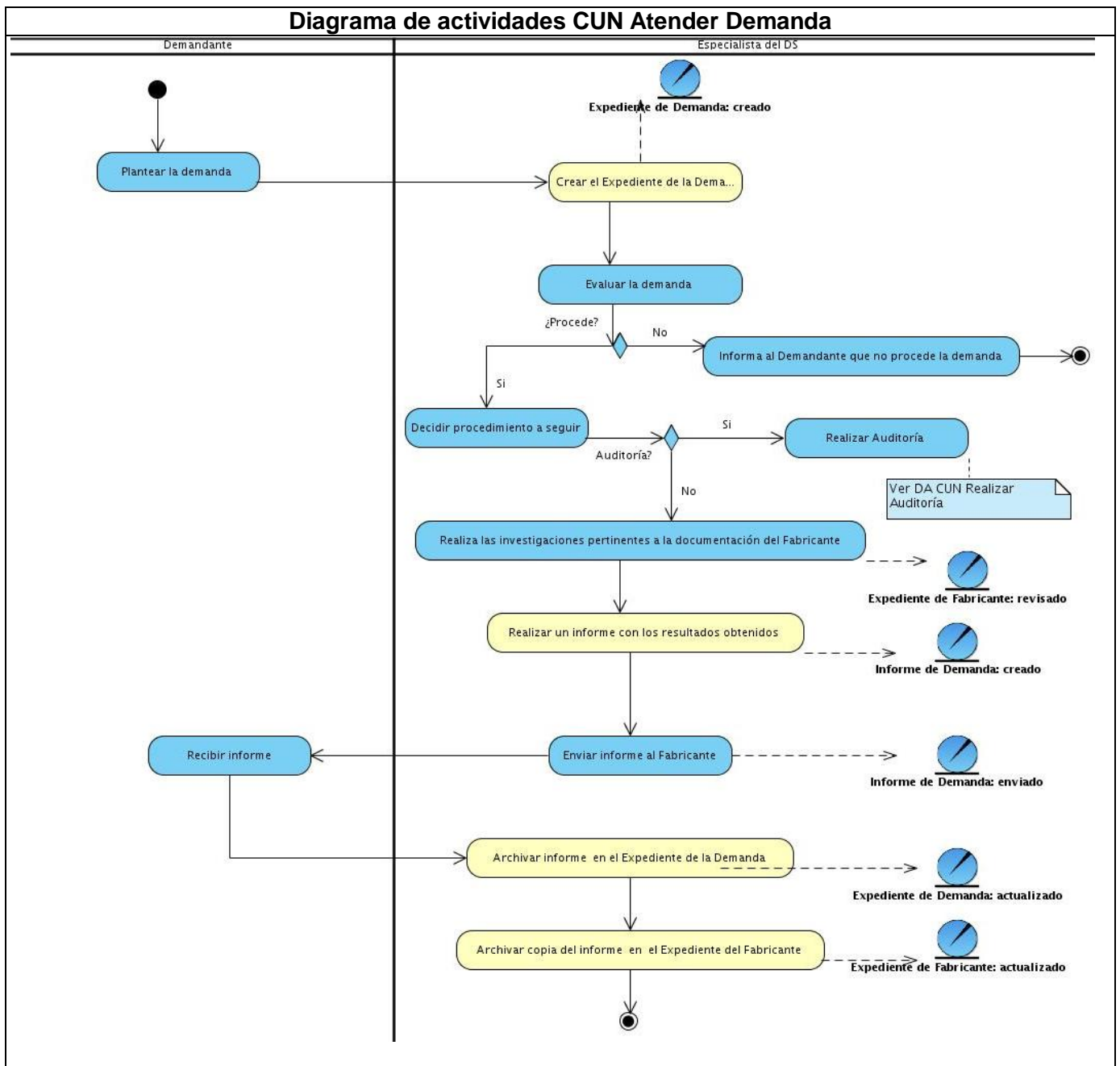


Figura 5 Diagrama de actividades CUN Atender Demanda.

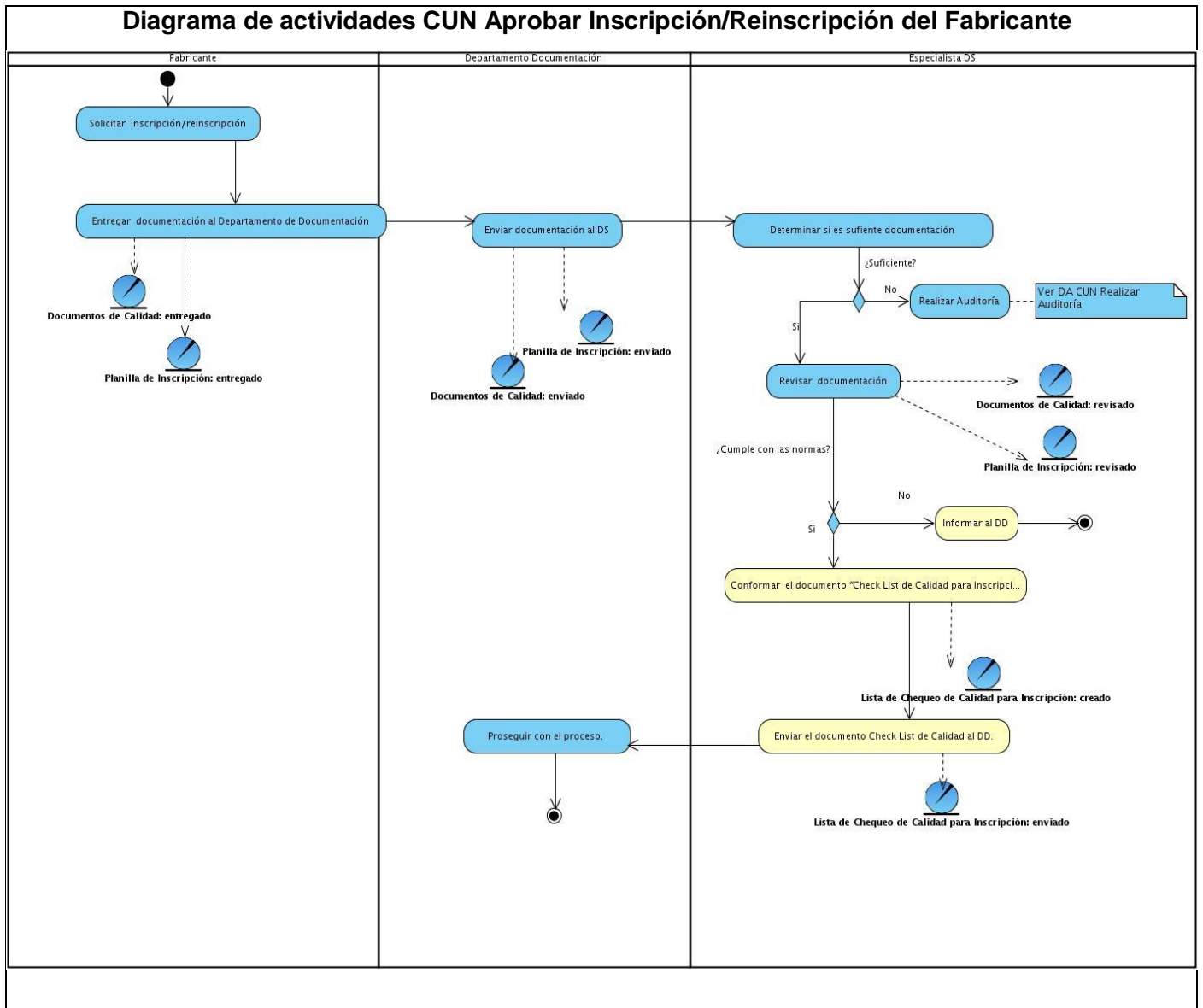


Figura 6 Diagrama de actividades CUN Aprobar Inscripción/Reinscripción del Fabricante.

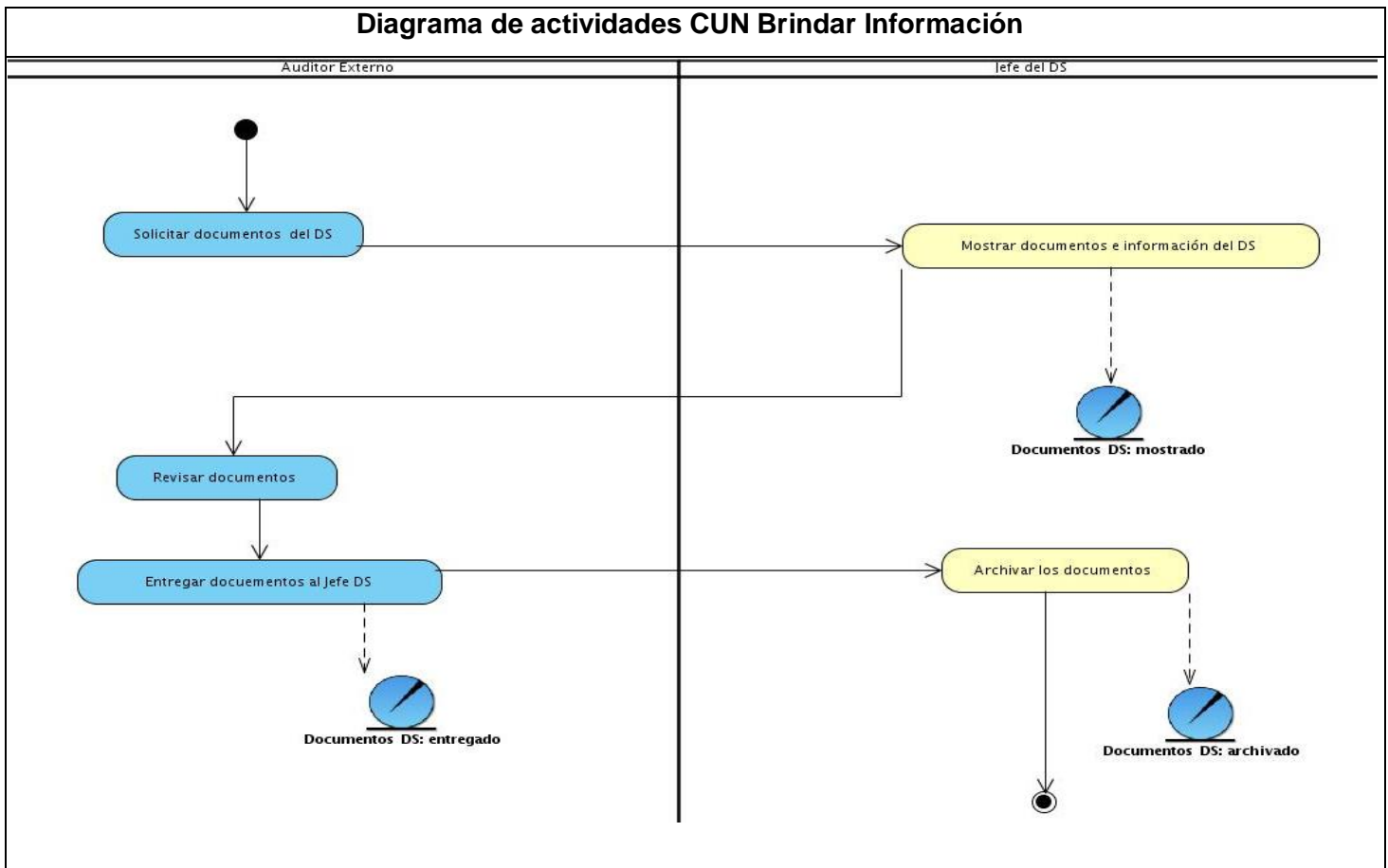


Figura 7 Diagrama de actividades CUN Brindar Información.

2.2.5. Diagrama de clases del Modelo de objetos

El diagrama de clases del modelo de objetos es el conjunto básico de entidades del negocio y los trabajadores del negocio que las manipulan. Estos diagramas para más organización se han realizado por cada uno de los trabajadores del negocio.

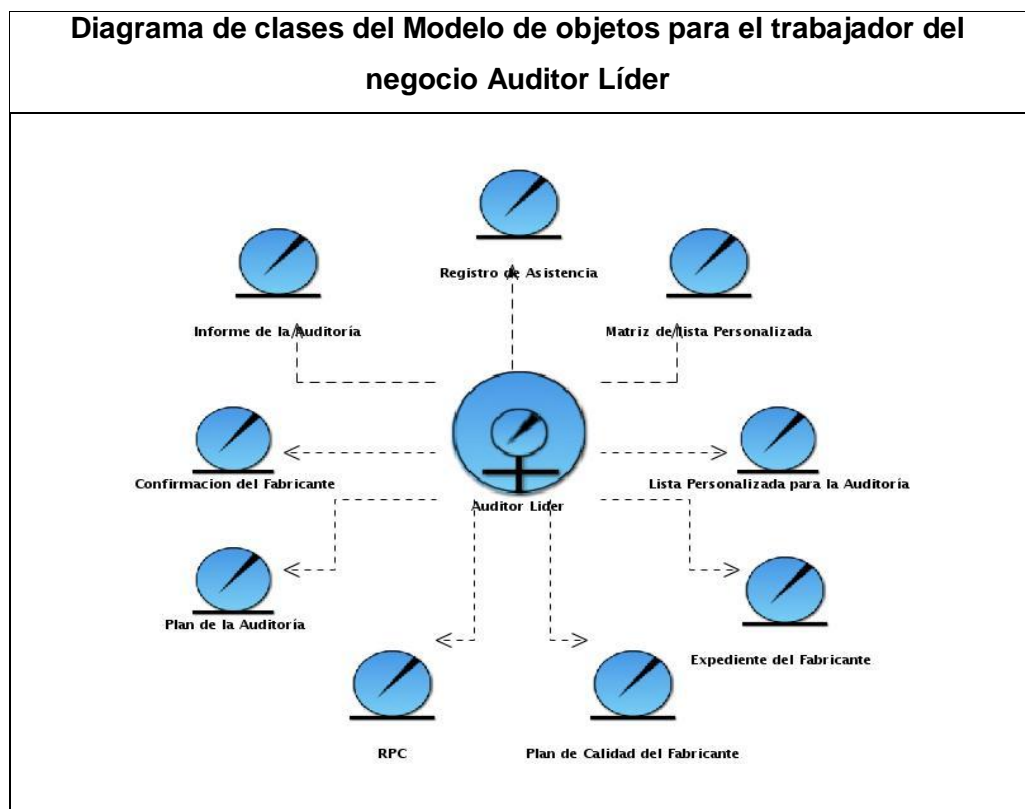


Figura 8 Diagrama de clases del Modelo de objetos para el trabajador del negocio Auditor Líder.

(Para ver el resto de los diagramas de clases del Modelo de objetos ir Anexo 3: Modelo de objetos del negocio.)

2.2.6. Reglas del negocio

Las reglas de negocio describen políticas que deben cumplirse o condiciones que deben satisfacerse, por lo que regulan algún aspecto del negocio.

Reglas de estructura

1. Término:

RN 1 El número de la auditoría se conforma por cuatro dígitos. Los dos primeros están designados para indicar el consecutivo del año en curso y los dos últimos indican el año donde se realiza la auditoría, p.e 0109, la primera auditoría del 2009.

Reglas de acción

1. Restricciones de operaciones:

RN 2 Para dar por terminado el proceso de programar auditoría, el fabricante debe haber enviado una notificación confirmando que conoce la fecha de la misma y la aprueba.

RN 3 Para llevar a cabo el proceso de Inscripción/Reinscripción de fabricante el DD debe contar con la aprobación del Departamento de Supervisión.

2. Estímulo y respuesta:

RN 4 Si se cumple la fecha de entrega de un Plan de Acciones Correctivas y el fabricante no la ha enviado se procede a enviarle una notificación.

2.3. Requerimientos. Modelo del sistema

El flujo de trabajo de RUP Requerimientos, se desarrolla en la fase de Inicio. El objetivo fundamental de este flujo de trabajo es establecer un convenio entre cliente y desarrolladores sobre lo que el sistema debe hacer. A partir de la especificación de los requisitos y la modelación gráfica de estos, agrupados en casos de uso, se brinda una visión de lo que se debe implementar.

El artefacto fundamental este flujo de trabajo es el Modelo del sistema que contiene actores y casos de uso. Este artefacto es un modelo de las funciones deseadas para el sistema y su entorno, y sirve como contrato entre el cliente y los desarrolladores. Se utiliza como entrada esencial para las actividades de análisis, diseño y prueba. La realización de los casos de uso del sistema se realiza en el flujo de trabajo de Análisis y Diseño.

2.3.1. Especificación de requisitos

Un requisito es:

- c) Condición o capacidad que necesita un usuario para resolver un problema o lograr un objetivo.
- d) Condición o capacidad que tiene que ser alcanzada o poseída por un sistema o componente de un sistema para satisfacer un contrato, estándar, u otro documento impuesto formalmente.
- e) Una representación documentada de una condición o capacidad como en las dos anteriores. (28)

Los requisitos se pueden clasificar en: funcionales y no funcionales. Los requisitos funcionales son capacidades o condiciones que el sistema debe cumplir. Los requisitos no funcionales son propiedades o cualidades que el producto debe tener, las características que hacen al producto atractivo, usable, rápido o confiable.

Estrategia de captura de requisitos utilizada

La captura de requisitos es la actividad mediante la cual el equipo de desarrollo de un sistema de software extrae, de cualquier fuente de información disponible, las necesidades que debe cubrir dicho sistema. Por la complejidad que esto implica, el equipo de desarrollo empleó técnicas que permitieron hacer este proceso de una forma más eficiente y precisa.

Técnicas de captura de requisitos

Entrevistas: Resultan una técnica muy aceptada dentro de la ingeniería de requisitos y su uso está ampliamente extendido. Las entrevistas le permiten al analista tomar conocimiento del problema y comprender los objetivos de la solución buscada. A través de esta técnica el equipo de trabajo se acerca al problema de una forma natural. Básicamente, la estructura de la entrevista abarca tres pasos: identificación de los entrevistados, preparación de la entrevista, realización de la entrevista y documentación de los resultados (protocolo de la entrevista). (29)

Tormenta de ideas: Es una técnica de reuniones en grupo cuyo objetivo es que los participantes muestren sus ideas de forma libre. Consiste en la acumulación de ideas e información sin evaluar las mismas. El grupo de personas que participa en estas reuniones no debe ser muy numeroso (máximo 10 personas), una de ellas debe asumir el rol de moderador de la sesión, pero sin carácter de controlador. Como técnica de captura de requisitos es sencilla de usar y de aplicar, puesto que no

requiere tanto trabajo en grupo como éste. Además suele ofrecer una visión general de las necesidades del sistema, pero normalmente no sirve para obtener detalles concretos del mismo, por lo que suele aplicarse en los primeros encuentros. (30)

Para realizar y guiar el levantamiento de requisitos se utilizó la técnica de la entrevista al cliente o usuario final, lo que ayudó a definir lo que éste desea. Luego se utilizó la técnica de tormenta de ideas, que se llevó a cabo por el equipo de trabajo, para analizar otras funcionalidades posibles y sugerencias que se le podían plantear al cliente. Durante de esta tormenta de ideas se utilizó la técnica de introspección, que consiste en ver las cosas desde el punto de vista del cliente. Otra técnica que se utilizó fue la de los casos de uso, la que consiste en llegar al requerimiento basándose en las funciones del sistema.

Lista de Requisitos funcionales

CU Visualizar Expediente de Fabricante.

RF 1 Buscar y visualizar expediente de fabricante.

RF 1.1 Realizar la búsqueda por nombre del fabricante.

RF 1.2 Realizar la búsqueda por número del expediente.

RF 1.3 Realizar la búsqueda por país del fabricante.

RF 1.4 Realizar la búsqueda por fecha de inscripción del fabricante.

CU Gestionar RAPAC.

RF 2.1 Insertar datos de un plan en el RAPAC.

RF 2.2 Actualizar datos de un plan.

RF 2.3 Buscar un plan.

RF 2.3.1 Permitir realizar la búsqueda por número de auditoría.

RF 2.3.2 Permitir realizar la búsqueda por fabricante.

RF 2.3.3 Permitir realizar la búsqueda por fecha de auditoría.

RF 2.3.4 Permitir realizar la búsqueda por fecha de entrega.

RF 2.3.5 Permitir realizar la búsqueda por fecha de revisión del plan.

RF 2.4 Imprimir el registro.

CU Gestionar RPA

RF 3.1 Insertar un programa en el registro.

RF 3.2 Actualizar un programa.

RF 3.3 Buscar un programa.

RF 3.3.1 Permitir realizar la búsqueda por fabricante.

RF 3.3.2 Permitir realizar la búsqueda por fecha de la auditoría.

RF 3.4 Imprimir el registro.

CU Gestionar RPC

RF 4.1 Insertar nuevo plan en el RPC.

RF 4.2 Actualizar datos de un plan en el RPC.

RF 4.3 Buscar un plan.

RF 4.3.1 Permitir la búsqueda por fabricante.

RF 4.4 Imprimir el registro.

CU Gestionar planilla LCCF.

RF 5.1 Registrar datos en la planilla Lista de Chequeo de Calidad para Fabricante.

RF 5.2 Actualizar datos en la planilla.

RF 5.3 Buscar y visualizar planilla.

RF 5.3.1 Permitir realizar la búsqueda por fabricante.

RF 5.4 Imprimir planilla.

RF 5.5 Posibilitar realizar la estadística de los datos de la planilla Lista de Chequeo de Calidad para Fabricante.

CU Gestionar planilla Plan de Auditoría.

RF 6.1 Registrar datos en la planilla Plan de Auditoría.

RF 6.2 Actualizar datos en la planilla Plan de Auditoría.

RF 6.3 Buscar y visualizar planilla.

RF 6.3.1 Permitir realizar la búsqueda por código de auditoría.

RF 6.4 Imprimir planilla.

CU Gestionar aviso.

RF 7.1 Insertar un nuevo aviso correspondiente a una fecha dentro de un registro.

RF 7.2 Actualizar el estado de un aviso.

RF 7.3 Buscar y visualizar un aviso.

RF 7.4 Eliminar aviso.

CU Gestionar planilla IA.

RF 8.1 Insertar datos en la planilla IA.

RF 8.2 Actualizar los datos en la planilla IA.

RF 8.3 Buscar y visualizar la planilla IA.

RF 8.3.1 Permitir la búsqueda por el código de Auditoría.

RF 8.4 Imprimir planilla.

CU Autenticar usuario.

RF 9.1 Autenticar usuario.

RF 9.2 Terminar sesión de usuario.

CU Gestionar usuario.

RF 10.1 Crear usuario.

RF 10.2 Eliminar usuario.

RF 10.3 Actualizar usuario.

RF 10.4 Buscar y visualizar usuario.

RF 10.4.1 Permitir realizar la búsqueda por nombre completo.

RF 10.4.2 Permitir realizar la búsqueda por nombre de usuario.

CU Realizar réplica BD.

RF 11 Realizar réplica de la BD.

CU Gestionar réplica BD.

RF 12.1 Realizar salva de la BD.

RF 12.2 Cargar salva de BD.

RF 12.3 Planificar réplica de la BD.

CU Gestionar planilla Registro Asistencia.

RF 13.1 Insertar datos en la planilla.

RF 13.2 Actualizar datos en la planilla.

RF 13.3 Buscar una planilla.

RF 13.4 Imprimir planilla.

CU Generar aviso.

RF 14.1 Avisar cuando se acerque la fecha de vencimiento de un plan de acciones correctivas.

RF 14.2 Avisar cuando se acerque la fecha de vencimiento de un plan de calidad.

RF 14.3 Avisar cuando se acerque la fecha de realizar una auditoría.

Lista de Requisitos no Funcionales

Usabilidad

RNF 1.1 El tiempo de entrenamiento requerido para que usuarios normales operen el sistema no debe ser muy extenso.

Los usuarios deben aprender a usar el sistema en un corto período de tiempo, se debe tener en cuenta que son médicos y profesionales no familiarizados con herramientas automatizadas.

RNF 1.2 Empleo de menús para facilitar la utilización del sistema.

El sistema debe implementar una interfaz sencilla con el empleo de menús para facilitar el uso del sistema.

RNF 1.3 Implementar una fácil navegación.

El sistema debe contar con una fácil navegación, indicar en todo momento al usuario en que lugar de la aplicación esta trabajando, mediante señalizaciones, para de esta forma hacer más fácil el uso del mismo.

Fiabilidad

RNF 2.1 Disponible el 100% del tiempo de utilización.

El sistema debe estar disponible para su utilización cada vez que el usuario lo requiera. Solamente debe cesar la disponibilidad cuando sea necesario realizar mantenimiento.

RNF 2.2 El tiempo medio de reparación no debe exceder las 4 horas.

Después de una falla del sistema esta se debe corregir en menos de 4 horas.

RNF 2.3 No deben ocurrir errores críticos.

Entre los errores críticos están: pérdida total de la información, o inhabilitación para el uso de ciertas partes del funcionamiento del sistema.

Eficiencia

RNF 3.1 El tiempo de respuesta por transacción no debe exceder 1 seg.

El sistema debe realizar una transacción sin exceder el tiempo máximo de 1 seg.

RNF 3.2 La capacidad debe ser de hasta 10 usuarios.

El sistema debe poder alojar hasta 10 usuarios al mismo tiempo.

Soporte

RNF 4.1 Asimilar nuevos módulos.

El sistema debe ser compatible con nuevos sistemas automatizados que intervengan en el funcionamiento de la empresa.

Interfaz de usuario

RNF 5.1 La interfaz de usuario debe ser amigable.

La aplicación será diseñada con una interfaz amigable, fácil de usar por el usuario, de manera que agilice y facilite el trabajo con el software, pues el sistema brindará servicios tanto a usuarios familiarizados con ambientes informáticos como a otros no familiarizados.

Portabilidad

RNF 6.1 El sistema debe ser multiplataforma.

El sistema será multiplataforma debido a que será capaz de ejecutarse sobre diferentes sistemas operativos sin importar sus versiones, y sin necesidad de modificar su código fuente.

Seguridad

RNF 7.1 El sistema debe restringir la autenticación de usuarios.

El sistema contará con una política de seguridad diseñada en base a la restricción de usuarios en dependencia del nivel jerárquico que cumpla su rol dentro del mismo.

Hardware

RNF 8.1 Hardware.

Requerimientos mínimos para el servidor: Computador Pentium a 2.8 GHz o superior. 512 MB RAM o superior. 4 GB de espacio libre en Disco Duro como mínimo. Requerimientos mínimos para la conexión del cliente: Computador Pentium a 333 MHz o superior. 128 MB RAM o superior. MODEM o red con TCP-IP para conexión al servidor.

Software

RNF 9.1 Software.

Se requiere del Navegador Mozilla para la visualización de la aplicación.

Técnicas de validación de requisitos empleadas

El objetivo de la validación de requisitos es descubrir problemas en la redacción de requisitos antes de comprometer recursos en su implementación. Los requisitos deben revisarse para descubrir omisiones, conflictos y ambigüedades.

Para la validación de los requisitos se emplearon varias técnicas:

Revisiones (Reviews)

Es la fórmula más empleada para validación. Un grupo de personas (incluyendo usuarios) se ocupan de revisar el documento de requisitos. Consta de tres fases:

1. Búsqueda de problemas.
2. Reunión.
3. Acuerdos.

Prototipado

Permite descubrir con rapidez si el usuario se encuentra satisfecho, o no, con los requisitos. Se presentan las interfaces no funcionales del software para que el usuario final entienda gráficamente que es lo que el sistema va a hacer y muestre su conformidad o inconformidad.

2.3.2. Justificación de los actores del sistema.

Un actor del sistema representa cualquier persona, entidad o sistema externo al mismo que interactúe con él para intercambiar información o como receptor pasivo de esta. Pueden representar el rol que juega una o varias personas, un equipo o un sistema automatizado.

Cada trabajador del negocio (inclusive si fuera un sistema ya existente) que tiene actividades a automatizar es un candidato a actor del sistema. Si algún actor del negocio va a interactuar con el sistema, entonces también será un actor del sistema.

Actor	Descripción
Administrador	Actor que se relaciona con casos de uso relacionados con la administración del sistema.
Especialista DS	Especialista del Departamento de Supervisión que realiza la gestión de información utilizando el sistema.
Usuario	Generalización de los actores del sistema que necesitan autenticarse.
Reloj	Actor que modela un evento externo del sistema, que ocurre con cierta periodicidad; dependiendo solamente de cuando un usuario realice alguna acción sobre la aplicación.

Tabla 10 Justificación de los actores del sistema.

2.3.3. Diagrama casos de uso del sistema

El diagrama de casos de uso del sistema representa la interacción entre los actores del mismo y los casos de uso con que se relacionan. Se muestra, además, la relación entre casos de uso, ya sea extensión, inclusión, asociación o generalización/especialización y las relaciones entre actores, por asociación o generalización/especialización. Sirve como contrato entre el cliente y los desarrolladores. Se utiliza como entrada esencial para las actividades de análisis, diseño y prueba. (Ver Figura 9 Diagrama de casos de uso del sistema)

Aplicación de patrones de casos de uso

Los casos de uso son artefactos narrativos que describen, bajo la forma de acciones y reacciones, el comportamiento del sistema desde el punto de vista del usuario. Para la obtención de los casos de uso se emplearon los patrones de casos de uso mencionados en el capítulo uno.

El patrón CRUD propone formar un caso de uso a partir de los requisitos funcionales relacionados con las acciones de insertar, listar o mostrar, modificar, y eliminar una determinada información. (Anexo 4: Patrones de casos de uso. Solución del patrón CRUD)

Para cada una de las planillas y registros que son objeto de informatización se repiten estos requisitos, que se agruparon dando lugar a los casos de uso Gestionar. Se le suman además operaciones relacionadas con otras funciones específicas.

Toda aplicación Web debe permitir la autenticación de los usuarios en el sistema para habilitar los privilegios que le permitan acceder al resto de los servicios. De igual forma es necesario que el usuario se desconecte o finalice su sesión al terminar. A partir del estudio de los requisitos funcionales relacionados con la autenticación de usuarios, se evidencia la necesidad de aplicar el patrón Login (Autenticarse) que propone la creación de un caso de uso independiente que englobe estas funcionalidades. Esta solución a la vez requiere de un caso de uso que gestione la información de los usuarios que casi siempre es inicializado por el administrador del sistema. (Anexo 4: Patrones de casos de uso. Solución del patrón Login)

La utilización de este patrón en la solución, dió como resultado la identificación del caso de uso Autenticar usuario inicializado por el actor Usuario y del caso de uso Gestionar usuario inicializado por el Administrador.

Un caso de uso no debe ser inicializado por más de un actor, cuando dos o más actores intervienen en un mismo caso de uso se debe modelar de forma tal que no se incumpla con esta primera regla. Existe un patrón de casos de uso que le da solución a este problema de una forma sencilla. El patrón Múltiples actores propone crear un nuevo actor del sistema que represente una generalización. (Anexo 4: Patrones de casos de uso. Solución del patrón Múltiples actores)

La solución del patrón aplicada al diagrama de casos de uso da como resultado la creación de un nuevo actor Usuario, que generaliza las responsabilidades de los actores Especialista DS y Administrador para representar su interacción con el caso de uso Autenticar usuario.

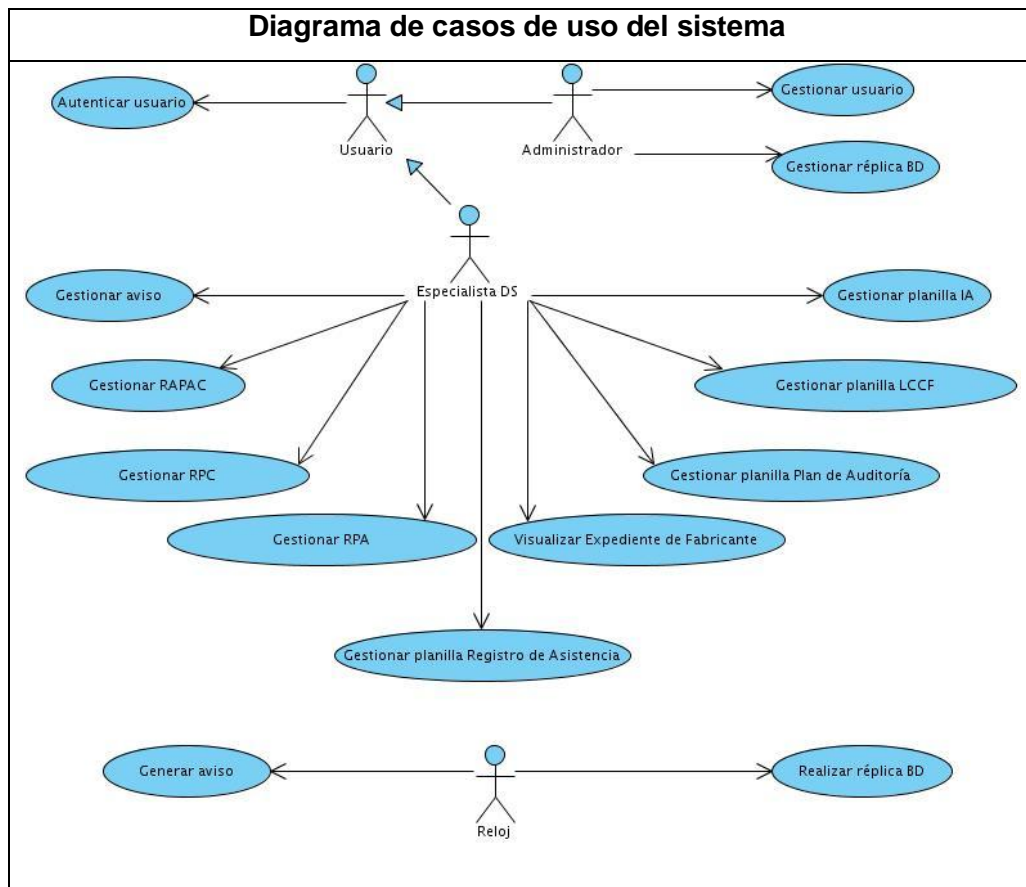


Figura 9 Diagrama de casos de uso del sistema.

2.3.4. Listado de los casos de uso del sistema.

1. Descripción textual resumida del caso de uso del sistema Visualizar Expediente de Fabricante

Caso de Uso:	Visualizar Expediente de Fabricante.
Actores:	Especialista DS(inicia)
Resumen:	El CU se inicia cuando el Especialista DS pide visualizar un expediente. El sistema realiza la búsqueda pertinente y muestra los datos definidos por el Especialista DS finalizando el CU.
Referencias	RF 1, RF 1.1, RF 1.2, RF 1.3, RF 1.4

Tabla 11 Descripción textual resumida del caso de uso del sistema Visualizar Expediente de Fabricante.

2. Descripción textual resumida del caso de uso del sistema Gestionar RAPAC

Caso de Uso:	Gestionar RAPAC.
Actores:	Especialista DS (Inicia)
Resumen:	<p>El CU inicia cuando el Especialista DS necesita realizar una de las siguientes acciones.</p> <p>Insertar un nuevo plan en el registro, el Especialista especifica los datos e indica guardar, el sistema guarda los datos y finaliza el caso de uso.</p> <p>Actualizar un plan, el Especialista selecciona el plan a actualizar e indica guardar los cambios, el sistema actualiza los nuevos datos y finaliza el caso de uso.</p> <p>Buscar un plan, el Especialista DS especifica las opciones y los criterios de búsqueda, el sistema muestra los planes que coincidan, finaliza así el caso de uso.</p> <p>Imprimir el registro, El Especialista DS indica imprimir el registro, el sistema muestra una vista previa, el usuario acepta y finaliza así el CU.</p>
Referencias	RF 2.1, RF 2.2, RF 2.3, RF 2.3.1, RF 2.3.2, RF 2.3.3, RF 2.3.4, RF 2.3.5, RF 2.4

Tabla 12 Descripción textual resumida del caso de uso del sistema Gestionar RAPAC.

3. Descripción textual resumida del caso de uso del sistema Gestionar RPA

Caso de Uso:	Gestionar RPA.
Actores:	Especialista DS (Inicia)
Resumen:	<p>El CU inicia cuando el Especialista DS necesita realizar una de las siguientes acciones.</p> <p>Insertar un nuevo programa en el registro, el Especialista DS especifica los datos e indica guardar, el sistema guarda los datos y finaliza el caso de uso.</p> <p>Actualizar un plan, el Especialista DS selecciona el programa a actualizar e indica guardar los cambios, el sistema actualiza los nuevos datos y finaliza el caso de uso.</p> <p>Buscar un programa, el Especialista DS especifica las opciones y los criterios de búsqueda, el sistema muestra los programas que coincidan, finaliza así el caso de uso.</p> <p>Imprimir el registro, El Especialista DS indica imprimir el registro, el sistema muestra una vista previa, el usuario acepta y finaliza así el CU.</p>
Referencias	RF 3.1, RF 3.2, RF 3.3, RF 3.3.1, RF 3.3.2, RF 3.4

Tabla 13 Descripción textual resumida del caso de uso del sistema Gestionar RPA.

4. Descripción textual resumida del caso de uso del sistema Gestionar RPC

Caso de Uso:	Gestionar RPC.
Actores:	Especialista DS (Inicia)
Resumen:	<p>El CU inicia cuando el Especialista DS necesita realizar una de las siguientes acciones.</p> <p>Insertar un nuevo plan en el registro, el Especialista DS especifica los datos e indica guardar, el sistema guarda los datos y finaliza el caso de uso.</p> <p>Actualizar un plan, el Especialista DS selecciona el plan a actualizar e indica guardar los cambios, el sistema actualiza los nuevos datos y finaliza el caso de uso.</p> <p>Buscar un plan, el Especialista DS especifica las opciones y los criterios de búsqueda, el sistema muestra los planes que coincidan, finaliza así el</p>

	<p>caso de uso.</p> <p>Imprimir el registro, El Especialista DS indica imprimir el registro, el sistema muestra una vista previa, el usuario acepta y finaliza así el CU.</p>
Referencias	RF 4.1, RF 4.2, RF 4.3, RF 4.3.1, RF 4.4

Tabla 14 Descripción textual resumida del caso de uso del sistema Gestionar RPC.

5. Descripción textual resumida del caso de uso del sistema Gestionar planilla LCCF

Caso de Uso:	Gestionar planilla LCCF
Actores:	Especialista DS (Inicia)
Resumen:	<p>El CU se inicia cuando el Especialista DS decide realizar una de la siguientes acciones:</p> <p>Registrar los datos en la planilla, introduce los datos, el sistema los guarda finalizando el CU.</p> <p>Actualizar datos en la planilla, el Especialista DS selecciona la planilla a modificar, inserta los datos, el sistema registra los cambios finalizando así el CU.</p> <p>Buscar y visualizar planilla, el Especialista DS escoge las opciones de búsqueda e inserta los criterios de búsqueda, el sistema muestra las coincidencias finalizando así el CU.</p> <p>Imprimir planilla, el Especialista DS escoge la planilla a imprimir, el sistema muestra las opciones de impresión e imprime el documento, finalizando así el CU.</p> <p>Calcular estadísticas, el Especialista DS selecciona las estadísticas a calcular, el sistema realiza los cálculos y muestra los resultados en forma de tabla.</p>
Referencias	RF 5.1, RF 5.2, RF 5.3, RF 5.3.1, RF 5.4, RF 5.5

Tabla 15 Descripción textual resumida del caso de uso del sistema Gestionar planilla LCCF.

6. Descripción textual resumida del caso de uso del sistema Gestionar planilla Plan de Auditoría

Caso de Uso:	Gestionar planilla Plan de Auditoría.
Actores:	Especialista DS (Inicia)
Resumen:	<p>El CU se inicia cuando el Especialista DS decide realizar una de la siguientes acciones:</p> <p>Registrar los datos en la planilla, introduce los datos y el sistema los guarda finalizando el CU.</p> <p>Actualizar datos en la planilla, el Especialista DS selecciona la planilla a modificar, inserta los datos, el sistema registra los cambios finalizando así el CU.</p> <p>Buscar y visualizar planilla, el Especialista DS escoge las opciones de búsqueda e inserta os criterios de búsqueda, el sistema muestra las coincidencia finalizando así el CU.</p> <p>Imprimir planilla, el Especialista DS escoge la planilla a imprimir, el sistema muestra las opciones de impresión e imprime el documento, finalizando así el CU.</p>
Referencias	RF 6.1, RF 6.2, RF 6.3, RF 6.3.1, RF 6.4

Tabla 16 Descripción textual resumida del caso de uso del sistema Gestionar planilla Plan de Auditoría.

7. Descripción textual resumida del caso de uso del sistema Gestionar planilla IA

Caso de Uso:	Gestionar planilla IA
Actores:	Especialista DS (Inicia)
Resumen:	<p>El CU se inicia cuando el Especialista DS decide realizar una de la siguientes acciones:</p> <p>Registrar los datos en la planilla, introduce los datos y el sistema los guarda finalizando el CU.</p> <p>Actualizar datos en la planilla, el Especialista DS selecciona la planilla a modificar, inserta los datos, el sistema registra los cambios finalizando así el CU.</p> <p>Buscar y visualizar planilla, el Especialista DS escoge las opciones de</p>

	<p>búsqueda e inserta os criterios de búsqueda, el sistema muestra las coincidencia finalizando así el CU.</p> <p>Imprimir planilla, el Especialista DS escoge la planilla a imprimir, el sistema muestra las opciones de impresión e imprime el documento, finalizando así el CU.</p>
Referencias	RF 8.1, RF 8.2, RF 8.3, RF 8.3.1, RF 8.4

Tabla 17 Descripción textual resumida del caso de uso del sistema Gestionar planilla IA.

8. Descripción textual resumida del caso de uso del sistema Gestionar planilla Registro de Asistencia

Caso de Uso:	Gestionar planilla Registro de Asistencia
Actores:	Especialista DS (Inicia)
Resumen:	<p>El CU se inicia cuando el Especialista DS decide realizar una de la siguientes acciones:</p> <p>Registrar los datos en la planilla, introduce los datos y el sistema los guarda finalizando el CU.</p> <p>Actualizar datos en la planilla, el Especialista DS selecciona la planilla a modificar, inserta los datos, el sistema registra los cambios finalizando así el CU.</p> <p>Buscar y visualizar planilla, el Especialista DS escoge las opciones de búsqueda e inserta los criterios de búsqueda, el sistema muestra las coincidencias finalizando así el CU.</p> <p>Imprimir planilla, el Especialista DS escoge la planilla a imprimir, el sistema muestra las opciones de impresión e imprime el documento, finalizando así el CU.</p>
Referencias	RF 13.1, RF 13.2, RF 13.3, RF 13.4

Tabla 18 Descripción textual resumida del caso de uso del sistema Gestionar planilla Registro de Asistencia.

9. Descripción textual resumida del caso de uso del sistema Generar aviso

Caso de Uso:	Generar aviso.
Actores:	Reloj (Inicia)

Resumen:	El caso de uso se inicia cuando se ejecuta la aplicación por alguno de los usuarios autenticados. Los reportes generales se generan de forma automática por el sistema. El caso de uso finaliza cuando los reportes son generados.
Referencias	RF 14.1, RF 14.2, RF 14.3

Tabla 19 Descripción textual resumida del caso de uso del sistema Generar aviso.

(Para ver descripciones textuales completas de los casos de uso del sistema ver Anexo 5: Descripciones textuales casos de uso del sistema.)

2.4. Conclusiones

- ✓ El estudio realizado a los procesos de la organización ayudó a comprender su entorno de trabajo por parte de los desarrolladores y otras personas involucradas. Además se identificaron posibles mejoras determinando los procesos que se deben automatizar.
- ✓ Se identificaron 64 requisitos funcionales con los cuales se llegó a una definición más formal de lo que el sistema debe hacer y ayudó a establecer un convenio entre desarrolladores y clientes de las características funcionales del software.
- ✓ A partir de la utilización de los patrones de casos de uso se determinaron los 14 casos de uso y la relación con los actores del sistema, esta relación se especificó en detalle en el diagrama de casos de uso del sistema.
- ✓ Para hacer una valoración del trabajo se emplearon técnicas de validación de requisitos, lo que dió como resultado la aprobación del cliente de los requerimientos identificados y aseguró la calidad de los mismos.

CAPÍTULO 3 DISEÑO DEL SISTEMA

En este capítulo se hace un estudio del diseño del sistema. Se representa la realización de casos de uso del diseño a través de los diagramas de clase del diseño y de interacción correspondientes a cada uno. A medida que se representan estos diagramas se hace alusión a las técnicas empleadas para obtener mejores resultados en la solución, específicamente los patrones de diseño empleados en la obtención de las clases del diseño y la estructuración de los diagramas. En el capítulo se abordan además los detalles del diseño de la base de datos mediante la representación del Modelo de Datos y la descripción de las tablas, incluyendo el Modelo de Despliegue. Además se describe el proceso de tratamiento de errores.

3.1. Descripción de estilos arquitectónicos y patrones de diseño

Los patrones arquitectónicos especifican un conjunto predefinido de subsistemas con sus responsabilidades y una serie de recomendaciones para organizar los distintos componentes.

El patrón arquitectónico empleado en la solución es el Modelo Vista Controlador (MVC) que permite organizar los componentes de las aplicaciones Web de una forma más flexible, modular y reutilizable. Como su nombre lo dice, MVC consiste en separar lo mejor posible las capas de Modelo (el Sistema de Gestión de Base de Datos; los objetos que interactúan con la base de datos y efectúan los procesos pesados o “lógica de negocios”), la Vista (la presentación final de los datos procesados al cliente, comúnmente en formato HTML, y el código que provee de datos dinámicos a dichas páginas) y el Controlador (la capa que se encarga de recibir las entradas de datos del usuario, delegar el trabajo a los Modelos apropiados e invocar las Vistas que correspondan; representa la separación clara entre el Modelo (lógica de negocio) y la Vista (interfaz gráfica), gracias a un controlador que los mantiene desacoplados).

La **capa del modelo** define la lógica de negocio, la base de datos pertenece a esta capa. Encapsula los datos y las funcionalidades. Es independiente de cualquier representación de salida y/o comportamiento de entrada. Esta es la representación específica de la información con la cual el sistema opera. La lógica de datos asegura la integridad de estos y permite derivar nuevos datos.

La **vista** es lo que utilizan los usuarios para interactuar con la aplicación, los gestores de plantillas pertenecen a esta capa. En Symfony la capa de la vista está formada principalmente por plantillas en

PHP. Este presenta el modelo en un formato adecuado para interactuar, usualmente la interfaz de usuario.

El **controlador** recibe las entradas, traducidas a solicitudes de servicio para el modelo. Es un bloque de código que realiza llamadas al modelo para obtener los datos y se los pasa a la vista para que los muestre al usuario. Este responde a eventos, usualmente acciones del usuario e invoca cambios en el modelo.

Patrones de diseño:

Una de las principales ventajas a señalar en el uso de un framework es que está basado en patrones de diseño. Symfony está concebido de tal manera que obliga al uso de diferentes patrones de diseño. Como se había mencionado anteriormente los patrones de diseño empleados en la solución pertenecen a dos principales grupos. El primero de ellos es el conjunto de patrones GRASP, la utilización de estos patrones ha ayudado a refinar el diseño y a asignar las responsabilidades de las distintas clases de diseño, haciéndolas más sencillas, reutilizables y encapsuladas.

Experto: Al incluir Propel como ORM, este genera las clases para la gestión de las entidades con las responsabilidades debidamente asignadas. Esta es precisamente la solución que propone el patrón Experto, pues cada una de estas clases cuenta con un conjunto de funcionalidades relacionadas directamente con la entidad que representan.

Alta Cohesión: Una de las características principales del framework Symfony es la organización del trabajo en cuanto a la estructura del proyecto, lo cual permite crear y trabajar con clases con una alta cohesión. En la solución se encuentra la clase LCCFActions que contiene varias funcionalidades las que poseen un propósito único, no desempeñado por el resto de los elementos, siendo estas funcionalidades las encargadas de controlar las acciones de las plantillas. Esto hace posible que el software sea flexible a cambios sustanciales con efecto mínimo.

Bajo Acoplamiento: En la capa del Modelo se encuentran las clases que implementan la lógica de negocio y de acceso a datos, estas clases tienen pocas asociaciones con otras de la Vista o el Controlador por lo que la dependencia en este caso es baja, poniéndose de manifiesto el patrón Bajo Acoplamiento.

Los otros patrones involucrados pertenecen al conjunto de patrones GOF.

Decorador: El patrón Decorador se ve implementado en el comportamiento de Symfony que dispone de un archivo llamado `layout.php` que contiene las etiquetas `<html>` y `<body>`. Este archivo, que también se denomina plantilla global, almacena el código HTML que es común a todas las páginas de la aplicación, para no tener que repetirlo en cada página. El contenido de la plantilla se integra en el `layout`, o si se mira desde el otro punto de vista, el `layout` decora la plantilla.

Factoría: Symfony dispone de un mecanismo para extender las clases en tiempo de ejecución, algo mucho más avanzado que una simple herencia de clases. Incluso es posible reemplazar las clases del núcleo de Symfony por clases propias, utilizando las opciones de las factorías utilizadas por Symfony (las factorías se basan en el patrón de diseño "factories"). Una vez realizadas las modificaciones, se pueden encapsular en forma de plugin para poder reutilizarlas en otras aplicaciones o por parte de otros programadores de Symfony.

3.2. Modelo del Diseño

El Modelo de Diseño se utiliza para concebir y documentar el diseño de una aplicación. Es un modelo de objeto que describe la realización de casos de uso, y sirve como una abstracción del Modelo de Implementación y del código fuente. El Modelo de Diseño se utiliza como entrada esencial para actividades en el flujo de trabajo Implementación y en el flujo de trabajo Prueba. Es un artefacto integral que abarca todas las clases de diseño, subsistemas, paquetes, colaboraciones y las relaciones entre ellos.

3.2.1. Realización de casos de uso del diseño

La realización de los casos de uso del diseño incluyen una representación estática y una representación dinámica de cómo se lleva a cabo el flujos de acciones. En el diseño la representación estática está dada por los diagramas de clases del diseño y la dinámica por los diagramas de interacción.

Diagramas de clases del diseño

Un diagrama de clases del diseño es una representación concreta de lo que se debe implementar. Estos diagramas representan la parte estática del sistema a través de la representación de las clases y sus relaciones.

Las clases más significativas son las controladoras, que contienen toda la lógica de la aplicación, utilizan el modelo y definen variables para la vista. En Symfony estas clases son llamadas siguiendo el patrón nombreModuloActions y se define una para cada caso de uso. Las acciones son métodos con el nombre que sigue el patrón executeNombreAccion que se encuentran dentro de estas clases. (Anexo 6: Descripciones de clases del diseño.)

Para modelar las clases del diseño de un sistema que emplea Symfony, es necesario delimitar bien lo que es del framework de lo que es propio del caso de uso. Los componentes de Symfony se encapsulan en un subsistema que contiene todos los elementos del framework que actúan o intervienen en la realización de los casos de uso, pero al ser propios de Symfony, la manera en que trabajan es transparente al programador. Para que el diagrama de clases del diseño no se tornara complejo se decidió modelar estos componentes como un subsistema con el nombre de Componentes de Symfony. Otro elemento fundamental en los diagramas es el ORM que utiliza Symfony para realizar el mapeo objeto-relacional, el cual proporciona la persistencia de los objetos y un servicio de consulta, en este caso Propel y se representa también como un subsistema. Para lograr una mayor organización, los diagramas están organizados de tal manera que cada elemento del diseño se ubica en la capa a la que responde según las definidas por el patrón MVC.

Diagrama de clases del diseño CU Gestionar RAPAC

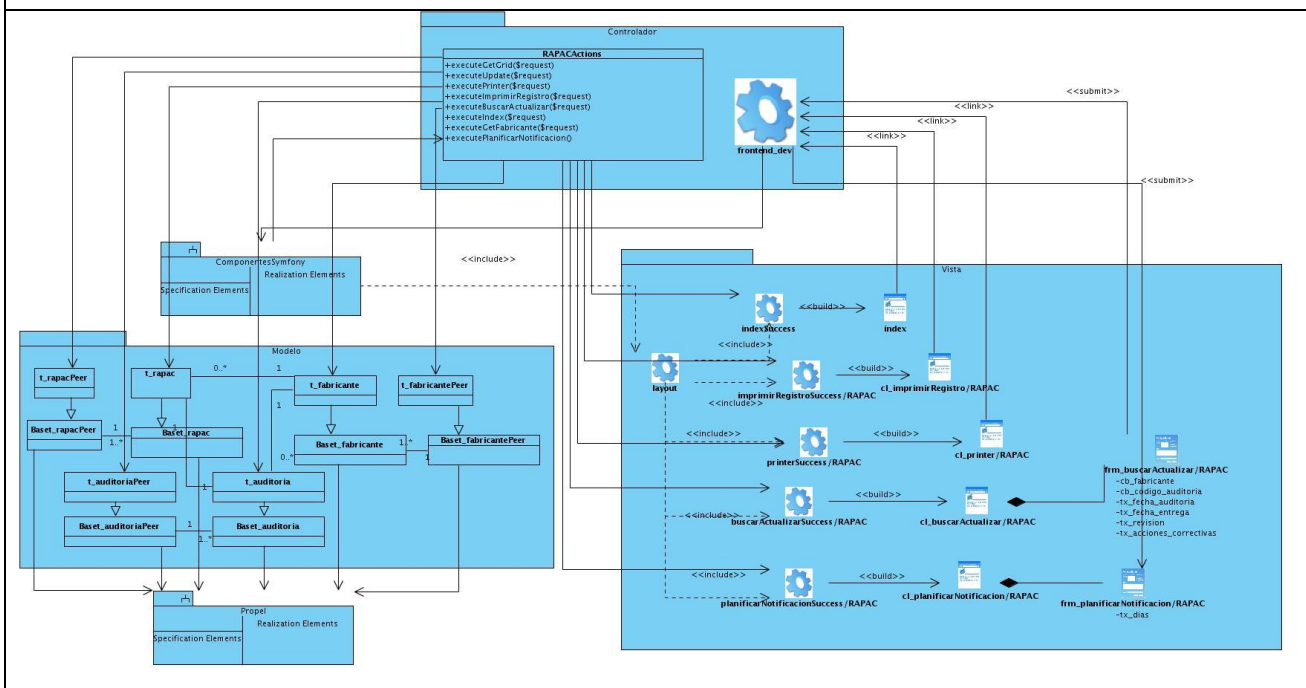


Figura 10 Diagrama de clases del diseño CU Gestionar RAPAC.

Diagrama de clases del diseño CU Gestionar planilla LCCF

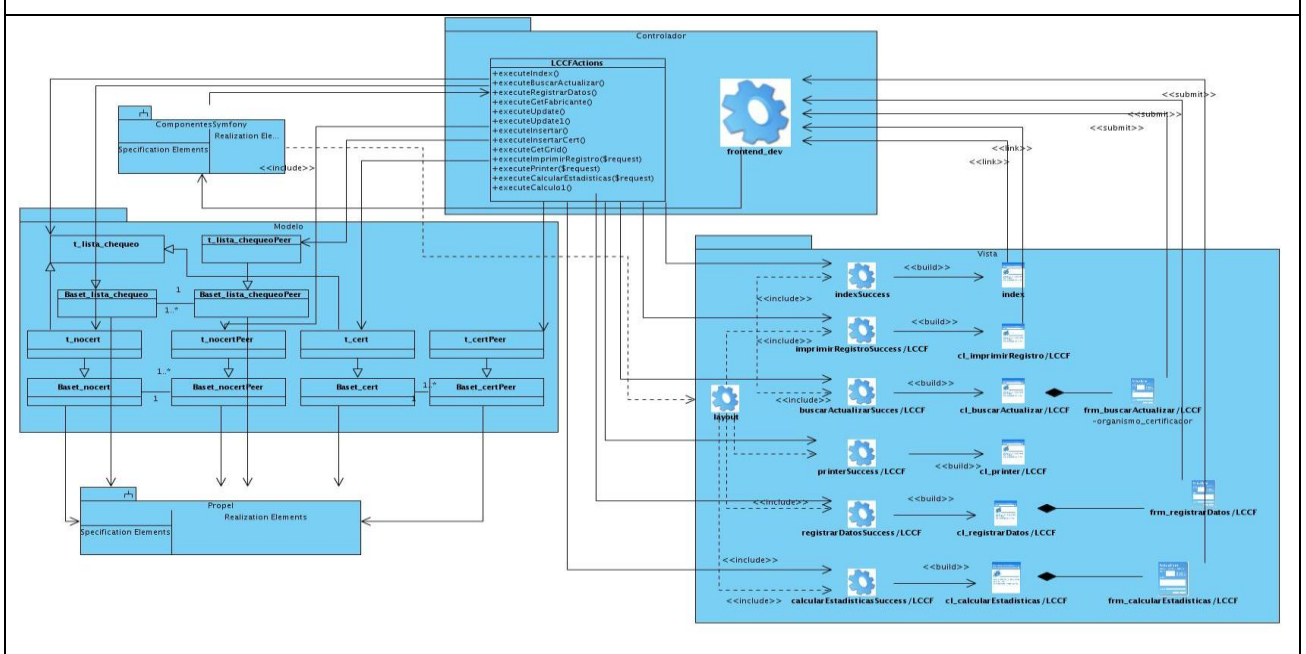


Figura 11 Diagrama de clases del diseño CU Gestionar planilla LCCF.

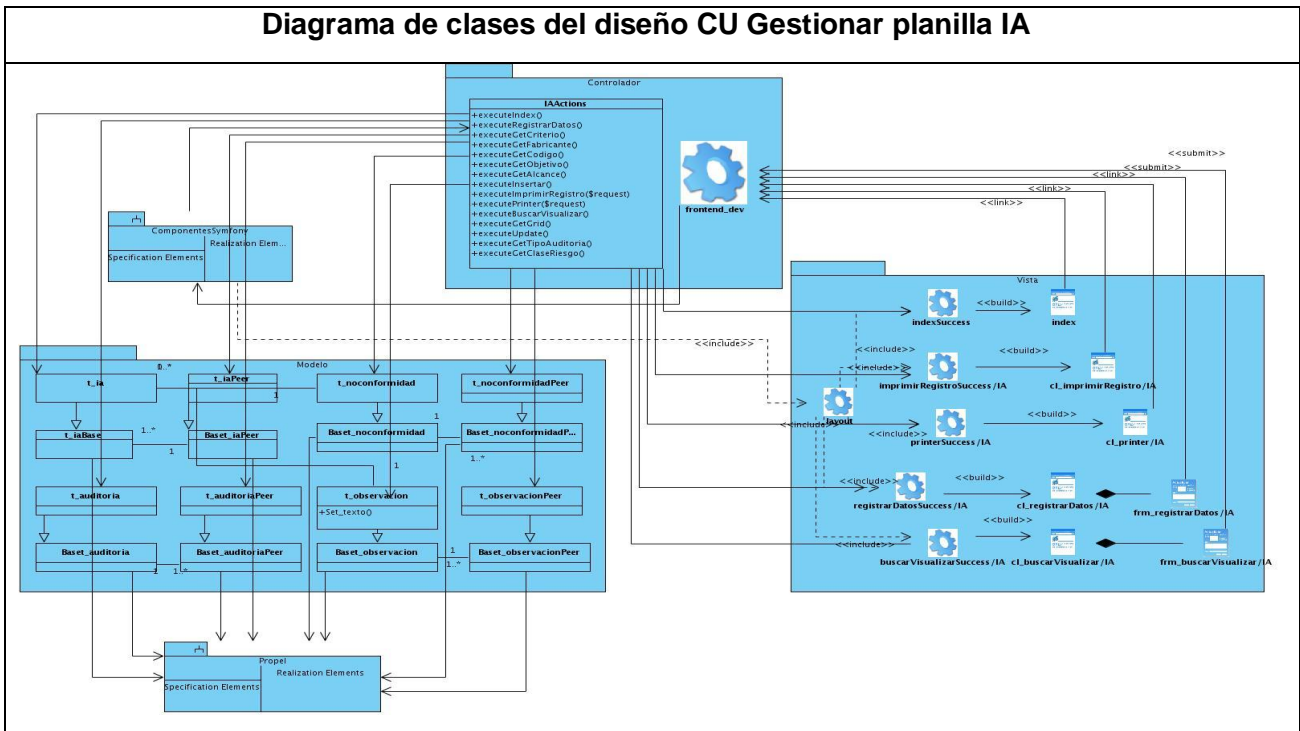


Figura 12 Diagrama de clases del diseño CU Gestionar planilla IA.

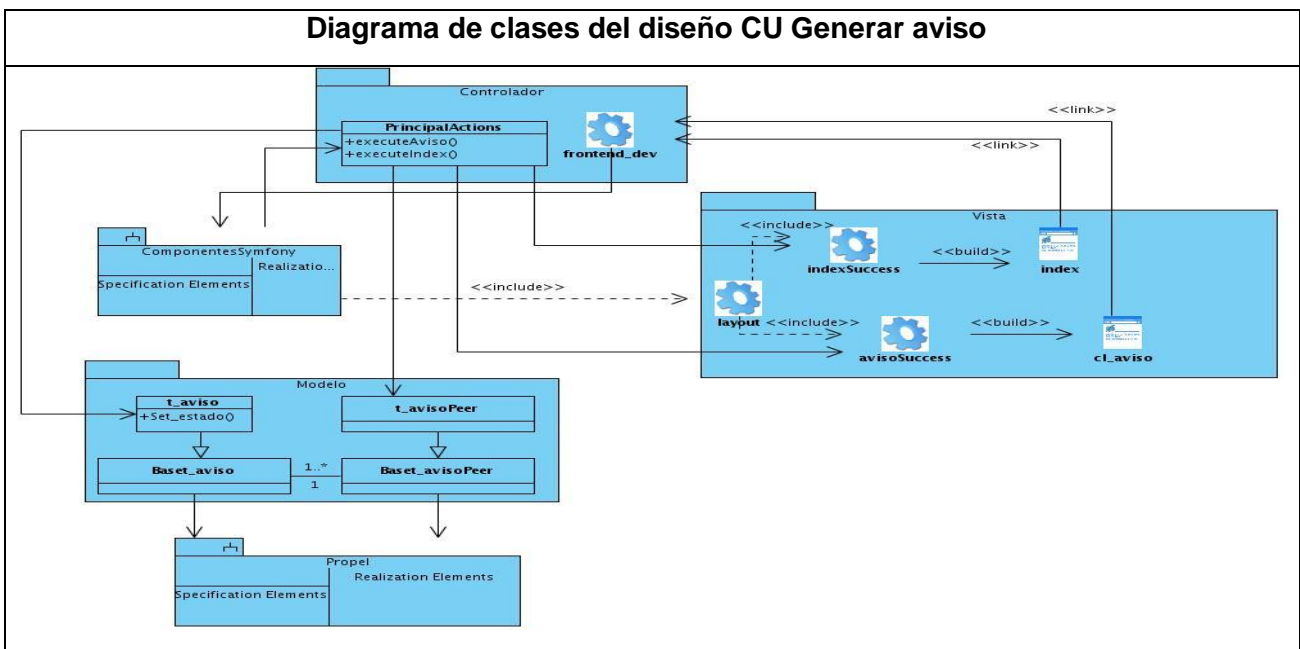


Figura 13 Diagrama de clases del diseño CU Generar aviso.

Diagrama de clases persistentes

Las clases persistentes representan información de larga duración o que persiste en el tiempo, es decir, es la información que se requiere almacenar para gestionarla en cualquier momento. El diagrama de clases persistentes representa dichas clases y las relaciones entre ellas (asociación, agregación/composición y generalización/especialización).

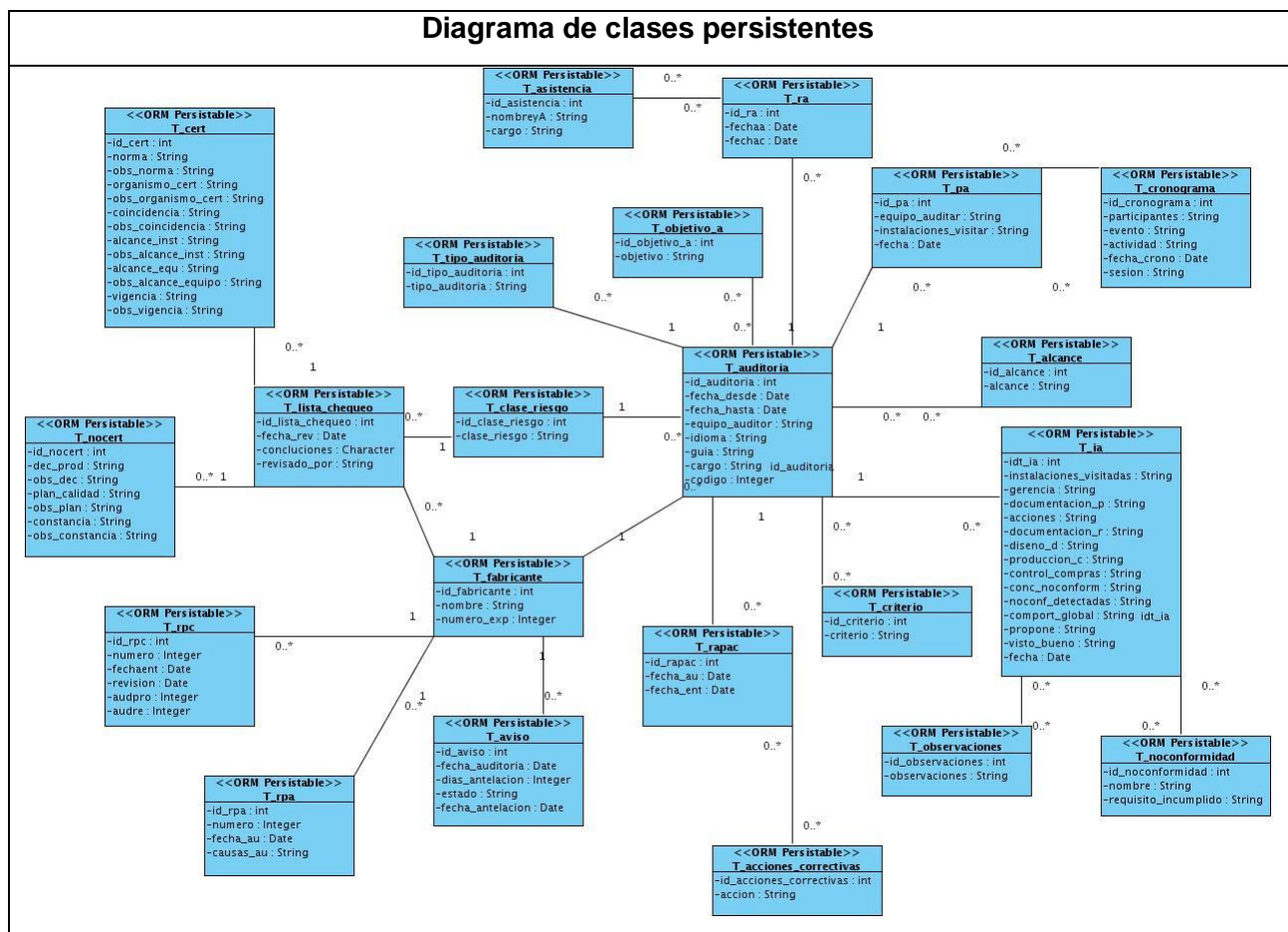


Figura 14 Diagrama de clases persistentes.

Diagramas de interacción del diseño. Secuencia

Los diagramas de interacción muestran las interacciones entre objetos mediante transferencia de mensajes entre objetos o subsistemas, por lo que son empleados para modelar aspectos dinámicos del sistema. Los diagramas de colaboración y de secuencia son dos tipos de diagramas de interacción que son semánticamente equivalentes pero sin embargo los diagramas de colaboración destacan el orden estructural de los objetos que interactúan y los de secuencia destacan el orden temporal de los mensajes. Para la realización de los casos de uso del diseño es más factible el empleo de los

diagramas de secuencia ya que representan con más claridad el flujo de las acciones que debe realizar el sistema.

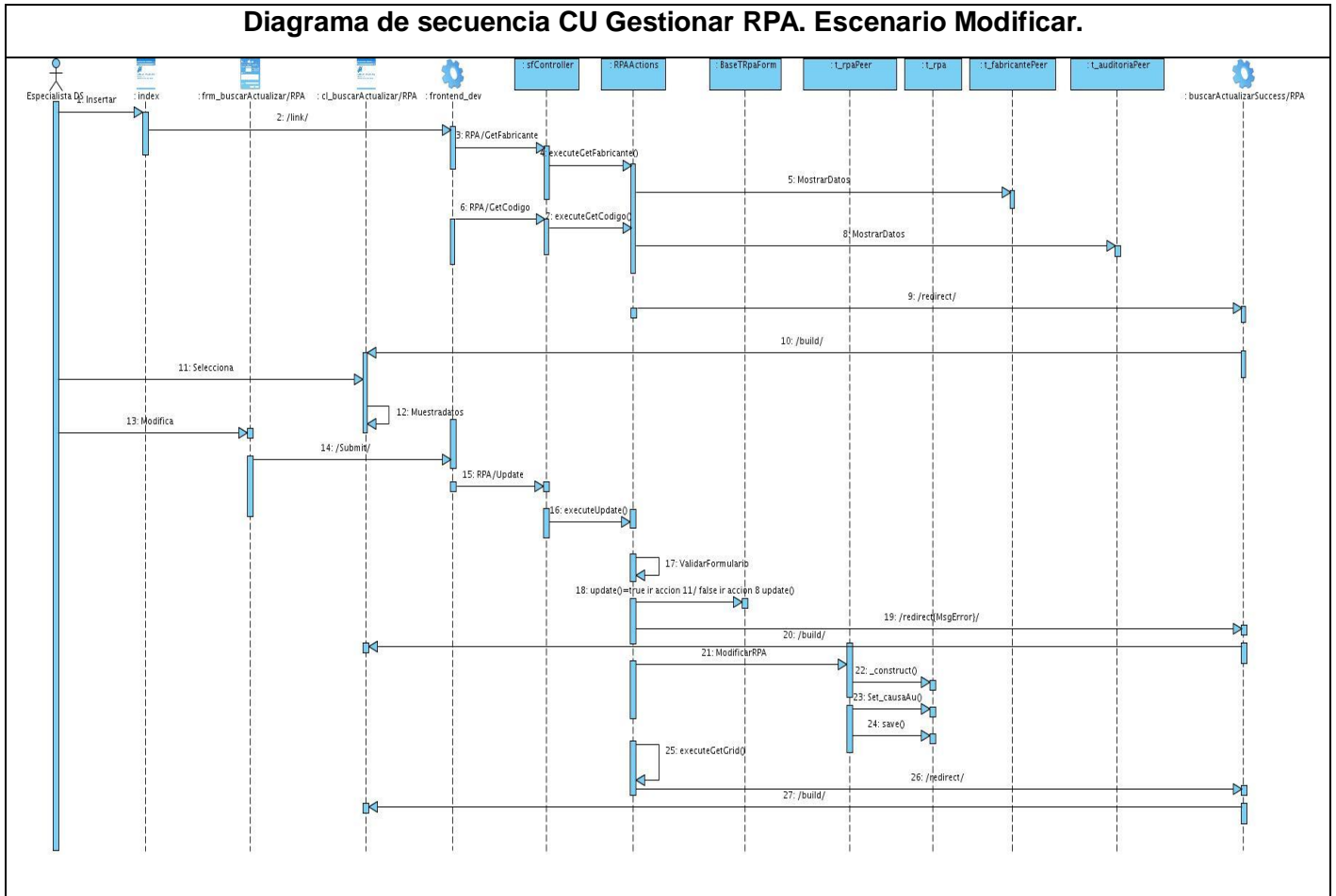


Figura 15 Diagrama de secuencia CU Gestionar RPA. Escenario Modificar.

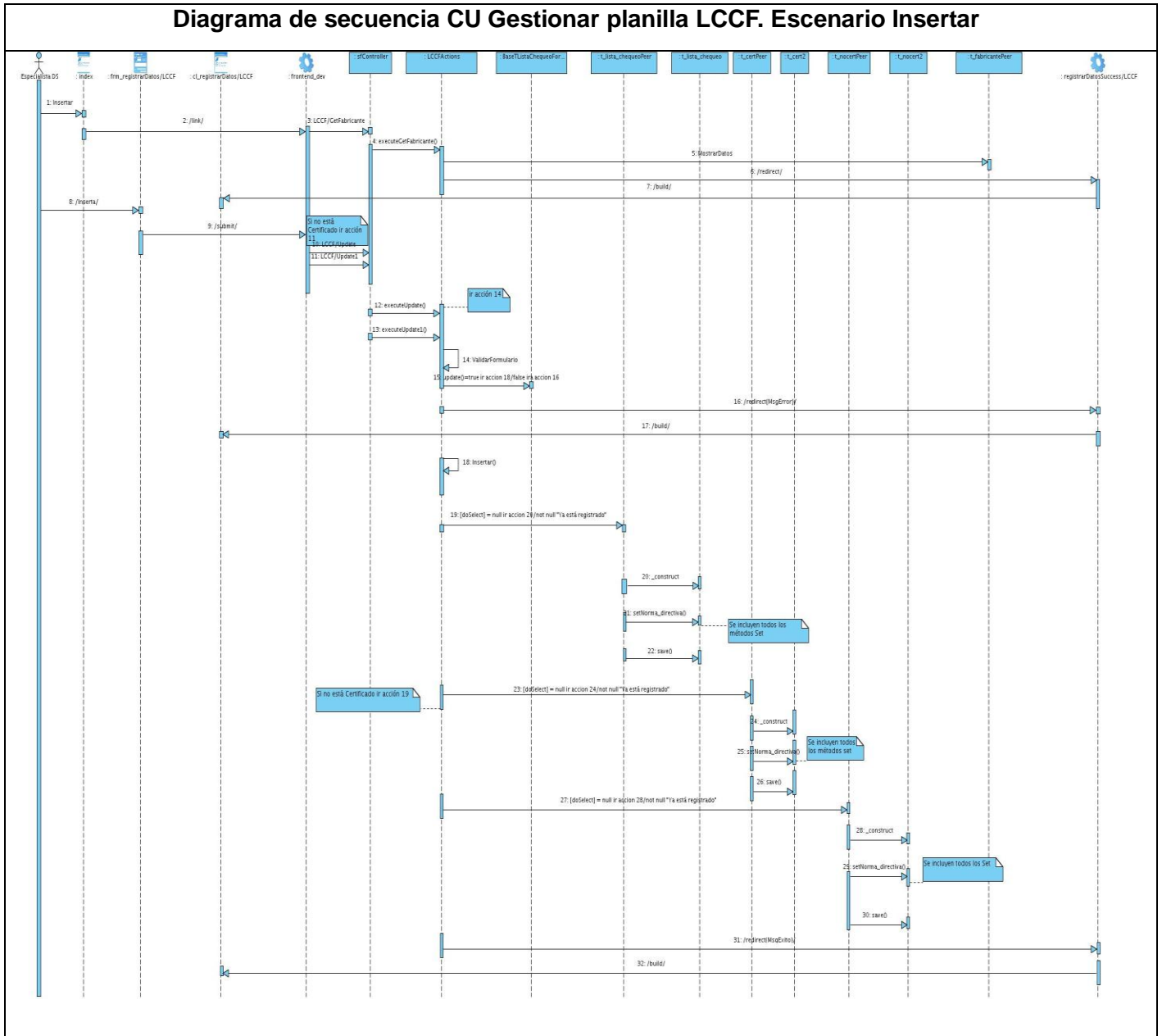


Figura 16 Diagrama de secuencia CU Gestionar planilla LCCF. Escenario Insertar.

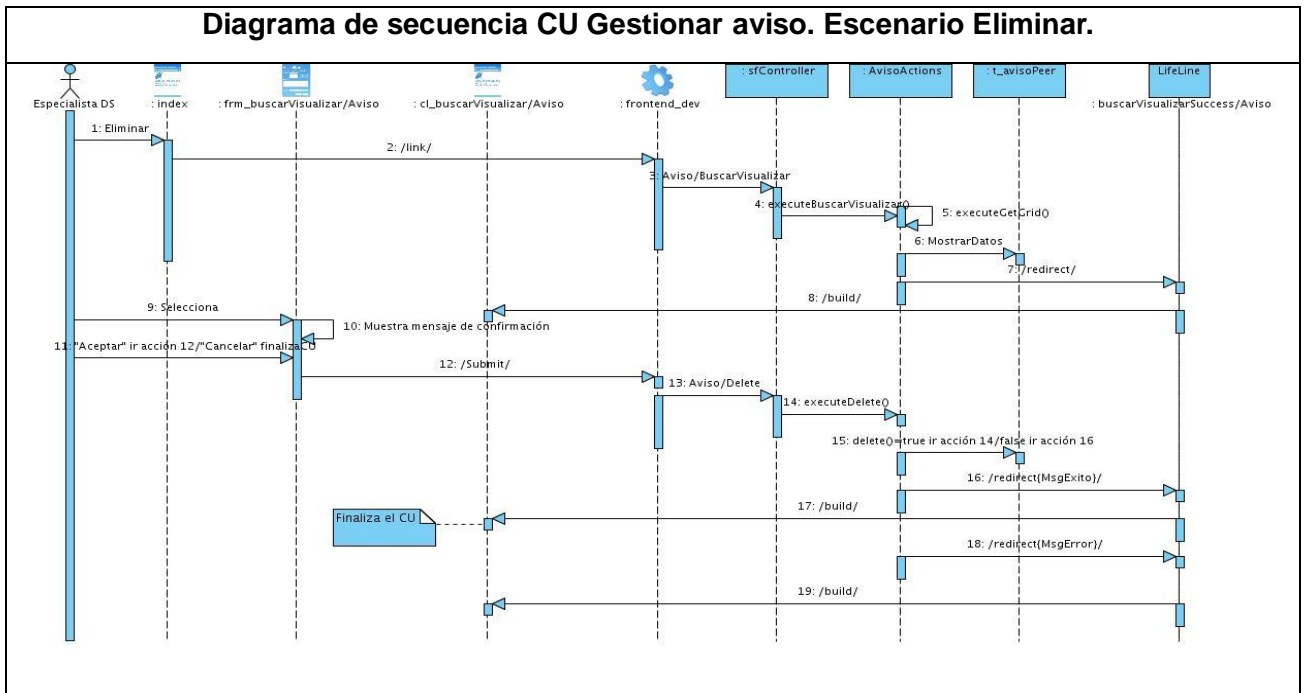


Figura 17 Diagrama de secuencia CU Gestionar aviso. Escenario Eliminar.

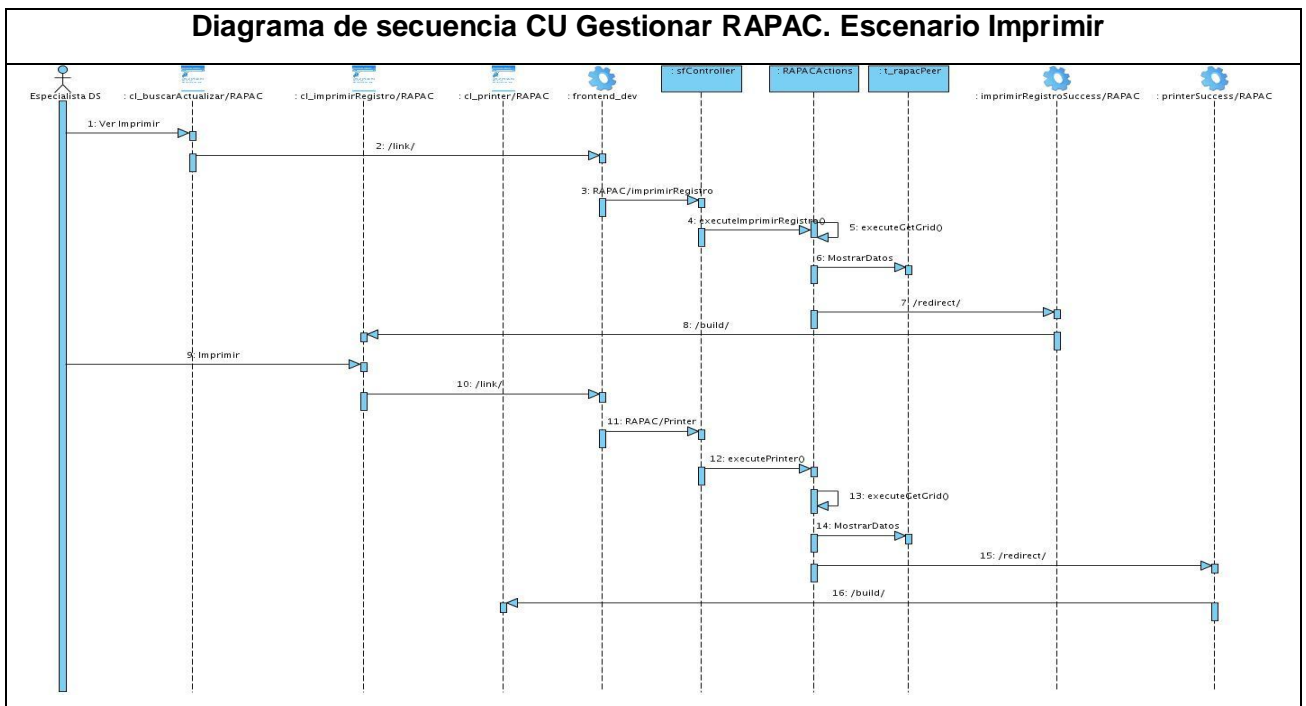


Figura 18 Diagrama de secuencia CU Gestionar RAPAC. Escenario Imprimir.

3.2.2. Modelo de Datos

El modelo de datos se utiliza para describir la estructura lógica y física de la información persistente gestionada por el sistema. También incluye elementos del modelo para procedimientos almacenados, desencadenantes (triggers), restricciones, etc. Se crea partiendo de las clases persistentes del Modelo de Diseño y se utiliza para definir la correlación entre las clases de diseño persistentes y las estructuras de datos persistentes, y para definir las estructuras de datos persistentes. Los elementos del modelo de datos poseen representaciones UML que permiten su representación.

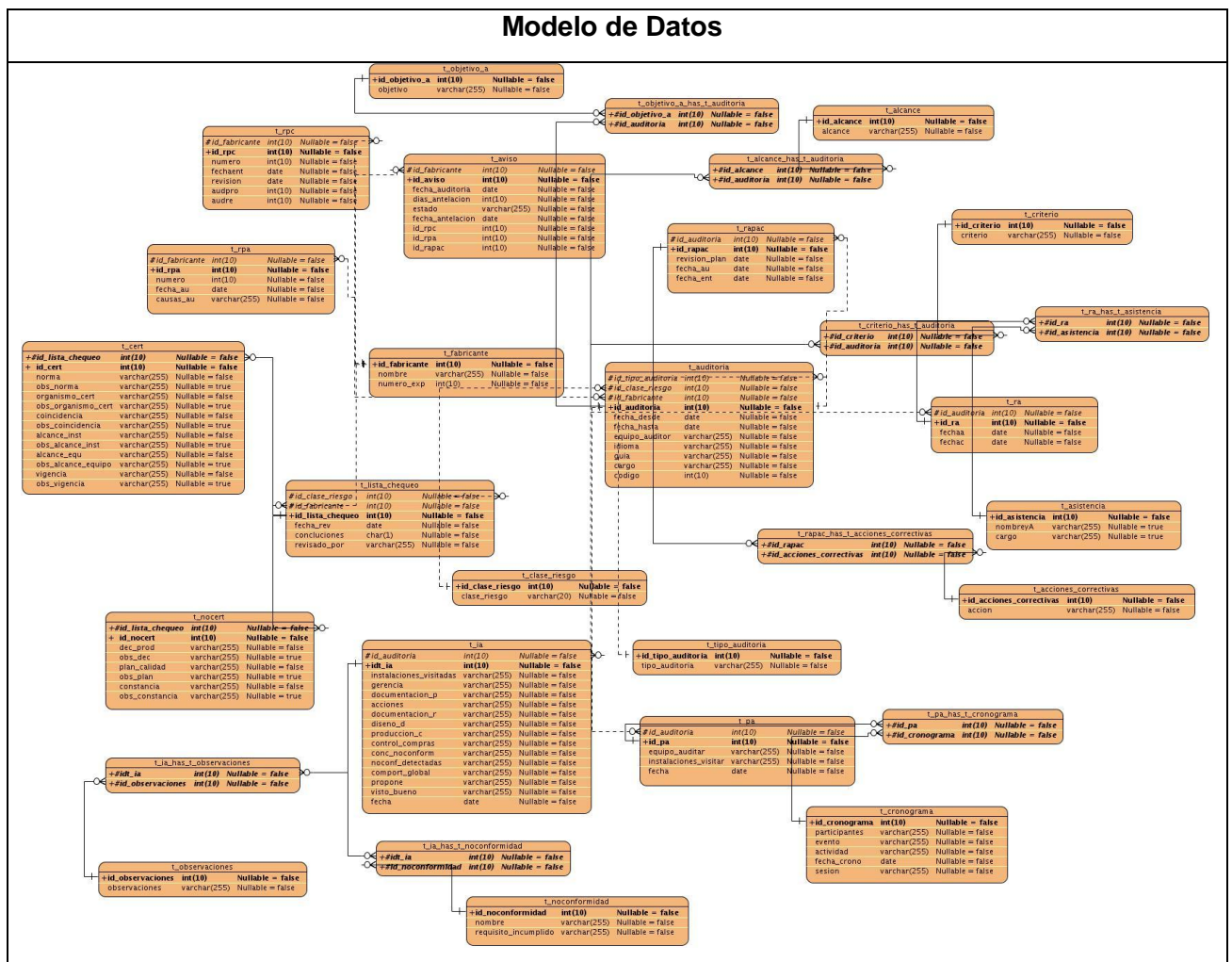


Figura 19 Modelo de Datos.

Descripción de las tablas de la base de datos (Anexo 7: Descripciones de las tablas de la base de datos.)

3.2.3. Modelo de Despliegue

El Modelo de Despliegue muestra la disposición física de los distintos nodos que componen un sistema y el reparto de los componentes sobre dichos nodos. La vista de despliegue representa la disposición de las instancias de componentes de ejecución en instancias de nodos conectados por enlaces de comunicación.

Se puede observar lo siguiente sobre el modelo de despliegue:

- Cada nodo representa un recurso de cómputo, normalmente un procesador o un dispositivo hardware similar.
- Los nodos poseen relaciones que representan medios de comunicación entre ellos, tales como HTTP, USB, TCP/IP, etc.
- Puede describir diferentes configuraciones de red, incluidas las configuraciones para pruebas y para simulación.
- La funcionalidad (los procesos) de un nodo se define por los componentes que se distribuyen sobre ese nodo.

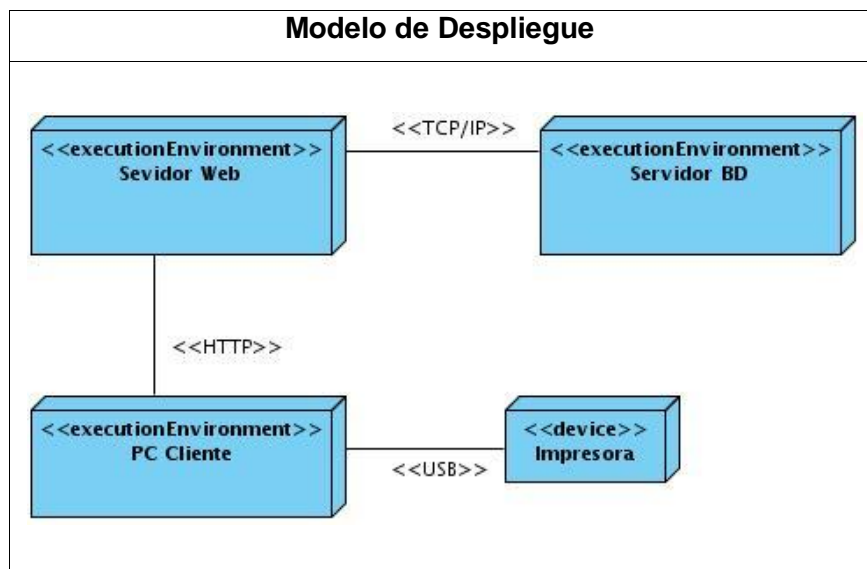


Figura 20 Modelo de Despliegue.

3.3. Tratamiento de errores

El tratamiento de errores se lleva a cabo para evitar que estos errores se conviertan en problemas mayores que afecten la funcionalidad de la aplicación. Con la validación de los datos enviados por los usuarios al servidor se evitan muchos de estos errores.

La validación de formularios se puede realizar en el lado del servidor y en el lado del cliente. La validación en el servidor es obligatoria para no corromper la base de datos con datos incorrectos. La validación en el lado del cliente es opcional, pero mejora la experiencia de usuario.

La validación en el lado del cliente se realiza mediante la interfaz Web que evita que el usuario asuma un papel activo en la introducción de la información empleando funcionalidades del framework de presentación ExtJS. Uno de los métodos de validación más usados fue la prohibición de campos en blanco: **allowBlank: false**. Otro tipo de validación ampliamente usada fueron las llamadas “built-in” con el uso de los **vtype**, usadas para restringir y validar la entrada de datos de los usuarios y enviar mensajes de error. Estas funciones son de gran uso ya que pueden trabajar en cualquier escenario dado a que utilizan expresiones regulares.

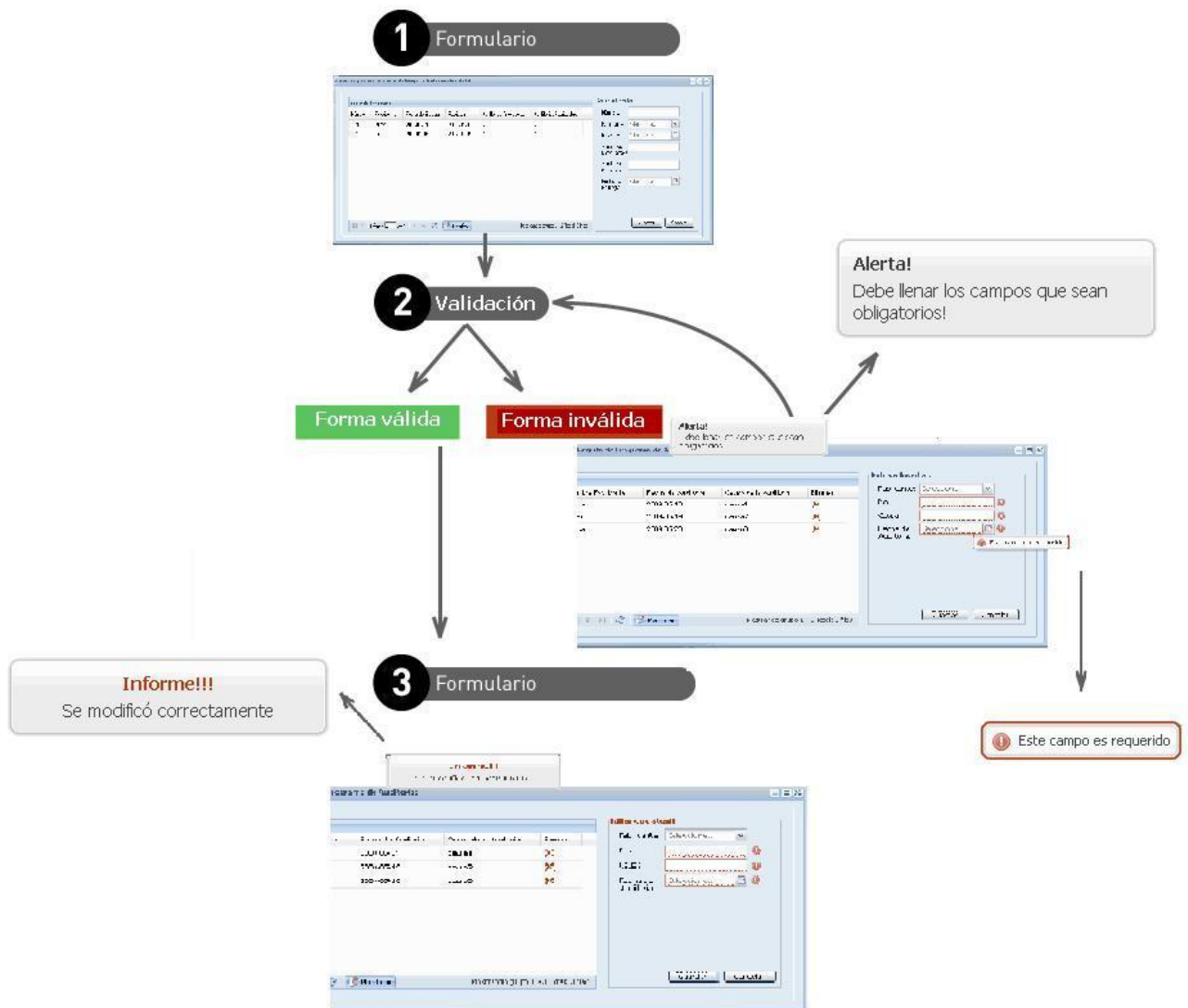


Figura 21 Validación de formularios.

Además se cuenta con cuadros de opción y menú de selección lo cual facilita la entrada de datos.

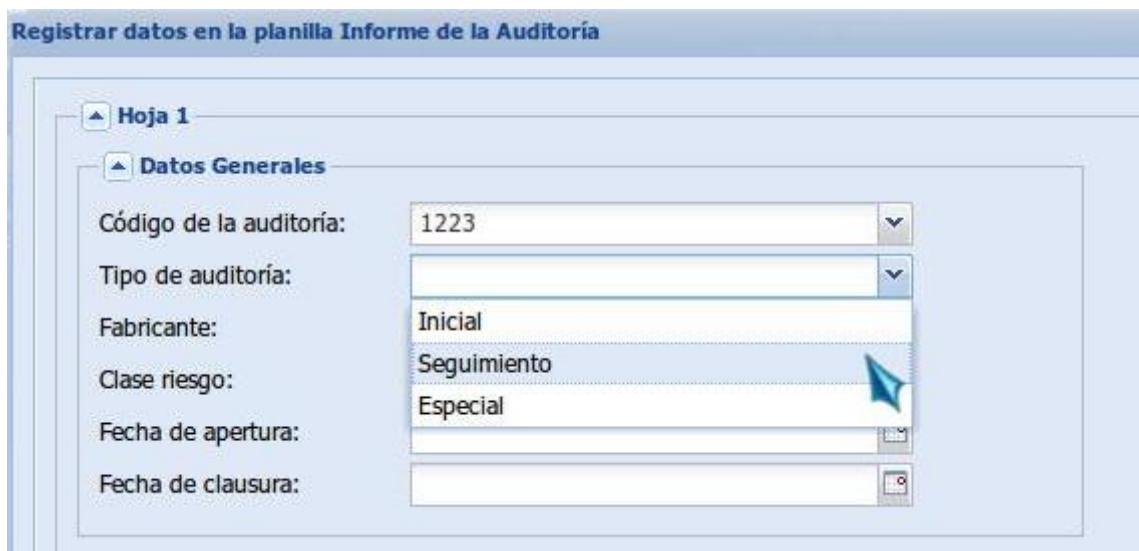


Figura 22 Interfaz de la aplicación. Muestra un menú de selección.

Se validan además las funciones correspondientes a la eliminación o modificación de datos para evitar que se realicen estas acciones por accidente. Cuando se desea eliminar algún elemento de la base de datos se muestra un mensaje de confirmación al usuario, al igual que cuando desee modificar alguna información. Solo se realizan las respectivas acciones si el usuario confirma que la desea realizar.



Figura 23 Mensajes de confirmación mostrados al usuario.

Para las validaciones del lado del servidor, la validación de cada campo es hecha por objetos de la clase `sfValidatorBase`, para validar las formas se crearon objetos por cada uno de los campos a validar usando el método `setValidators()`.

Cuando se hace uso del método `setValidators()` se crea un objeto de la clase `sfValidatorSchema`. Esta clase define una colección de validadores dentro de la cual se encuentran los utilizados para la validación. Estos campos que aparecen en los validadores coinciden directamente con los campos del esquema de la base de datos. De esta forma se verifica que los datos sean correctos.

3.4. Conclusiones

- ✓ El uso del patrón MVC, al igual que la aplicación de patrones de diseño, ha hecho más fácil el diseño de las clases permitiendo que se obtengan clases mejor diseñadas, modulares, flexibles y reutilizables.
- ✓ Se modeló un diagrama de clases del diseño por cada caso de uso y un diagrama de secuencia por cada escenario de caso de uso.
- ✓ A partir del diseño de clases persistentes y la estructuración del diagrama de clases persistentes se obtuvo el modelo de datos, lo que facilitó el diseño de 30 tablas de la base de datos.
- ✓ La validación de entradas de datos al sistema disminuyó las posibilidades de introducir datos incorrectos a la base de datos y de realizar acciones no deseadas por el usuario.
- ✓ La realización del diagrama de despliegue propició una visión de cómo está distribuido el sistema físicamente.

CAPÍTULO 4 IMPLEMENTACIÓN

En este capítulo se aborda el flujo de trabajo de Implementación, donde se describen sus principales artefactos, destacando el Modelo de Implementación que incluye componentes, subsistemas de implementación y diagramas de componentes. En este flujo de trabajo se comienza a implementar el sistema en términos de componentes a partir de las clases del diseño. Se hace una validación de lo que se ha implementado probando los componentes por separado antes de integrarlos al sistema para que funcione como un todo.

4.1. Modelo de Implementación

El Modelo de Implementación describe cómo los elementos del modelo de diseño, como las clases, se implementan en términos de componentes, ficheros de código fuente, ejecutables, etc. Describe también cómo se organizan los componentes de acuerdo a los mecanismos de estructuración y modularización disponibles en el entorno de implementación y lenguaje o lenguajes de implementación empleados, y cómo dependen los componentes unos de otros. (16) Esta descripción es de gran utilidad a la hora de implementar el sistema, facilita la organización del trabajo y lo hace más entendible a los desarrolladores.

4.1.1. Diagrama de componentes

Dentro del Modelo de Implementación se encuentran los diagramas de componentes. Un componente es el empaquetamiento físico de los elementos de un modelo, como son las clases en el modelo de diseño. El diagrama de componentes describe cómo se organizan los componentes de acuerdo con los mecanismos de estructuración y modularización disponibles en el entorno de implementación y en el lenguaje o lenguajes de programación utilizados, y cómo dependen los componentes unos de otros.

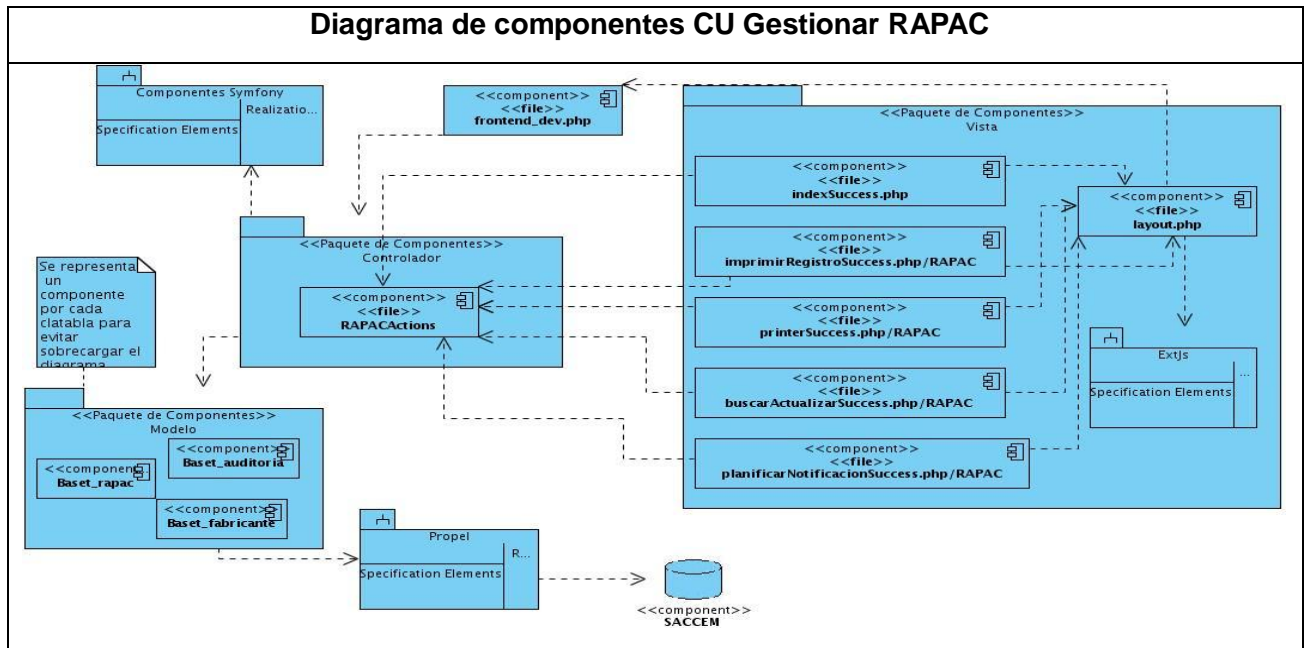


Figura 24 Diagrama de componentes CU Gestionar RAPAC.

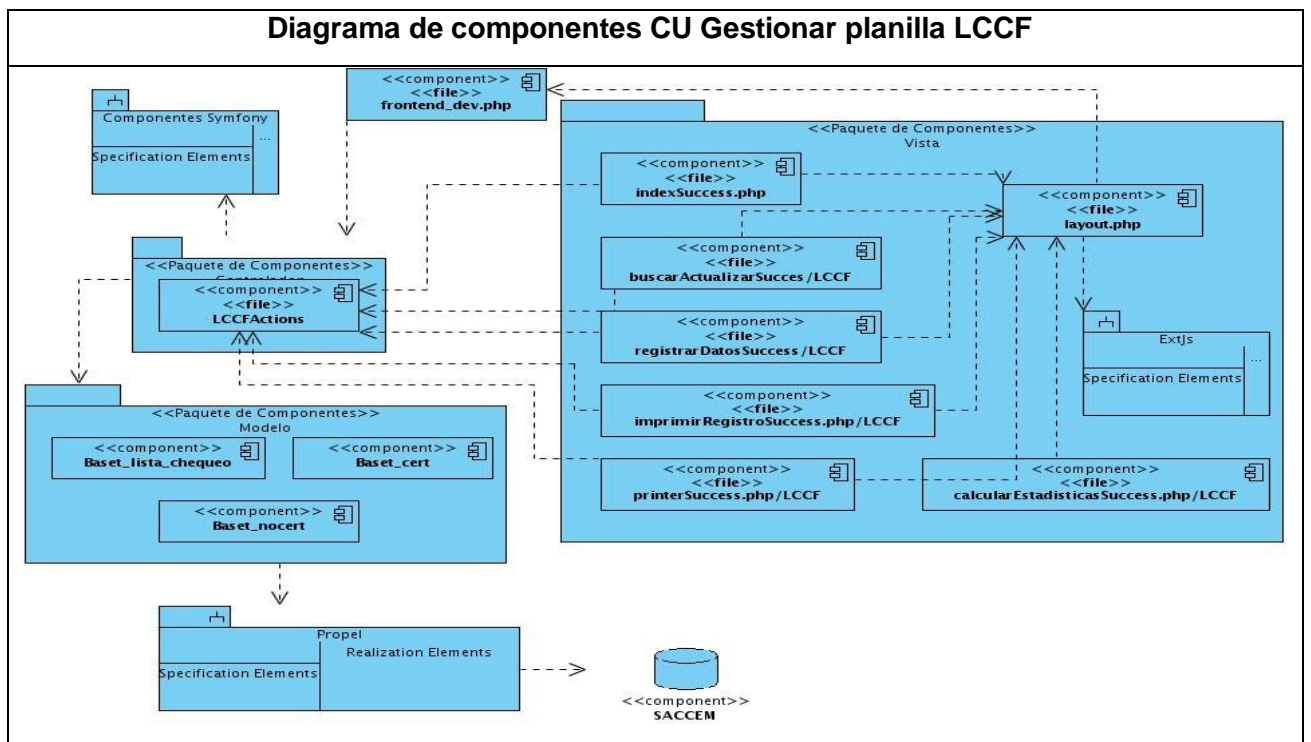


Figura 25 Diagrama de componentes CU Gestionar planilla LCCF.

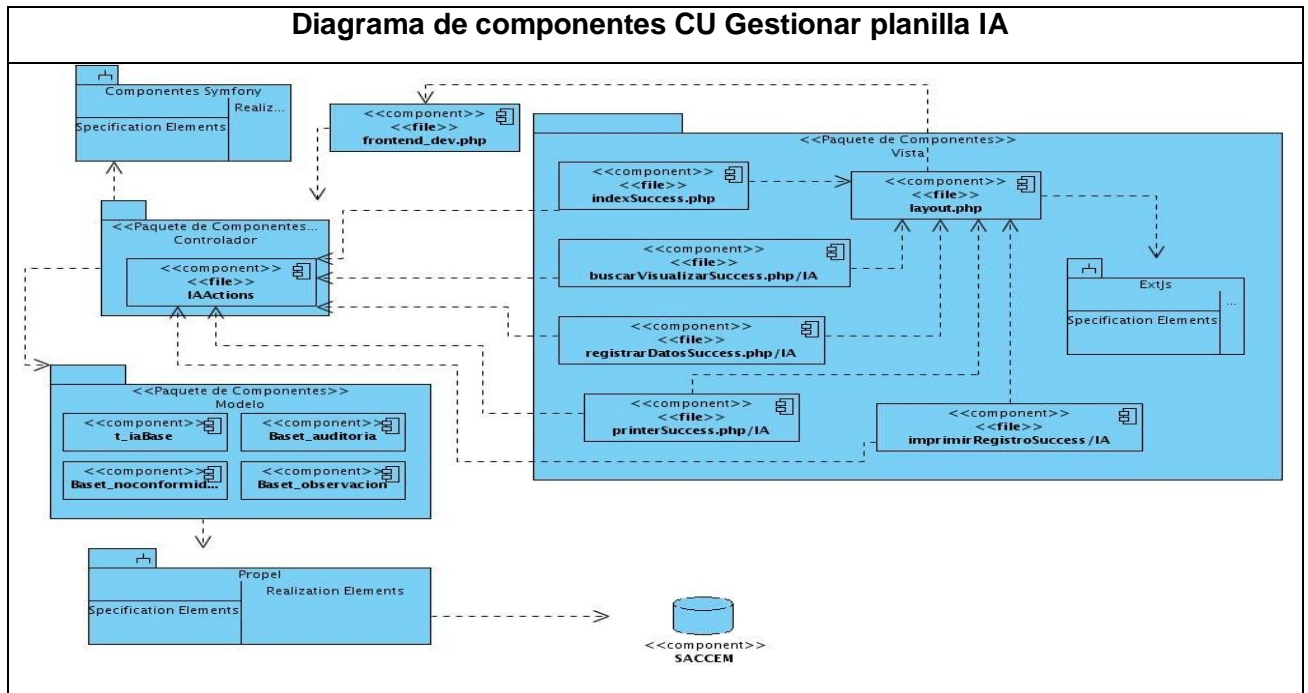


Figura 26 Diagrama de componentes CU Gestionar planilla IA.

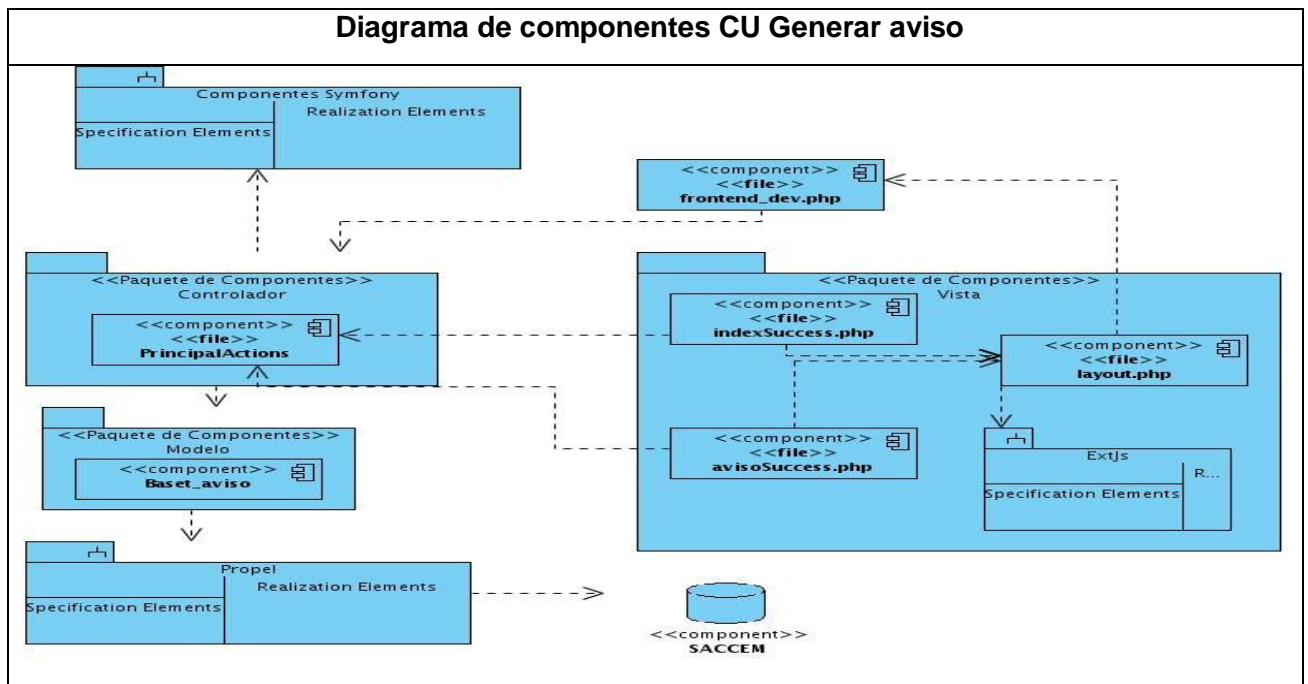


Figura 27 Diagrama de componentes CU Generar aviso.

4.2. Código fuente

Para obtener una versión funcional de la aplicación se deben implementar los componentes que se han definido, como resultado se obtienen archivos que contienen el código fuente de la aplicación.

El código fuente de un software es un conjunto de líneas de texto que son las instrucciones que debe seguir la computadora para ejecutar dicho programa. Por tanto, en el código fuente de un programa está descrito por completo su funcionamiento. Estas instrucciones son escritas en un lenguaje de programación que consiste en un conjunto de símbolos, reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos y expresiones.

4.2.1. Estándar de codificación

Un estándar de codificación comprende todos los aspectos de la generación de código. Usar técnicas de codificación sólidas y realizar buenas prácticas de programación con vistas a generar un código de alta calidad es de gran importancia para la calidad del software y para obtener un buen rendimiento.

Utilizando un buen estilo de codificación, se logra que el código cuente con algunas cualidades:

1. Extensibilidad: La facilidad con que se adapta el software a cambios de especificación. Un buen estilo de código fomenta programas que no sólo resuelven el problema, sino que también reflejan claramente la relación problema/solución. Esto tiene como efecto que muchos cambios simples en el problema reflejen de forma obvia los cambios a hacer en el programa.
2. Verificabilidad: La facilidad con que pueden comprobarse propiedades de un sistema. Si el estilo de código hace obvia la estructura del programa, eso ayuda a verificar que el comportamiento sea el esperado.
3. Reparabilidad: La posibilidad de corregir errores sin demasiado esfuerzo.
4. Capacidad de evolución: La capacidad de adaptarse a nuevas necesidades.
5. Comprensibilidad: La facilidad con que el programa puede ser comprendido.

Estilo de codificación empleado

- Todas las etiquetas php deben ser completas (<?php ?>)... no reducidas (<? ?>).

- Los bloques de código siempre deben estar encerrados por llaves (incluso si solo constan de una línea).
- Todas las funciones y clases deben estar comentariadas. Los comentarios deben ser añadidos de forma que resulten prácticos, para explicar el flujo del código y el propósito de las funciones o variables.
- Indentación: tamaño = 4 (espacios) para:
 - Declaraciones dentro de las clases.
 - Enunciado dentro de métodos y funciones.
 - Enunciados dentro de bloques de comandos.
 - Enunciados dentro de cuerpos switch /case.
- Espacios en blanco:
 - Antes de abrir paréntesis.
 - Luego de habierto el paréntesis.
 - Antes de abrir corchetes.
 - Luego de la coma en declaraciones.
- Líneas en blanco:
 - Antes de la declaración de una constante.
 - Luego de la declaración de un campo al comienzo del cuerpo de una declaración/método.
- Nueva línea:
 - En enunciados vacíos.
- Líneas de código:
 - Máximo tamaño de línea (ancho) 300 caracteres.

4.2.2. Ejemplo de código fuente

La mayoría de los métodos implementados, responden a funciones básicas de inserción, actualización, eliminación o búsqueda, que se vuelven realmente sencillos mediante el uso que hace Symfony de Propel para realizar el mapeo objeto-relacional. Este genera la mayor parte del código de acceso a datos proporcionando una abstracción al punto que permite que los implementadores tengan que utilizar muy pocas consultas a la base de datos.

A diferencia de estos, el método **executeAvisos** se encarga de mostrar un mensaje cuando el usuario inicia sesión, este mensaje solo se mostrará si existen avisos pendientes que el usuario necesita conocer. Para esto el método verifica si en la base de datos el campo *fecha_antelacion* de la tabla **t_avisos** coincide con la fecha actual.

```
public function executeAviso()
{
    $fecha_hoy = date ( 'Y/m/d' );
    $c = new Criteria ( );
    if ($fecha_hoy != '')
    {
        $c->add ( TAVisoPeer::FECHA_ATELACION, $fecha_hoy );
    }
    $acciones = TAVisoPeer::doSelect ( $c );
    $cont = 0;
    for($i = 0; $i < count ( $acciones ); $i ++)
    {
        $estado = $acciones [$i]->getEstado ();
        if ($estado != "Atendido")
        {
            $cont ++;
        }
    }
    echo ($cont);
    exit ();
}
```

Figura 28 Ejemplo de código fuente. Método **executeAviso**.

4.2.3. Pantallas principales de la aplicación

El diseño de la interfaz de la aplicación se encaminó a lograr interfaces agradables, sencillas y atractivas al cliente. Para ello mediante la utilización del framework de presentación ExtJS se lograron interfaces que recrean el entorno de trabajo de Windows, lo que posibilita que los usuarios se familiaricen con el uso de la aplicación rápidamente.

El empleo de menús y accesos directos permite que el acceso a las diferentes funcionalidades que brinda el software sea más rápido.

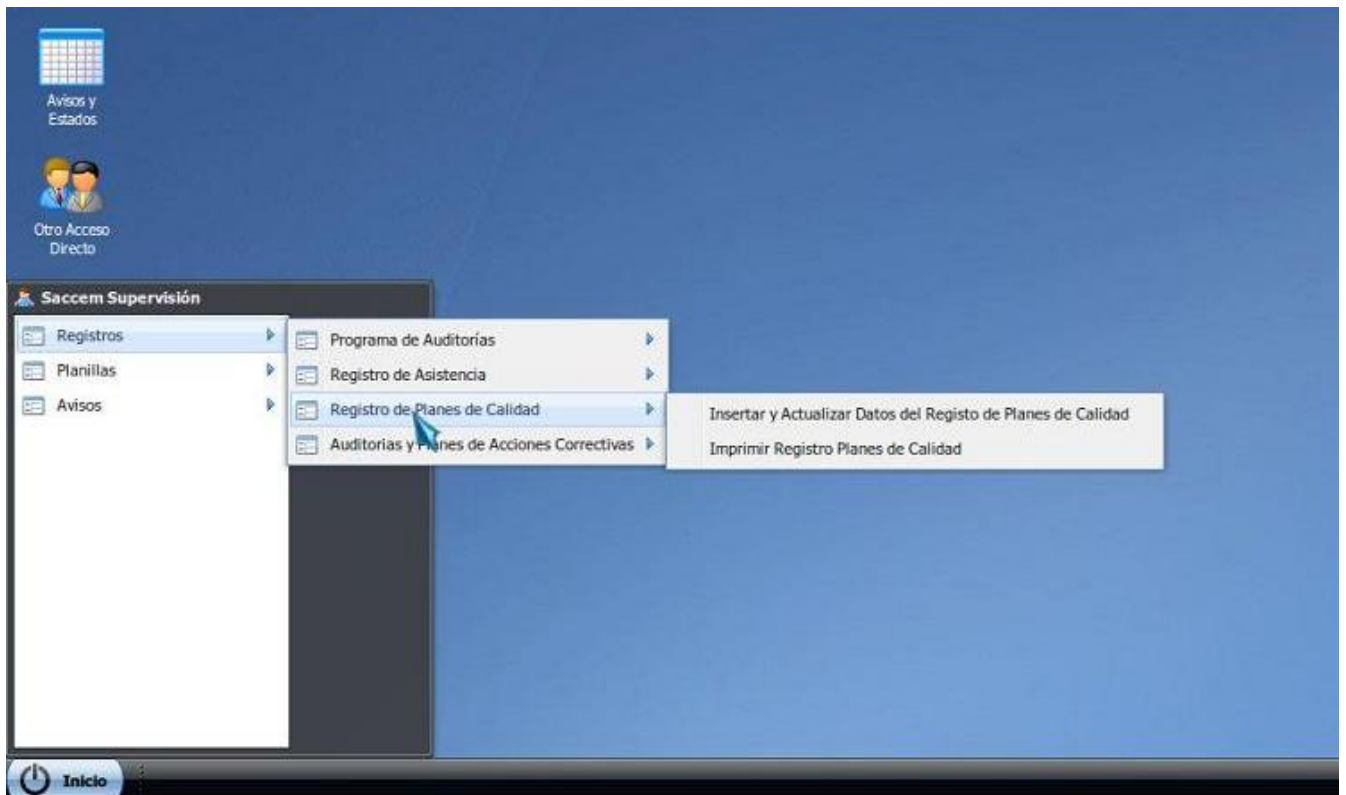


Figura 29 Interfaz de la aplicación. Muestra el menú de inicio y los accesos directos.

Para realizar las búsquedas se implementaron filtros que posibilitan ver los resultados de la búsqueda instantáneamente, mejorando la experiencia del usuario ya que pueden ver si los resultados satisfacen sus necesidades antes de terminar la búsqueda.

Si los elementos que se deben mostrar en pantalla son muchos, se puede entorpecer su visualización o hacerla más incómoda. El paginado permite organizar estos resultados en diferentes páginas haciendo más fácil la búsqueda y visualización de los mismos. Estas ventajas las proporcionan funciones del propio framework ExtJS.

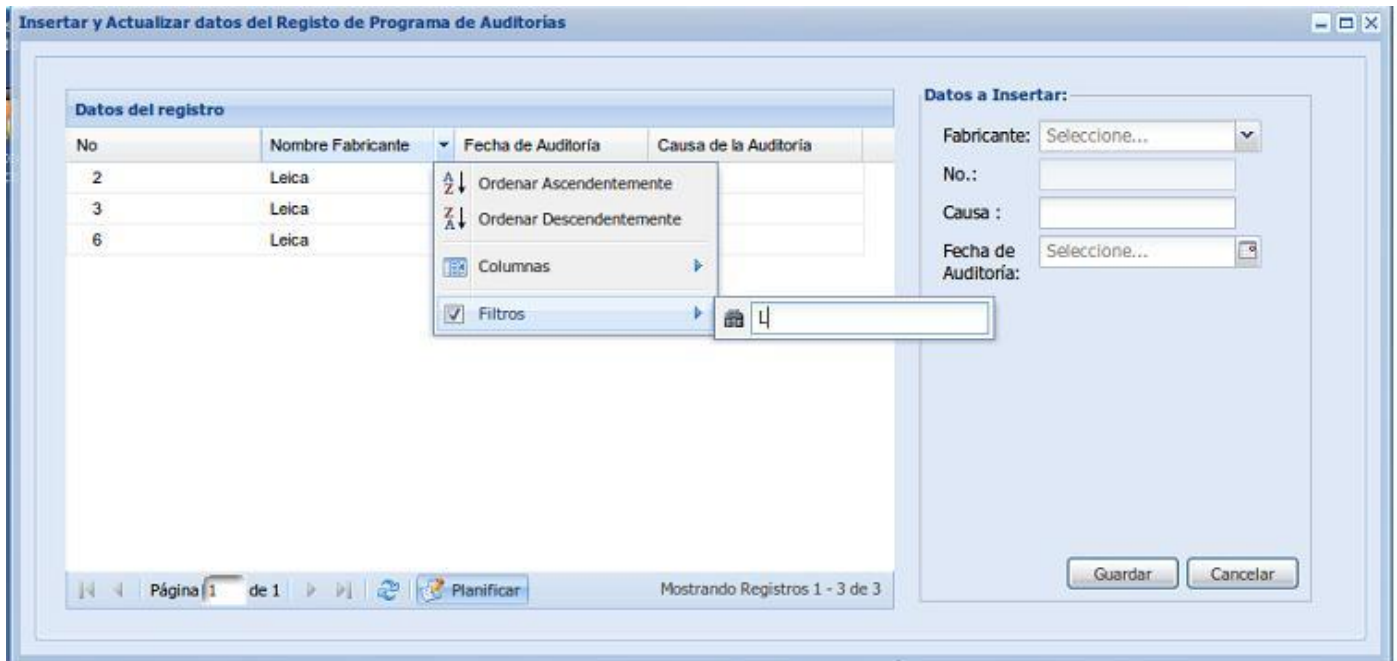


Figura 30 Interfaz de la aplicación. Muestra el paginado y los filtros.

4.3. Validación a nivel de desarrollador

Una actividad importante dentro del flujo de trabajo de implementación es implementar las pruebas a nivel de desarrollador. Esta actividad se realiza con el objetivo de crear un conjunto de pruebas para comprobar que el sistema funciona correctamente antes de que se realicen más pruebas formales en él.

Para esta actividad se toman como entrada los elementos de implementación que abarcan los componentes físicos que forman una implementación, que incluyen archivos y directorios. Incluyen los archivos de código de software (origen, binario o ejecutable), los archivos de datos y los archivos de documentación, como los archivos de ayuda en línea. Como resultado se obtiene la implementación de un subconjunto de pruebas necesarias de forma efectiva y eficaz. (31)

Las pruebas a nivel de desarrollador incluyen las pruebas de unidad y parte de las pruebas de funcionalidad. Las pruebas unitarias aseguran que un único componente de la aplicación produce una salida correcta para una determinada entrada. Este tipo de pruebas validan la forma en la que las funciones y métodos trabajan en cada caso particular. Las pruebas funcionales no solo validan la transformación de una entrada en una salida, sino que validan una característica completa.

Un paso importante es determinar la técnica adecuada para implementar las pruebas. Existen varias técnicas disponibles para implementar una prueba, pero se pueden considerar dos categorías generales: prueba manual y automatizada. (31) La mayoría de pruebas de desarrollador se implementan utilizando las técnicas de prueba automatizada. El framework Symfony cuenta con funcionalidades que hacen la automatización de pruebas una tarea realmente sencilla. Los conjuntos de casos de prueba garantizan que la aplicación hace lo que se supone que debe hacer. Incluso cuando el código interno de la aplicación cambia constantemente, las pruebas automatizadas permiten garantizar que los cambios no introducen incompatibilidades en el funcionamiento de la aplicación.

Las pruebas unitarias de Symfony son archivos PHP cuyo nombre termina en Test.php. Symfony incluye su propio framework llamado Lime para crear pruebas unitarias. Cada prueba unitaria consiste en una llamada a un método de la instancia de **lime_test**. El último parámetro de estos métodos siempre es una cadena de texto opcional que se utiliza como resultado del método.

```

<?php
require_once dirname ( __FILE__ ) . '/../bootstrap/unit.php';
new sfDatabaseManager ( ProjectConfiguration::getApplicationConfiguration ( 'Supervision', 'home', true ) );

function PorCiento($a, $b, $parte)[]
function Update() //Fabricante certificado[]
function Insertar() // Fabricante no certificado[]
function GetGrid()[]

$t = new lime_test ( 4, new lime_output_color ( ) );

$t->is ( PorCiento ( 50, 50, 50 ), '50', 'PorCiento retorna "50" );
$t->is ( Update (), 'Se ha modificado correctamente la lista de chequeo de fabricante certificado', 'Update()
$t->is ( Insertar(), 'Lista de chequeo para fabricante certificado se ha insertado correctamente', 'Insertar()
$t->is ( GetGrid(), 'true', 'GetGrid() retorna "Lista de chequeo se ha mostrado correctamente" );

```

Figura 31 Ejemplo de código fuente que implementa una prueba unitaria.

Las pruebas funcionales simulan la navegación del usuario, realizan peticiones y comprueban los elementos de la respuesta, tal y como lo realizaría manualmente un usuario para validar que una determinada acción hace lo que se supone que tiene que hacer. En las pruebas funcionales, se ejecuta un escenario correspondiente a un caso de uso. Symfony dispone de un objeto especial, llamado **sfBrowser**, que actúa como un navegador que está accediendo a una aplicación Symfony. El framework Symfony también dispone de una extensión de esta clase llamada **sfTestBrowser**, que está especialmente diseñada para las pruebas funcionales y que tiene todas las características de **sfBrowser**.

```

include(dirname(__FILE__).'../../bootstrap/functional.php');

$browsers = new sfTestFunctional(new sfBrowser());

$browsers->info('1 - Buscar Visualizar')->
  get('/Avisos/buscarVisualizar')->

  with('request')->begin()->
    isParameter('module', 'Avisos')->
    isParameter('action', 'buscarVisualizar')->
    |
  end()->

  with('response')->begin()->
    isStatusCode(200)->

  end()
;

```

Figura 32 Ejemplo de código fuente que implementa una prueba funcional.

4.3.1. Resultados de pruebas exploratorias

Las pruebas automatizadas comparan un resultado con la salida esperada para ese método. El valor de salida es verdadero (**true**) o falso (**false**), lo que determina si la prueba tiene éxito o si falla.

Para ejecutar el conjunto de pruebas, se utiliza la tarea **test:unit** desde la línea de comandos. El resultado de esta tarea en la línea de comandos es muy explícito, lo que permite localizar fácilmente las pruebas que han fallado y las que se han ejecutado correctamente.

```

gdem@project-fac6:/var/www/Supervision$ symfony test:unit RPA
1..1
ok 1 - PlanificarNotificacion() retorna "Se planificó la notificación correctamente"
not ok 2 - Delete() retorna "Se eliminó correctamente"
#   Failed test (./test/unit/RPATest.php at line 49)
#       got: 'Ocurrió un error al ejecutar la acción'
#       expected: 'Se eliminó correctamente'
Looks like you planned 1 tests but ran 1 extra.
Looks like you failed 1 tests of 2.

```

Figura 33 Resultado de una prueba de unidad que indica error.

```

gdem@project-fac6:/var/www/Supervision$ symfony test:unit RPA
1..2
ok 1 - PlanificarNotificacion() returns "Se planificó la notificación correctamente"
ok 2 - Update() retorna "Se insertó correctamente"
Looks like everything went fine.

```

Figura 34 Resultado de una prueba de unidad que indica éxito.

Para ejecutar una prueba funcional, se utiliza la tarea `test:functional` de la línea de comandos de Symfony. Los argumentos que se indican a la tarea son el nombre de la aplicación y el nombre de la prueba.

```

gdem@project-fac6:/var/www/Supervision$ symfony test:functional Supervision RAPACActions
# get /RAPAC/imprimirRegistro
ok 1 - request parameter module is RAPAC
ok 2 - request parameter action is imprimirRegistro
ok 3 - status code is 200
# post /RAPAC/update
ok 4 - request parameter module is RAPAC
ok 5 - request parameter action is update
ok 6 - status code is 200
# get /RAPAC/planificarNotificacion
ok 7 - request parameter module is RAPAC
ok 8 - request parameter action is planificarNotificacion
not ok 9 - status code is 200
# Failed test (/usr/share/php/symfony/test/sfTesterResponse.class.php at line 257)
# got: 500
# expected: 200
not ok 10 - last request threw an uncaught exception PropelException: Unable to execute INSERT statement. [wrapped: SQLSTATE[23000]: Integrity constraint violation: 1452 Cannot add or update a child row: a foreign key constraint fails (`sacem/t_avisos`, CONSTRAINT `t_avisos_ibfk_1` FOREIGN KEY (`id_fabricante`) REFERENCES `t_fabricante` (`id_fabricante`) ON DELETE NO ACTION ON UPDATE NO ACTION)]
# Failed test (/usr/share/php/symfony/test/sfTestFunctionalBase.class.php at line 526)
1..10
Looks like you failed 2 tests of 10.

```

Figura 35 Resultado de una prueba funcional que indica errores.

```

gdem@project-fac6:/var/www/Supervision$ symfony test:functional Supervision RPAActions
> 1 - update
# post /RPA/update
ok 1 - request parameter module is RPA
ok 2 - request parameter action is update
> Se realiza la actualizacion
ok 3 - status code is 200
ok 4 - response selector body does not match regex /This is a temporary page/
1..4
Looks like everything went fine.

```

Figura 36 Resultado de una prueba funcional que indica éxito.

El comando **test:all** ejecuta todas las pruebas definidas hasta el momento, y muestra los resultados de todas, incluyendo un resumen que indica por ciento que representan los resultados correctos del total de resultados obtenidos.

```

jdem@project-fac6:/var/www/Supervision$ symfony test:all
functional/Supervision/AvisosActionsTest.....ok
functional/Supervision/IAActionsTest.....ok
functional/Supervision/LCCFActionsTest.....ok
functional/Supervision/PAActionsTest.....ok
functional/Supervision/PrincipalActionsTest.....ok
functional/Supervision/RAActionsTest.....ok
functional/Supervision/RAPACActionsTest.....not ok
  Failed tests: 9, 10
functional/Supervision/RPAActionsTest.....not ok
  Failed tests: 3, 5, 11, 12
functional/Supervision/RPCActionsTest.....not ok
  Failed tests: 9, 10
jnit/AvisoTest.....not ok
  Failed tests: 1
jnit/IATest.....ok
jnit/LCCFTest.....ok
jnit/RATest.....ok
jnit/RPATEST.....ok
Failed Test          Stat  Total  Fail  List of Failed
-----
functional/Supervision/RAPACActionsTest  0     2    2  9 10
functional/Supervision/RPAActionsTest    0     4    4  3 5 11 12
functional/Supervision/RPCActionsTest    0     2    2  9 10
jnit/AvisoTest                          0     1    1  1
Failed 4/14 test scripts, 71.43% okay. 9/146 subtests failed, 93.84% okay.
    
```

Figura 37 Resultado de ejecutar el comando test:all que muestra el resultado de todas las pruebas definidas.

La ejecución de las prueba puede finalizar de dos formas: **normal**: todos los procedimientos de prueba (o scripts) se ejecutaron según lo previsto; **anormal o prematura**: los procedimientos de prueba, o los scripts, no se han ejecutado completamente o según lo previsto. Cuando las pruebas terminan anormalmente, los resultados de las pruebas son poco fiables. La siguiente tabla muestra un resumen de los resultados que se obtuvieron de la ejecución de las pruebas.

Cantidad de pruebas realizadas	Cantidad de pruebas normales	Cantidad de pruebas fallidas	Cantidad de pruebas exitosas	Por ciento que representan
146	137	9	128	93.43%

Tabla 20 Resumen de los resultados de las pruebas a nivel de desarrollador.

Cuando finaliza la prueba, se revizan los resultados para garantizar que son fiables y que las anomalías, los avisos y los resultados inesperados notificados no han sido provocados por influencias externas como, por ejemplo, configuraciones o datos incorrectos. Si estos resultados indican que las anomalías se deben propiamente a errores en el código se procede a realizar cambios y posteriormente se vuelven a ejecutar las pruebas. De esta forma los desarrolladores conocen qué errores presenta el código y la aplicación en general antes de que se realicen pruebas más formales en el software.

4.4. Conclusiones

- ✓ En este capítulo se estructuró el Modelo de Implementación a partir de los resultados del diseño, y se obtuvieron los diferentes diagramas de componentes vinculados a cada caso de uso.
- ✓ Gracias al uso de un estándar de codificación se logró que el código fuera extensible, verificable, reparable y comprensible.
- ✓ Mediante la validación a nivel de desarrollador se validó que el sistema cumpliera con las funcionalidades definidas.

CONCLUSIONES

- ✓ A partir del estudio realizado a los procesos que se llevan a cabo en el Departamento de Supervisión del CCEEM y de una adecuada captura de requisitos, se definieron las funcionalidades del módulo de “Gestión de la información del Departamento de Supervisión del CCEEM versión 1.0.” El empleo de técnicas de captura y validación de requisitos, así como la aplicación de patrones de casos de uso, facilitó el trabajo con los mismos y ayudó a establecer un acuerdo entre desarrolladores y clientes sobre las funcionalidades del sistema.
- ✓ El empleo de patrones de diseño facilitó la obtención de las clases del diseño y una mejor estructuración de los diagramas. En general, la realización del Modelo de Diseño, incluyendo el diseño de la base de datos, facilitó el desarrollo del módulo de “Gestión de la información del Departamento de Supervisión del CCEEM versión 1.0.” al ser este la primera entrada para las actividades de implementación.
- ✓ Se implementaron los componentes definidos y se obtuvo la primera versión de una aplicación informática que permite la gestión de la información del Departamento de Supervisión del CCEEM.

RECOMENDACIONES

Se recomienda:

- ✓ Realizar las pruebas de liberación necesarias para validar la calidad del sistema.
- ✓ Integrar la aplicación al Sistema Automatizado Cubano para el Control de Equipos Médicos actualmente en desarrollo.

REFERENCIAS BIBLIOGRÁFICAS

1. **INFOMED.** Aspectos Generales del Sistema Nacional de Salud. *Infomed. Portal de Salud de Cuba.* [En línea] 2000. [Citado el: 10 de 11 de 2008.] http://www.sld.cu/sistema_de_salud/aspectos.html.
2. **126.a Sesión del Comité Ejecutivo, Organización Panamericana de la Salud.** *Equipos y dispositivos médicos.* Washington, DC. : s.n., 2000.
3. **Stair, Ralph M. y Reynolds, George W.** *Principios de sistemas de información.* http://www.google.com.cu/books?id=k_sKKIF0.
4. **Contreras, Yimian de Lyz y Rivero, Soleydi.** Diseño del Sistema de Gestión de información del Centro de Estudios de Medio Ambiente y Recursos Naturales(CEMARNA)de la Universidad de Pinar del Rio. *GestioPolis.com.* [En línea] 4 de 10 de 2007. [Citado el: 20 de 12 de 2008.] <http://www.gestiopolis.com/administracion-estrategia/sistemas-de-gestion-de-informacion-en-estudio-de-medio-ambiente.htm>.
5. *Gestión documental, de información y del conocimiento... puntos de contacto y diferencias.* **Ponjuán, Gloria.** 3, 12 de 2003, Ciencias de la Información, Vol. 36. <http://biblioteca.uci.cu>. 0864-4659.
6. *Los Sistemas Informáticos: conceptos básicos y funcionamiento general.* **Universidad de Andalucía.** Sistemas informáticos.
7. *What is Content Management?* **Gilbane, Frank.** 8, Octubre de 2000, The Gilban Group, Vol. 8.
8. Características de Interlan System. *Interlan System.* [En línea] 2009. [Citado el: 1 de 2 de 2009.] <http://www.inter-lan.com/caracteristicas.html> 14638..
9. **López, Alain y Rivas, Ronni.** *Aplicación para la Gestión de la información del Puesto de Mando Informático de los Centros de Diagnóstico Integrales en Venezuela.* La Habana : s.n., 2008. págs. 8-11, Trabajo de Diploma.
10. **Pérez, Renier.** Impacto de la Informatización en la Sociedad Cubana. Ciencia, tecnología y sociedad. *Monografías.com.* [En línea] 2005. [Citado el: 11 de 1 de 2009.] <http://www.monografias.com/trabajos24/informatizacion-cuba/informatizacion-cuba.shtml>.

11. **Gutiérrez, Octavio.** Fundamentos de la ciencia y la innovación tecnológica. . *Monografias.com*. [En línea] [Citado el: 10 de 1 de 2009.] [http://www.monografias.com/trabajos38/ciencia-innovacion-tecnologica/ciencia-innovacion-tecnologica.shtml#sistema 15549..](http://www.monografias.com/trabajos38/ciencia-innovacion-tecnologica/ciencia-innovacion-tecnologica.shtml#sistema%2015549)
12. **Borges, Zenaida López y Machado, Kisimira Díaz.** El Liderazgo Ético como una alternativa necesaria en la cultura organizacional cubana. [En línea] [Citado el: 10 de 1 de 2009.] III Conferencia Internacional, La obra de Carlos Marx y los desafíos del siglo XXI, Universidad de Cienfuegos “Carlos Rafael Rodríguez”. http://www.nodo50.org/cubasigloXXI/congreso06/conf3_lborges.pdf.
13. **IDICT.** La Política Nacional de Información en Cuba. [En línea] Abril de 2001. [Citado el: 10 de 1 de 2009.] <http://www.procig.org/material-inde/policy-other/politica-nacional-informacion-cuba.pdf>.
14. **Ministerio del Interior.** *Decreto Ley del MININT sobre la seguridad y protección de la información oficial*. Enero de 1999.
15. Arquitectura y Patrones de diseño. [En línea] 2009. [Citado el: 3 de 3 de 2009.] teleformacion.uci.cu.
16. **Jacobson, Ivar, Booch, Grady y Rumbaugh, James.** *El Proceso Unificado de Desarrollo de Software*. España : Adison-Wesley.
17. **IEEE.** *Estándar IEEE*. Enero de 1998.
18. **Gang of Four.** *Design Patterns: Elements of Reusable Object-Oriented Software*. 1995. (Gamma, Erich; Helm, Richard; Johnson, Ralph; Vlissides, John).
19. **Larman, Craig.** *UML y Patrones*. México : Prentice Hall, 1999. pág. 536.
20. **Object Management Group** . UML. [En línea] 2009. [Citado el: 15 de 3 de 2009.] [http://www.uml.org/ 13304..](http://www.uml.org/)
21. **Visual Paradigm International.** New Releases and Feature Enhancements of Visual Paragim Products. *Visual Paradigm*. [En línea] Abril de 2009. [Citado el: 20 de 4 de 2009.] <http://www.visual-paradigm.com/product/news.jsp#VPUML50>.

22. **fabFORCE.net**. *The DBDesigner 4 Successor*. [En línea] 2003-2009. [Citado el: 12 de 2 de 2009.] <http://www.fabforce.net/dbdesigner4/>.
23. **Alvarez, Miguel Angel**. Zend Studio. *desarrolloweb.com*. [En línea] 4 de 6 de 2003. [Citado el: 10 de 1 de 2009.] <http://www.desarrolloweb.com/articulos/1178.php>.
24. The Definitive JavaScript Resource: JavaScript Tutorials, Free Java Scripts, Source Code and Other Scripting Resources. . *JavaScript.com* . [En línea] 2009. [Citado el: 3 de 4 de 2009.] <http://www.javascript.com/> .
25. **The PHP Group**. News Archive - 2007. *PHP*. [En línea] 2 de 4 de 2009. <http://www.php.net/archive/2007.php>.
26. **Ext Management Team**. Ext JS- A foundation you can build on. *Introducing Ext JS 2.0*. [En línea] 2006-2009. [Citado el: 10 de 2 de 2009.] <http://www.extjs.com/>.
27. Intrepidlbex. *Ubuntu*. [En línea] 2009. [Citado el: 20 de 2 de 2009.] <https://wiki.ubuntu.com/Intrepidlbex>.
28. **IEEE**. *IEEE Standard Glossary of Software Engineering Terminology*.
29. **Pan, D, Zhu, D y Johnson, K**. *Requierements Engeenering Thecniques*. Canadá : s.n., 2001. Departamento de Ciencias de la Computación.Universidad de Calgary..
30. **Raghavan, Sridhar, Zelesnik, Gregory y Ford, Gary**. *Lecture Notes on Requirements Elicitation*. CARNEGIE-MELLON UNIV PITTSBURGH PA SOFTWARE ENGINEERING INST : Defense Technical Information Center, 1994.
31. **Rational Unified Process**. *Extended Help for RUP*. 2003.

BIBLIOGRAFÍA

1. **INFOMED.** Aspectos Generales del Sistema Nacional de Salud. *Infomed. Portal de Salud de Cuba.* [En línea] 2000. [Citado el: 10 de 11 de 2008.] http://www.sld.cu/sistema_de_salud/aspectos.html.
2. **126.a Sesión del Comité Ejecutivo, Organización Panamericana de la Salud.** *Equipos y dispositivos médicos.* Washington, DC. : s.n., 2000.
3. **Stair, Ralph M. y Reynolds, George W.** *Principios de sistemas de información.* http://www.google.com/cu/books?id=k_sKKIF0.
4. **Contreras, Yimian de Lyz y Rivero, Soleydi.** Diseño del Sistema de Gestión de información del Centro de Estudios de Medio Ambiente y Recursos Naturales(CEMARNA)de la Universidad de Pinar del Rio. *GestioPolis.com.* [En línea] 4 de 10 de 2007. [Citado el: 20 de 12 de 2008.] <http://www.gestiopolis.com/administracion-estrategia/sistemas-de-gestion-de-informacion-en-estudio-de-medio-ambiente.htm>.
5. *Gestión documental, de información y del conocimiento... puntos de contacto y diferencias.* **Ponjuán, Gloria.** 3, 12 de 2003, Ciencias de la Información, Vol. 36. <http://biblioteca.uci.cu>. 0864-4659.
6. *Los Sistemas Informáticos: conceptos básicos y funcionamiento general.* **Universidad de Andalucía.** Sistemas informáticos.
7. *What is Content Management?* **Gilbane, Frank.** 8, Octubre de 2000, The Gilban Group, Vol. 8.
8. Características de Interlan System. *Interlan System.* [En línea] 2009. [Citado el: 1 de 2 de 2009.] <http://www.inter-lan.com/caracteristicas.html> 14638..
9. **López, Alain y Rivas, Ronni.** *Aplicación para la Gestión de la información del Puesto de Mando Informático de los Centros de Diagnóstico Integrales en Venezuela.* La Habana : s.n., 2008. págs. 8-11, Trabajo de Diploma.
10. **Pérez, Renier.** Impacto de la Informatización en la Sociedad Cubana. Ciencia, tecnología y sociedad. *Monografias.com.* [En línea] 2005. [Citado el: 11 de 1 de 2009.] <http://www.monografias.com/trabajos24/informatizacion-cuba/informatizacion-cuba.shtml>.
11. **Gutiérrez, Octavio.** Fundamentos de la ciencia y la innovación tecnológica. . *Monografias.com.* [En línea] [Citado el: 10 de 1 de 2009.] <http://www.monografias.com/trabajos38/ciencia-innovacion-tecnologica/ciencia-innovacion-tecnologica.shtml#sistema> 15549..
12. **Borges, Zenaida López y Machado, Kisimira Díaz.** El Liderazgo Ético como una alternativa necesaria en la cultura organizacional cubana. [En línea] [Citado el: 10 de 1 de 2009.] III Conferencia Internacional, La obra de Carlos Marx y los desafíos del siglo XXI, Universidad de Cienfuegos “Carlos Rafael Rodríguez”. http://www.nodo50.org/cubasigloXXI/congreso06/conf3_lborges.pdf.

13. **IDICT**. La Política Nacional de Información en Cuba. [En línea] Abril de 2001. [Citado el: 10 de 1 de 2009.] <http://www.procig.org/material-inde/policy-other/politica-nacional-informacion-cuba.pdf>.
14. **Ministerio del Interior**. *Decreto Ley del MININT sobre la seguridad y protección de la información oficial*. Enero de 1999.
15. Arquitectura y Patrones de diseño. [En línea] 2009. [Citado el: 3 de 3 de 2009.] teleformacion.uci.cu.
16. **Jacobson, Ivar, Booch, Grady y Rumbaugh, James**. *El Proceso Unificado de Desarrollo de Software*. España : Adison-Wesley.
17. **IEEE**. *Estándar IEEE*. Enero de 1998.
18. **Gang of Four**. *Design Patterns: Elements of Reusable Object-Oriented Software*. 1995. (Gamma, Erich; Helm, Richard; Johnson, Ralph; Vlissides, John).
19. **Larman, Craig**. *UML y Patrones*. México : Prentice Hall, 1999. pág. 536.
20. **Object Management Group** . UML. [En línea] 2009. [Citado el: 15 de 3 de 2009.] <http://www.uml.org/13304..>
21. **Visual Paradigm International**. New Releases and Feature Enhancements of Visual Paragim Products. *Visual Paradigm*. [En línea] Abril de 2009. [Citado el: 20 de 4 de 2009.] <http://www.visual-paradigm.com/product/news.jsp#VPUML50>.
22. **fabFORCE.net**. *The DBDesigner 4 Successor*. [En línea] 2003-2009. [Citado el: 12 de 2 de 2009.] <http://www.fabforce.net/dbdesigner4/>.
23. **Alvarez, Miguel Angel**. Zend Studio. *desarrolloweb.com*. [En línea] 4 de 6 de 2003. [Citado el: 10 de 1 de 2009.] <http://www.desarrolloweb.com/articulos/1178.php>.
24. The Definitive JavaScript Resource: JavaScript Tutorials, Free Java Scripts, Source Code and Other Scripting Resources. . *JavaScript.com* . [En línea] 2009. [Citado el: 3 de 4 de 2009.] <http://www.javascript.com/> .
25. **The PHP Group**. News Archive - 2007. *PHP*. [En línea] 2 de 4 de 2009. <http://www.php.net/archive/2007.php>.
26. **Ext Management Team**. Ext JS- A foundation you can build on. *Introducing Ext JS 2.0*. [En línea] 2006-2009. [Citado el: 10 de 2 de 2009.] <http://www.extjs.com/>.
27. Intrepidlbex. *Ubuntu*. [En línea] 2009. [Citado el: 20 de 2 de 2009.] <https://wiki.ubuntu.com/Intrepidlbex>.
28. **IEEE**. *IEEE Standard Glossary of Software Engineering Terminology*.
29. **Pan, D, Zhu, D y Johnson, K**. *Requierements Engeeneering Thecniques*. Canadá : s.n., 2001. Departamento de Ciencias de la Computación.Universidad de Calgary..

30. **Raghavan, Sridhar, Zelesnik, Gregory y Ford, Gary.** *Lecture Notes on Requirements Elicitation.* CARNEGIE-MELLON UNIV PITTSBURGH PA SOFTWARE ENGINEERING INST : Defense Technical Information Center, 1994.
31. **Rational Unified Process.** *Extended Help for RUP.* 2003.
32. **Remedios, Midel A.** Breve historia de la informatización de la sociedad cubana. *Monografias.com.* [En línea] 2007. [Citado el: 12 de 12 de 2008.] <http://www.monografias.com/trabajos41/historia-informatica-cuba/historia-informatica-cuba2.shtml>.
33. **Dueñas, Lourdes M.** Caracterización de un Sistema de Gestión de Información Científico Tecnológica con enfoque a procesos: garantía para la mejora continua. [En línea] Abril de 2004. [Citado el: 20 de 1 de 2009.] Palacio de las Convecciones, La Habana. <http://www.bibliociencias.cu/gsdll/collect/eventos/index/assoc/HASH5042.dir/doc.pdf>.
34. **Datahouse Company.** SOFTWARE ERP, SOFTWARE DE GESTION COMERCIAL Y SOLUCIONES PARA EMPRESAS - FACTURA ELECTRONICA. [En línea] 2009. [Citado el: 1 de 01 de 2009.] <http://www.datahousecompany.com.ar/>.
35. **del Valle, Félix.** Documentos digitales: hipertexto y edición digital. *Universidad Complutense Madrid.* [En línea] 2006. [Citado el: 10 de 1 de 2009.] <http://www.ucm.es/info/multidoc/prof/fvalle/tema20.htm>.
36. **Taramullas, Jesús.** Herramientas de software libre para la gestión de contenidos. *hipertext.net.* [En línea] 5 de 2005. [Citado el: 10 de 1 de 2009.] <http://www.hipertext.net/web/pag258.htm>.
37. **Instituto de Historia de Cuba, Colectivo de Autores.** Historia de Cuba. *cubagob.* [En línea] [Citado el: 20 de 11 de 2008.] http://www.cubagob.gov.cu/otras_info/historia/revolucion.htm.
38. **Visual Paradigm International.** Increase productivity, communication, and collaboration using UML visual modeling platform. *Visual Paradigm.* [En línea] 2009. [Citado el: 10 de 2 de 2009.] <http://www.visual-paradigm.com/>.
39. **Pressman, Roger.** *Ingeniería del Software: Un enfoque práctico.* Quinta. Madrid : Mac Graw Hill, 2002.
40. **WebMediaBrands.** The Definitive JavaScript Resource: JavaScript Tutorials, Free Java Scripts, Source Code and Other Scripting Resources. *JavaScript.com* . [En línea] 2009. [Citado el: 20 de 2 de 2009.] <http://www.javascript.com/>.
41. **Castro, Fidel.** *La Batalla de Ideas ha significado mucho para todas las familias cubanas.* 5 de 12 de 2004.
42. **Ballenilla, Thais M.** La evaluación de la tecnología de equipos médicos en el contexto regulador cubano. [En línea] 25 de 5 de 2001. [Citado el: 1 de 12 de 2008.] <http://www.memsoctbio.sld.cu/arrepdf/00152.pdf>.

43. **Martínez, Dulce Ma., Castellanos, Vivian y Ríos, Miriam.** Papel de las normas en el Programa Regualdor cubano de Equipos Médicos. [En línea] 26 de Enero de 2005. <http://www.uh.cu/infogral/areasuh/vri/archivos/Calidad/Tallernormal/Normas%20Equipos%20Medicos.pdf>.
44. **García, Julio A.** Programa para la implantación de un sistema de gestión por el conocimiento en la Empresa de Proyectos de Arquitectura e Ingeniería de Matanzas. [En línea] 2004. [Citado el: 12 de 1 de 2009.] Universidad de Matanzas "Camilo Cienfuegos". <http://www.congreso-info.cu/UserFiles/File/Info/Info2006/Ponencias/70.pdf>.
45. **Suárez, E, Durán, I y González, D.** Tendencias del nuevo siglo en las regulaciones de calidad para equipos médicos. *Imbiomed*. [En línea] 2005. [Citado el: 12 de 12 de 2008.] http://www.imbiomed.com.mx/1/1/articulos.php?method=showDetail&id_articulo=32477&id_seccion=1676&id_ejemplar=3340&id_revista=110.
46. **Jacobson, Ivar, Ericcson, Maria y Jacobson, Agneta.** *The object advantage: Buisnes Process Reengineering with Object Technology*. s.l. : Adison-Wesley , 1995.
47. **Potencier, Fabien y Zaninotto, François.** *Symfony 1.2, la guía definitiva*. 2008.

ANEXOS

Anexo 1: Estructura y comportamiento de un sistema informático.

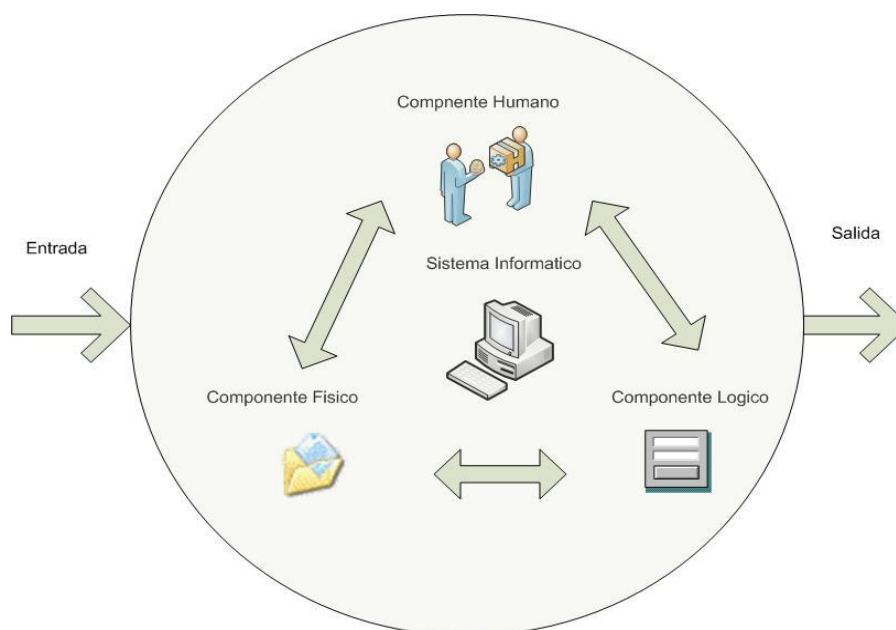


Figura 38 Estructura y comportamiento de un sistema informático.

Anexo 2: Descripciones textuales de casos de uso del negocio.

Nombre del CU	Programar Auditoría
Actores	Fabricante
Trabajadores	Jefe DS(Inicia)
Resumen	El CU se inicia cuando el Jefe DS debe programar una auditoría. Establece el tipo de auditoría y la fecha en que se va a realizar y se la comunica al fabricante. Este envía confirmación y el Jefe DS registra la auditoría en el Registro Programa de Auditorías, finalizando así el CU.
Precondiciones	-
Flujo Normal de los Eventos	
Acciones del Actor	Respuesta del Negocio
	1. Chequea los documentos Registro de Planes de Calidad (RPC), Registro de Auditorías y

	<p>Planes de Acciones Correctivas (RAPAC) y las Solicitudes de Inscripción de Fabricantes recibidas.</p> <p>2. Establece el tipo de auditoría que se va a realizar:</p> <ul style="list-style-type: none"> • Inicial • De Seguimiento • Especial <p>3. Escoge la fecha en la que se va a realizar y envía una notificación al fabricante para su conocimiento y aprobación. Esta puede ser por correo electrónico, correo tradicional, llamada telefónica, o físicamente.</p>
4. Aprueba la fecha de la auditoría y envía una confirmación al Departamento de Supervisión como constancia de que se le ha avisado de la misma.	<p>5. Archiva la confirmación recibida en el Expediente de Fabricante.</p> <p>6. Registra la auditoría en el Registro Programa de Auditorías (RPA) anota el número de la auditoría, la causa de la auditoría, el fabricante al que se le va a realizar y la fecha en que se va a realizar.</p>
Flujo Alternativo de Eventos	
Acciones del Actor	Respuesta del Negocio
4.1 Desaprueba la fecha propuesta para la auditoría.	5.1 Cambia la fecha prevista y propone otra al Fabricante.
Poscondiciones	Se actualiza el Registro Programa de Auditorías y el Expediente de Fabricante.
Posibles mejoras	Se recomienda automatizar la manipulación de datos para agilizar el proceso de actualización de documentos.

Tabla 21 Descripción textual del caso de uso del negocio Programar Auditoría.

Nombre del CU	Realizar Auditoría
Actores	Fabricante (Inicia)
Trabajadores	Auditor Líder, Especialista DS, Jefe DS

Casos de uso asociados	Programar Auditoría(inclusión)
Resumen	El caso de uso se inicia cuando se programa una auditoría, se le informa al fabricante y se procede a realizar la misma. Se actualizan los documentos pertinentes finalizando así el caso de uso.
Precondiciones	-
Flujo Normal de los Eventos	
Acciones del Actor	Respuesta del Negocio
	<ol style="list-style-type: none"> 1. Ir al CUN Programar Auditoría. 2. El Jefe DS define el Equipo Auditor y al Auditor Líder. 3. El Jefe DS define el tipo de Auditoría, objetivos, alcance y criterios. 4. El Jefe DS solicita al fabricante los documentos relevantes para la auditoría. Los mismos son: <ul style="list-style-type: none"> • Manual de Calidad • Lista de Procedimientos • Copia de las Certificaciones del Sistema de Gestión de la Calidad • Copia del Informe de las visitas pre-certificación • Informe de auditorías internas
5. Envía la documentación requerida	<ol style="list-style-type: none"> 6. El Jefe DS determina la viabilidad de la auditoría teniendo en cuenta: <ul style="list-style-type: none"> • la información suficiente y apropiada para planificar la auditoría. • la cooperación adecuada del auditado, y • el tiempo y los recursos adecuados. 7. El Auditor Líder elige la “Matriz para la Lista Personalizada de Auditorías” (MLPA) que le corresponde al fabricante de acuerdo a la clase de riesgo mayor de los equipos médicos que estén comprendidos en el alcance de la auditoría.

	<p>8. El Auditor Líder selecciona las preguntas que conformarán la “Lista personalizada para la Auditoría”(LPA).</p> <p>9. El Auditor Líder elabora el Plan de Auditoría (PA).</p> <p>10. Al mismo tiempo, el Auditor Líder asigna a cada miembro del equipo auditor los elementos del Sistema de Calidad que le corresponderá auditar.</p> <p>11. El Auditor Líder envía al fabricante una copia del PA por correo electrónico, fax, correo ordinario o personalmente.</p>
<p>12. Envía constancia escrita de que ha recibido el PA</p>	<p>13. El Auditor Líder incluye en el Expediente de Fabricante (EF) copia del PA y constancia escrita del fabricante.</p> <p>14. El Auditor Líder efectúa la reunión inicial</p> <p>15. El Auditor Líder llena el Registro de Asistencia (RegA).</p> <p>16. El Especialista DS recopila y verifica información a través de entrevistas, observación, exámenes</p> <p>17. El Especialista DS Registra las evidencias objetivas en el espacio destinado a este aspecto en la LPA</p> <p>18. El Especialista DS evalúa las evidencias de la auditoría contra los criterios de la misma para generar los Hallazgos de la Auditoría</p> <p>19. El Especialista de DS marca en la LPA si el Hallazgo indica conformidad o no conformidad con los criterios de auditoría.</p> <p>20. El Auditor Líder recopile y analiza las No Conformidades encontradas por los Especialistas DS individualmente y ratifica las que proceden.</p> <p>21. El Auditor Líder presenta al fabricante las no</p>

	<p>conformidades encontradas en la auditoría.</p> <p>22. El Auditor Líder establece una fecha para la entrega del Plan de Acciones Correctivas (PAC).</p>
23. Aprueba la fecha de entrega del PAC	<p>24. El Auditor Líder llena el Registro de Asistencia de la reunión de cierre.</p> <p>25. El Especialista DS prepara el Informe de la Auditoría (IA).</p> <p>26. El Jefe DS aprueba el IA.</p> <p>27. El Especialista DS envía al fabricante por correo ordinario, fax o entrega personal, el IA.</p>
28. Envía constancia de que ha recibido el IA	<p>29. El Especialista DS archiva todos los documentos de la auditoría en el Expediente del Fabricante. Los documentos imprescindibles para archivar son:</p> <ul style="list-style-type: none"> • Plan de la Auditoría. • “Lista personalizada para la Auditoría” utilizada en la auditoría. • Informe de la auditoría. • Registro de Asistencia. <p>30. El Jefe DS considera terminada la auditoría cuando conoce que el fabricante tiene en su poder el IA y que los registros de la Auditoría estén actualizados.</p>
31. Envía el PAC	<p>32. El Especialista DS revisa el PAC para comprobar que se han abarcado todas las no conformidades señaladas y que el cumplimiento de las acciones está programado para un plazo razonable.</p> <p>33. El Especialista DS refleja la fecha de entrega del PAC, de la revisión del mismo y las acciones de seguimiento propuestas en el Registro de Auditorías y Planes de Acciones Correctivas</p>

	(RAPAC). 34. El Especialista DS archiva el PAC junto con el resultado de la revisión en el Expediente del Fabricante.
Flujo Alternativo de Eventos	
Acciones del Actor	Respuesta del Negocio
	6.1 El jefe DS ante la inexistencia de alguno de los elementos analizados, declara inviable la auditoría.
	13.1 El Auditor Líder comunica al fabricante si existen cambios en el contenido del Plan de la Auditoría.
Poscondiciones	Se actualiza el Expediente del Fabricante.
Posibles mejoras	Se propone automatizar el proceso de acceso a la información requerida para agilizar todas las actividades relacionadas así como la actualización de la misma.

Tabla 22 Descripción textual del caso de uso del negocio Realizar Auditoría.

Nombre del CU	Atender Demanda
Actores	Demandante(Inicia)
Trabajadores	Especialista DS
Casos de uso asociados	Realizar Auditoría(extensión)
Resumen	El CU se inicia cuando el Demandante se presenta en el DS y presenta una demanda contra un producto de uno de los fabricantes inscritos. El DS valida la demanda y decide el procedimiento a seguir. Envía copia al demandante de los resultados.
Precondiciones	La demanda debe ser realizada contra un fabricante inscrito en el CCEEM.
Flujo Normal de los Eventos	
Acciones del Actor	Respuesta del Negocio
1. Se presenta en el Departamento de Supervisión y plantea la demanda contra algún equipo o producto de	2. Crea el Expediente de la Demanda con la documentación y evidencia presentada por el Demandante.

uno de los fabricantes inscritos en el CCEEM.	<p>3. Determina que la demanda procede de acuerdo a las evidencias presentadas.</p> <p>4. Realiza las investigaciones pertinentes a la documentación del fabricante de acuerdo a la gravedad de la demanda.</p> <p>5. Realiza un informe con los resultados obtenidos y lo envía al Demandante.</p>
6. Recibe respuesta de la demanda y aprueba los resultados.	<p>7. Archiva el informe de los resultados en el Expediente de la Demanda.</p> <p>8. Archiva una copia del informe de los resultados en el Expediente del Fabricante</p>
Flujo Alternativo de Eventos	
Acciones del Actor	Respuesta del Negocio
	3.2. Determina que la queja no procede porque es escasa o irrelevante la evidencia presentada.
	4.1. Realiza una Auditoría al fabricante (Ir CUN Realizar Auditoría).
Poscondiciones	Se actualiza el Expediente del Fabricante y la documentación general del DS.
Posibles mejoras	Se propone automatizar el proceso de acceso a la información requerida para agilizar todas las actividades relacionadas así como la actualización de la misma.

Tabla 23 Descripción textual del caso de uso del negocio Atender Demanda.

Nombre del CU	Aprobar Inscripción/Reinscripción del Fabricante
Actores	Fabricante(inicia)
Trabajadores	Especialista DS
Casos de uso asociados	Realizar Auditoría(extensión)
Resumen	El CU se inicia cuando el fabricante se dirige al DD y pide inscribirse o reinscribirse en el CCEEM. El DD le traspasa la documentación al DS. Este departamento decide si la documentación presentada es suficiente o si se debe realizar una

	auditoría. Al finalizar emite su aprobación al DD de acuerdo a las normas establecidas a través de un informe completo finalizando así el CU.
Precondiciones	-
Flujo Normal de los Eventos	
Acciones del Actor	Respuesta del Negocio
1. Se presenta en el CCEEM y pide su inscripción/reinscripción al Departamento de Documentación. 2. Entrega su documentación al DD. (Planilla de Inscripción, Documentos de Calidad).	3. El DD le pasa la documentación del fabricante al DS y pide su conformidad para continuar con el proceso de inscripción/reinscripción. 4. El DS determina que la documentación presentada por el fabricante es suficiente para realizar una valoración. (Documentos de Calidad) 5. El DS revisa la documentación y comprueba si tienen validez y cumple con las normas de calidad establecidas. 6. Conformar el documento "Lista de Cheque de Calidad para Fabricante" (LCCF), que contiene los resultados de la investigación. 7. Envía el documento LCCF al DD.
8. Recibe el documento y prosigue con el procedimiento de inscripción/reinscripción de fabricantes.	
Flujo Alternativo de Eventos	
Acciones del Actor	Respuesta del Negocio
	4.1 El DS determina que la documentación presentada por el fabricante no es suficiente y determina realizar una auditoría (Ir CUN Realizar Auditoría)
	5.1 El DS revisa la documentación y comprueba que la Certificación de Calidad no es válida o que no cumple con las normas de

	calidad vigentes. 6.1 Informa al DD.
Poscondiciones	Se actualiza toda la documentación empleada.
Posibles mejoras	Se recomienda automatizar el proceso de acceso a la información requerida para agilizar todas las actividades relacionadas así como la actualización de la misma.

Tabla 24 Descripción textual del caso de uso del negocio Aprobar Inscripción/Reinscripción del Fabricante.

Nombre del CU	Brindar Información	
Actores	Auditor Externo(Inicia)	
Trabajadores	Jefe DS	
Resumen	El CU se inicia cuando el actor se presenta en el Departamento de Supervisión y pide realizar una revisión de los documentos del mismo. El Jefe DS entrega los documentos solicitados. Estos son revisados por el Auditor Externo que al terminar de revisarlos se los devuelve al Jefe DS y en caso de encontrarlos menciona sus señalamientos.	
Especialista DS	-	
Flujo Normal de los Eventos		
	Acciones del Actor	Respuesta del Negocio
	1. Se presenta en el centro y solicita la documentación e información perteneciente al DS que desea revisar.	2. Entrega los documentos e información (pueden ser reportes estadísticos) .
	3. Revisa que los documentos se encuentran en orden de acuerdo con las normas y procedimientos establecidos. 4. Entrega los documentos al DS .	5. Archiva los documentos nuevamente.
Poscondiciones	-	
Posibles mejoras	Se recomienda informatizar el proceso de acceso a la información requerida para agilizar todas las actividades relacionadas.	

Tabla 25 Descripción textual del caso de uso del negocio Brindar Información.

Anexo 3: Modelo de objetos del negocio.



Figura 39 Diagrama de clases del Modelo de objetos para el trabajador del negocio Departamento de Documentación.

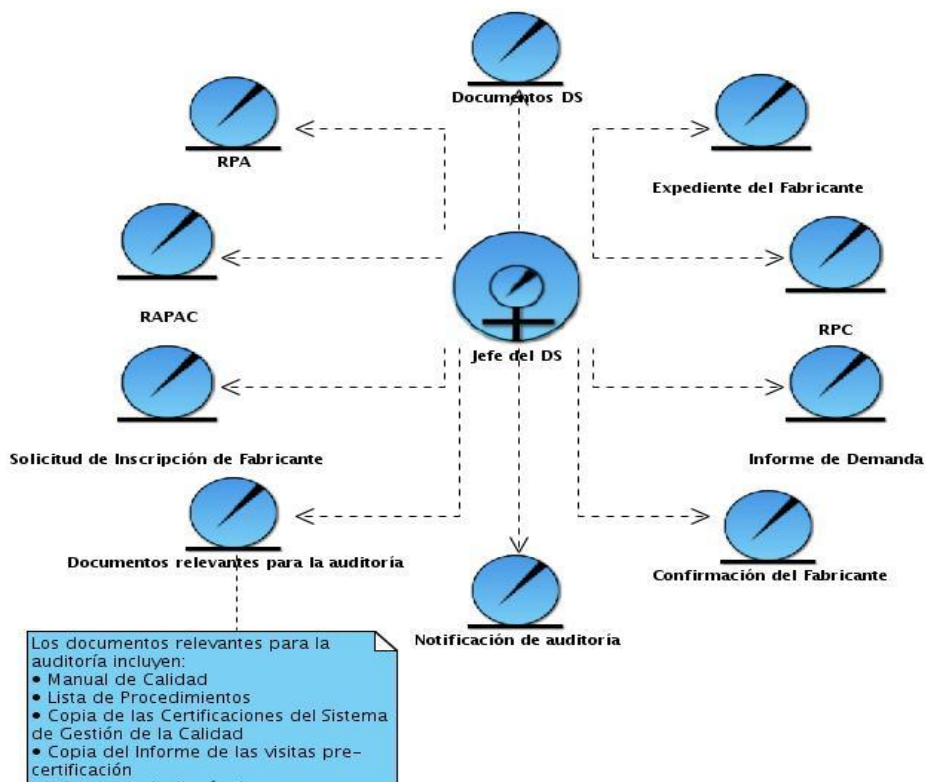


Figura 40 Diagrama de clases del Modelo de objetos para el trabajador del negocio Jefe DS.

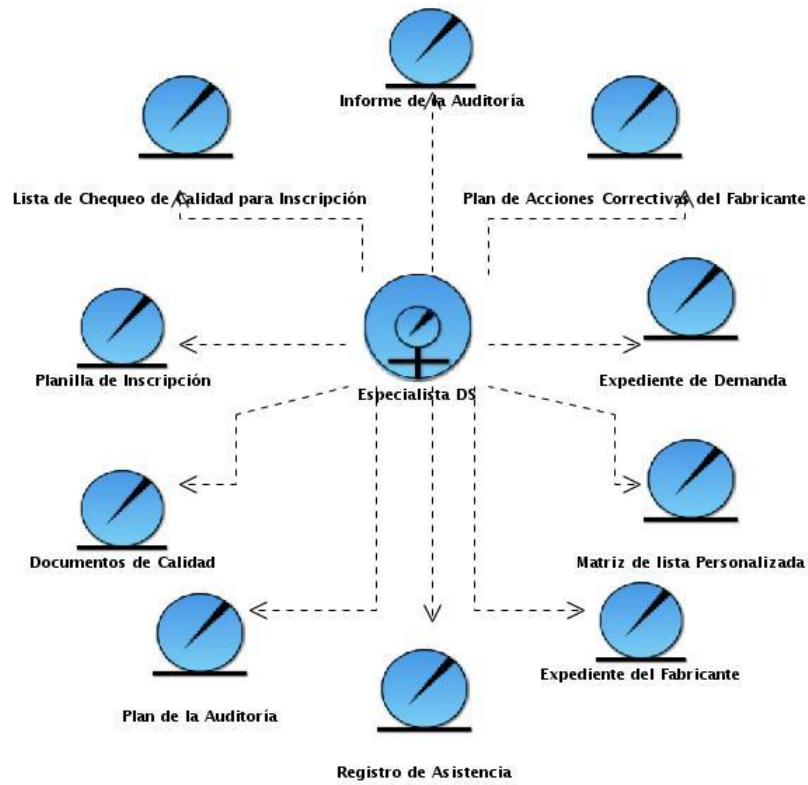


Figura 41 Diagrama de clases del Modelo de objetos para el trabajador del negocio Especialista DS.

Anexo 4: Patrones de casos de uso

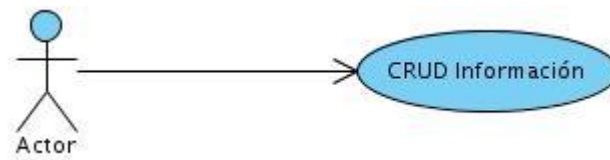


Figura 42 Patrón CRUD

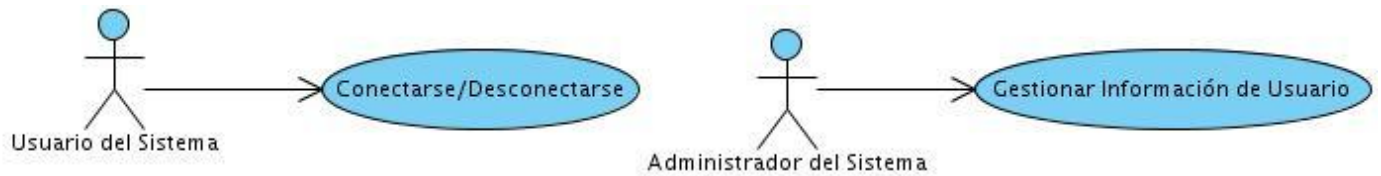


Figura 43 Patrón Login

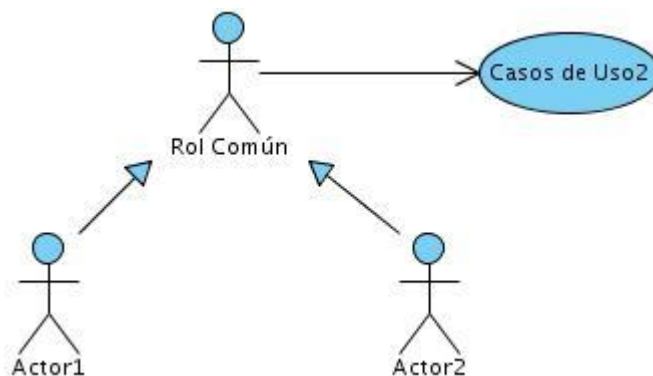


Figura 44 Patrón Múltiples actores

Anexo 5: Descripciones textuales casos de uso del sistema.

Caso de Uso:	Visualizar Expediente de Fabricante
Actores:	Especialista DS(inicia)
Resumen:	El CU se inicia cuando el Especialista DS pide visualizar un expediente. El sistema realiza la búsqueda pertinente y muestra los datos definidos por el Especialista DS finalizando el CU.
Precondiciones:	El Especialista DS debe estar autenticado.
Referencias	RF 1, RF 1.1, RF 1.2, RF 1.3, RF 1.4
Prioridad	Secundario
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
	1.El sistema muestra las opciones por las que se puede realizar la búsqueda: a) Número del Expediente b) Nombre del Fabricante c) Fecha de Inscripción d) País
2.Selecciona las opciones y especifica los criterios de búsqueda.	3.El sistema busca todos los expedientes que coincidan con los criterios especificados.
	4.El sistema muestra un listado con todos los expedientes resultantes de

	la búsqueda, visualizando: número del Expediente, nombre del fabricante y país.
5.El Especialista DS selecciona el expediente cuyos datos desea que sean mostrados.	6.El sistema muestra los datos del expediente seleccionado.
	7.El sistema brinda la opción de buscar otro expediente.
8.El Especialista DS indica que desea realizar otra búsqueda.	9.El sistema realiza las acciones descritas en el paso 1 de esta Sección.

Prototipo de Interfaz



Flujos alternos

Acción del Actor	Respuesta del Sistema
	4.1 Si no existe ningún expediente que coincida con los parámetros indicados, el sistema muestra un mensaje "No se encontraron coincidencias".
10.1 Si el Especialista DS no desea realizar otra búsqueda, finaliza el CU.	
Poscondiciones:	-

Tabla 26 Descripción textual CU Visualizar Expediente de Fabricante.

Caso de Uso:	Gestionar RAPAC
Actores:	Especialista DS (Inicia)
Resumen:	<p>El CU inicia cuando el Especialista DS necesita realizar una de las siguientes acciones.</p> <p>Insertar un nuevo plan en el registro, el Especialista DS especifica los datos e indica guardar, el sistema guarda los datos y finaliza el caso de uso.</p> <p>Actualizar un plan, el Especialista DS selecciona el plan a actualizar e indica guardar los cambios, el sistema actualiza los nuevos datos y finaliza el caso de uso.</p> <p>Buscar un plan, el Especialista DS especifica las opciones y los criterios de búsqueda, el sistema muestra los planes que coincidan, finaliza así el caso de uso.</p> <p>Imprimir el registro, El Especialista DS indica imprimir el registro, el sistema muestra una vista previa, el usuario acepta y finaliza así el CU.</p>
Precondiciones:	El Especialista DS debe estar autenticado
Referencias	RF 2.1, RF 2.2, RF 2.3, RF 2.3.1, RF 2.3.2, RF 2.3.3, RF 2.3.4, RF 2.3.5, RF 2.4
Prioridad	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. Selecciona la opción "Registro de Auditorías y Planes de Acciones de Correctivas."	<p>2. Muestra las opciones</p> <p>a) Insertar y Actualizar datos del Registro de Auditorías y planes de Acciones Correctivas.</p> <p>b) Imprimir Registro de Auditorías y planes de Acciones Correctivas.</p>
3. Selecciona que opción desea realizar.	<p>4. En dependencia de la opción seleccionada por el Especialista DS el sistema realiza la acción correspondiente:</p> <p>a) Selecciona acción a: Ir a la sección "Insertar plan" y sección "Actualizar plan".</p>

	b) Selecciona acción b: Ir a la sección “Imprimir registro”.
Sección “Insertar plan”	
Acción del Actor	Respuesta del Sistema
	1.El sistema habilita los campos para insertar un nuevo plan. (Código de la Auditoría, fabricante, Fecha de la Auditoría, Fecha de Entrega del Plan de Acciones Correctivas, Revisión del Plan de Acciones Correctivas, Acciones de Seguimiento). (Ver Prototipo de Interfaz)
2.El Especialista llena los campos correspondientes al nuevo plan e indica salvar los cambios.	3.El sistema valida la completitud de los datos.
	4.El sistema inserta los datos del plan.
	5.El sistema muestra un mensaje ““Se insertó correctamente””.
	6.El sistema brinda la opción de insertar otro plan.
7.El Especialista DS indica realizar otra inserción.	8.El sistema va a la acción 1 de esta Sección.

Prototipo de Interfaz

El prototipo de interfaz muestra una ventana de diálogo con el título "Insertar Plan de Acciones Correctivas". El fondo de la ventana es verde claro. Contiene los siguientes campos de entrada:

- Código de la Auditoría:
- Fabricante:
- Revisión del Plan de Acciones Correctivas:
- Fecha entrega del Plan de Acciones Correctivas:
- Fecha de la Auditoría:
- Acciones de seguimiento:

En la parte inferior de la ventana hay dos botones: "Insertar" y "Cancelar".

Flujos Alternos Sección “Insertar plan”	
Acción del Actor	Respuesta del Sistema
	3.1 Si falta algún campo por llenar, el sistema emite un mensaje “Debe llenar los campos obligatorios”.
7.1 Si el Especialista DS no desea realizar otra inserción, termina el CU.	
Sección “Actualizar plan”	
Acción del Actor	Respuesta del Sistema
	1. El sistema brinda la opción de buscar un plan: Ir a la Sección “Buscar Plan”.
2. Selecciona el plan a modificar.	3. El sistema habilita los campos del plan para que sean modificados.
4. Modifica los campos que desea e indica guardar.	5. Valida los datos modificados.
	6. Actualiza los datos del plan.
	7. Muestra un mensaje ” Se modificó correctamente”.
	8. Posibilita realizar otra actualización.
9. Selecciona otro plan a actualizar.	10. El sistema realiza las acciones descritas en el paso 2 de esta sección.
Flujos Alternos Sección “Actualizar plan”	
Acción del Actor	Respuesta del Sistema
	5.1 Si algún campo obligatorio está vacío muestra un mensaje “Debe llenar los campos obligatorios”.
9.1 El especialista decide no realizar otra actualización, finaliza así el CU.	
Sección “Buscar Plan”	
Acción del Actor	Respuesta del Sistema
	1. El sistema muestra los campos por

	<p>los que se puede realizar la búsqueda:</p> <ul style="list-style-type: none"> • Código de la auditoría • Fabricante • Fecha de auditoría • Fecha de entrega • Revisión del plan
2. Selecciona el campo y especifica el criterio de búsqueda.	3. El sistema busca todos los planes que coincidan con el criterio especificado y los muestra.
	4. Posibilita realizar otra búsqueda.
5. Decide realizar otra búsqueda.	6. Realiza las acciones descritas en el paso 1 de esta Sección.
Flujos Alternos Sección “Buscar Plan”	
Acción del Actor	Respuesta del Sistema
5.1 Decide no realizar otra búsqueda, finaliza el CU.	
Sección “Imprimir registro”	
Acción del Actor	Respuesta del Sistema
	1.El sistema brinda la opción de buscar un plan: Ir a la Sección “Buscar Plan”.
2.Selecciona “Imprimir”.	3.Muestra una vista previa de lo se desea imprimir. Muestra las opciones de impresión.
4.Selecciona las opciones de impresión e indica “Imprimir”.	5.Brinda la posibilidad de realizar otra impresión.
6.Decide realizar otra impresión.	7.Ejecuta las acciones descritas en el paso 1 de esta sección.
Flujos Alternos Sección “Imprimir registro”	
Acción del Actor	Respuesta del Sistema
	5.1 Si no se puede imprimir el documento muestra un mensaje “Falló la impresión del

	documento”.
7.1 Decide no realizar otra impresión, finaliza el caso de uso.	
Poscondiciones	En dependencia de la acción realizada por el usuario: Se actualiza un plan Se inserta un plan

Tabla 27 Descripción textual CU Gestionar RAPAC

Caso de Uso:	Gestionar RPA
Actores:	Especialista DS (Inicia)
Resumen:	<p>El CU inicia cuando el Especialista DS necesita realizar una de las siguientes acciones.</p> <p>Insertar un nuevo programa en el registro, el Especialista DS especifica los datos e indica guardar, el sistema guarda los datos y finaliza el caso de uso.</p> <p>Actualizar un plan, el Especialista DS selecciona el programa a actualizar e indica guardar los cambios, el sistema actualiza los nuevos datos y finaliza el caso de uso.</p> <p>Buscar un programa, el Especialista DS especifica las opciones y los criterios de búsqueda, el sistema muestra los programas que coincidan, finaliza así el caso de uso.</p> <p>Imprimir el registro, El Especialista DS indica imprimir el registro, el sistema muestra una vista previa, el usuario acepta y finaliza así el CU.</p>
Precondiciones:	El Especialista DS debe estar autenticado.
Referencias	RF 3.1, RF 3.2, RF 3.3, RF 3.3.1, RF 3.3.2, RF 3.4
Prioridad	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. Selecciona la opción “Programa de Auditorías”.	<p>2. Muestra las opciones:</p> <p>a) Insertar y Actualizar datos del Registro Programa de Auditorías</p> <p>b) Imprimir Registro Programa de Auditorías.</p>

3. Selecciona que opción desea realizar.	4. En dependencia de la opción seleccionada por el Especialista DS el sistema realiza la acción correspondiente: a) Selecciona acción a: Ir a la sección “Insertar plan” y sección “Actualizar plan”. b) Selecciona acción b: Ir a la sección “Imprimir registro”.
Sección “Insertar Programa”	
Acción del Actor	Respuesta del Sistema
	1. El sistema habilita los campos para insertar un nuevo programa. (Fabricante, Causa de la Auditoría, Fecha de la Auditoría). (Ver Prototipo de Interfaz).
2. El Especialista llena los campos correspondientes al nuevo programa e indica guardar.	3. El sistema valida la completitud de los datos.
	4. Otorga un número consecutivo para el nuevo registro. 5. El sistema inserta los datos del programa.
	6. El sistema muestra un mensaje “Se insertó correctamente”.
	7. El sistema brinda la opción de insertar otro programa.
8. El Especialista indica realizar otra inserción.	9. El sistema va a la acción 1 de esta Sección.
Prototipo de interfaz	

The image shows a dialog box titled "Insertar Programa de Auditorías". It has a light green background and a blue border. Inside, there are four text input fields arranged in a 2x2 grid. The top-left field is labeled "Número de Auditoría", the top-right "Fabricante", the bottom-left "Fecha" (with a date mask of __/__/__), and the bottom-right "Causas de Auditoría". Below the fields are two buttons: "Insertar" on the left and "Cancelar" on the right. The dialog box also has standard window controls (minimize, maximize, close) in the top right corner.

Flujos Alternos Sección "Insertar Programa"

Acción del Actor	Respuesta del Sistema
	3.1 Si falta algún campo por llenar, el sistema emite un mensaje "Debe llenar los campos obligatorios".
8.1 Si el Especialista DS no desea realizar otra inserción, finaliza el CU.	

Sección "Actualizar programa"

Acción del Actor	Respuesta del Sistema
	1. El sistema brinda la opción de buscar un plan: Ir a la Sección "Buscar Plan".
2. Selecciona el programa a modificar.	3. El sistema habilita los campos del programa para que sean modificados.
4. Modifica los campos que desea e indica guardar.	5. Valida los datos modificados.
	6. Actualiza los datos del programa.
	7. Muestra un mensaje " Se modificó correctamente".
	8. Posibilita realizar otra actualización.
9. Selecciona otro programa a actualizar.	10. El sistema realiza las acciones descritas en el paso 2 de esta sección.

Flujos Alternos Sección "Actualizar programa"

Acción del Actor	Respuesta del Sistema
	5.1 Si algún campo obligatorio está vacío muestra un mensaje "Debe llenar los

	campos obligatorios”.
9.1 El Especialista DS decide no realizar otra actualización, finaliza así el CU.	
Sección “Buscar programa”	
Acción del Actor	Respuesta del Sistema
	1. El sistema muestra los campos por los que se puede realizar la búsqueda <ul style="list-style-type: none"> • Número • Nombre Fabricante • Fecha Auditoría • Causa Auditoría
2. Selecciona el campo y especifica el criterio de búsqueda.	3. El sistema busca todos los programas que coincidan con el criterio especificado y los muestra.
	4. Posibilita realizar otra búsqueda.
5. Decide realizar otra búsqueda.	6. Realiza las acciones descritas en el paso 1 de esta Sección.
Flujos Alternos Sección “Buscar programa”	
Acción del Actor	Respuesta del Sistema
5.1 Decide no realizar otra búsqueda, finaliza el CU.	
Sección “Imprimir registro”	
Acción del Actor	Respuesta del Sistema
	1. El sistema brinda la opción de buscar un plan: Ir a la Sección “Buscar Plan”.
2. Indica imprimir.	3. Muestra una vista previa de lo se desea imprimir. Muestra la opción de imprimir.
4. Selecciona las opciones de impresión e indica “Imprimir”.	5. Imprime el documento y muestra un mensaje “Se ha impreso el documento”.
	6. Brinda la posibilidad de realizar otra impresión.
7. Decide realizar otra impresión.	8. Ejecuta las acciones descritas en el paso 1 de esta sección.

Flujos Alternos Sección “Imprimir registro”	
Acción del Actor	Respuesta del Sistema
	5.1 Si no se puede imprimir el documento muestra un mensaje “Falló la impresión del documento”.
7.1 Decide no realizar otra impresión, finaliza el caso de uso.	
Poscondiciones:	En dependencia de la acción realizada por el usuario: Se actualiza un programa Se inserta un programa.

Tabla 28 Descripción textual CU Gestionar RPA.

Caso de Uso:	Gestionar RPC
Actores:	Especialista DS (Inicia)
Resumen:	<p>El CU inicia cuando el Especialista DS necesita realizar una de las siguientes acciones.</p> <p>Insertar un nuevo plan en el registro, el Especialista DS especifica los datos e indica guardar, el sistema guarda los datos y finaliza el caso de uso.</p> <p>Actualizar un plan, el Especialista DS selecciona el plan a actualizar e indica guardar los cambios, el sistema actualiza los nuevos datos y finaliza el caso de uso.</p> <p>Buscar un plan, el Especialista DS especifica las opciones y los criterios de búsqueda, el sistema muestra los planes que coincidan, finaliza así el caso de uso.</p> <p>Imprimir el registro, El Especialista DS indica imprimir el registro, el sistema muestra una vista previa, el usuario acepta y finaliza así el CU.</p>
Precondiciones:	El Especialista DS debe estar autenticado.
Referencias	RF 4.1, RF 4.2, RF 4.3, RF 4.3.1, RF 4.4
Prioridad	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema

1. Selecciona la opción “Registro de Planes de Calidad.”	2. Muestra las opciones: a) Insertar y Actualizar datos del Registro Planes de Calidad. b) Imprimir Registro Planes de Calidad.
3. Selecciona que opción desea realizar.	4. En dependencia de la opción seleccionada por el Especialista DS el sistema realiza la acción correspondiente: a) Selecciona acción a: Ir a la sección “Insertar plan” y sección “Actualizar plan”. b) Selecciona acción b: Ir a la sección “Imprimir registro”.
Sección “Insertar plan”	
Acción del Actor	Respuesta del Sistema
	1. El sistema habilita los campos para insertar un nuevo plan. (Número, Fabricante, Fecha de Entrega del Plan de Calidad, Revisión, Auditorías Propuestas, Auditorías Realizadas). (Ver Prototipo de Interfaz).
2. El Especialista DS llena los campos correspondientes al nuevo plan e indica guardar.	3. El sistema valida la completitud de los datos.
	4. El sistema inserta los datos del plan.
	5. El sistema muestra un mensaje “Se insertó correctamente”.
	6. El sistema brinda la opción de insertar otro plan.
7. El Especialista DS indica realizar otra inserción.	8. El sistema va a la acción 1 de esta Sección.

Prototipo de Interfaz

Insertar planes en el Registro de Plan de Calidad

código:SV.02-01

No

Fabricante

Fecha de entrega

Revisión

Auditorías

Propuestas

Auditorías

Realizadas

Insertar Cancelar

Flujos Alternos Sección "Insertar plan"

Acción del Actor	Respuesta del Sistema
	3.1 Si falta algún campo por llenar, el sistema emite un mensaje "Debe llenar los campos obligatorios".
7.1 Si el Especialista DS no desea realizar otra inserción, termina el CU.	

Sección "Actualizar plan"

Acción del Actor	Respuesta del Sistema
	1. El sistema brinda la opción de buscar un plan: Ir a la Sección "Buscar Plan".
2. Selecciona el plan a modificar.	3. El sistema habilita los campos del plan para que sean modificados.
4. Modifica los campos que desea e indica guardar.	5. Valida los datos modificados.
	6. Actualiza los datos del plan.
	7. Muestra un mensaje " Se modificó correctamente".
	8. Posibilita realizar otra actualización.
9. Selecciona otro plan a actualizar.	10. El sistema realiza las acciones descritas en el paso 2 de esta sección.

Flujos Alternos Sección "Actualizar plan"

Acción del Actor	Respuesta del Sistema
	5.1 Si algún campo obligatorio está vacío muestra un mensaje “Debe llenar los campos obligatorios”.
9.1 El Especialista DS decide no realizar otra actualización, finaliza así el CU.	
Sección “Buscar Plan”	
Acción del Actor	Respuesta del Sistema
	1.El sistema muestra los campos por los que se puede realizar la búsqueda <ul style="list-style-type: none"> • Número • Fabricante • Fecha de Entrega • Revisión • Auditorías Propuestas • Auditorías Realizadas
2.Selecciona el campo y especifica el criterio de búsqueda.	3.El sistema busca todos los planes que coincidan con los criterios especificados y los muestra.
	4.Posibilita realizar otra búsqueda.
5.Decide realizar otra búsqueda.	6.Realiza las acciones descritas en el paso 1 de esta Sección.
Flujos Alternos Sección “Buscar Plan”	
Acción del Actor	Respuesta del Sistema
5.1 Decide no realizar otra búsqueda, finaliza el CU.	
Sección “Imprimir registro”	
Acción del Actor	Respuesta del Sistema
	1. El sistema brinda la opción de buscar un plan: Ir a la Sección “Buscar Plan”.
2. Indica imprimir.	3. Muestra una vista previa de lo se desea imprimir. Muestra las opciones de impresión.

4. Selecciona las opciones de impresión e indica "Imprimir".	5. Imprime el documento y muestra un mensaje "Se ha impreso el documento"
	6. Brinda la posibilidad de realizar otra impresión.
7. Decide realizar otra impresión.	8. Ejecuta las acciones descritas en el paso 1 de esta sección.
Flujos Alternos Sección "Imprimir registro"	
Acción del Actor	Respuesta del Sistema
	5.1 Si no se puede imprimir el documento muestra un mensaje "Falló la impresión del documento"
7.1 Decide no realizar otra impresión, finaliza el caso de uso.	
Poscondiciones	En dependencia de la acción realizada por el usuario: Se actualiza un plan Se inserta un plan

Tabla 29 Descripción textual CU Gestionar RPC

Caso de Uso:	Gestionar planilla LCCF
Actores:	Especialista DS (Inicia)
Resumen:	<p>El CU se inicia cuando el Especialista DS decide realizar una de la siguientes acciones:</p> <p>Registrar los datos en la planilla, introduce los datos, el sistema los guarda finalizando el CU.</p> <p>Actualizar datos en la planilla, el Especialista DS selecciona la planilla a modificar, inserta los datos, el sistema registra los cambios finalizando así el CU.</p> <p>Buscar y visualizar planilla, el Especialista DS escoge las opciones de búsqueda e inserta los criterios de búsqueda, el sistema muestra las coincidencias finalizando así el CU.</p> <p>Imprimir planilla, el Especialista DS escoge la planilla a imprimir, el sistema muestra las opciones de impresión e imprime el documento, finalizando así el CU.</p>

	Calcular estadísticas, el Especialista DS selecciona las estadísticas a calcular, el sistema realiza los cálculos y muestra los resultados en forma de tabla.
Precondiciones:	El Especialista DS debe estar autenticado
Referencias	RF 5.1, RF 5.2, RF 5.3, RF 5.3.1, RF 5.4, RF 5.5.
Prioridad	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El Especialista DS selecciona “Lista de Chequeo de Calidad para Fabricante”.	2. El sistema muestra las opciones: a) Registrar datos en la planilla Lista de Chequeo. b) Buscar/Actualizar datos en la planilla Lista de Chequeo. c) Calcular estadísticas d) Imprimir Lista de Chequeo.
3. Selecciona la opción.	4. En dependencia de la opción seleccionada el sistema realiza una de las siguientes acciones. a) Registrar datos en la planilla Lista de Chequeo: Ir a la sección “Registrar datos en la planilla” b) Buscar/Actualizar datos en la planilla Lista de Chequeo: Ir a la sección “Actualizar datos en la planilla” y “Buscar y visualizar planilla” c) Calcular estadísticas: Ir a la sección “Calcular estadísticas” d) Imprimir Lista de Chequeo: Ir a la sección “Imprimir planilla”.
Sección “Registrar datos en la planilla”	
Acción del Actor	Respuesta del Sistema

<p>1. Selecciona el tipo de fabricante. Certificado o no Certificado.</p>	<p>2. El sistema habilita los campos según el tipo de fabricante. (Ver Prototipo de Interfaz)</p>
<p>3. Inserta los datos necesarios e indica guardar.</p>	<p>4. Valida la completitud de los datos</p>
	<p>5. Inserta los datos y muestra un mensaje:</p> <ul style="list-style-type: none"> • Si se insertó un fabricante certificado: “Se ha insertado una Lista de Chequeo para fabricante certificado correctamente.” • Si se insertó un fabricante no certificado: “Se ha insertado una lista de chequeo para fabricante no certificado correctamente.”

Prototipo de interfaz

1. Conformidad del sistema de la calidad con la Norma Internacional ISO y/o Directivas CCE y/o buenas prácticas de fabricación...

ISO 9001 ISO 13485 BPF Marcado CE

2. Organismo certificador

Nombre del fabricante

3. Declaración del productor si no está certificado

Número del expediente

Fecha de revisión

4. Plan de Calidad

Clase de riesgo mayor de los equipos que declaran

Flujos Alternos Sección “Registrar datos en la planilla”

Acción del Actor	Respuesta del Sistema
	<p>4.1 Si falta algún campo por llenar muestra el mensaje “Debe obligatoriamente llenar el</p>

	campo”.
Sección “Actualizar datos en la planilla”	
Acción del Actor	Respuesta del Sistema
	1. El sistema ejecuta las acciones descritas en la sección “Buscar y visualizar planilla”.
2. El Especialista DS selecciona una planilla e indica actualizar.	3. Muestra los datos insertados y habilita los campos para modificarlos.
4. Actualiza los datos e indica Modificar.	5. Muestra un mensaje pidiendo confirmación.
6. Indica Aceptar.	7. Valida la completitud de los datos.
	8. Guarda los datos y muestra un mensaje “Se ha modificado una Lista de Chequeo correctamente. “”
Flujos Alternos Sección “Actualizar datos en la planilla”	
Acción del Actor	Respuesta del Sistema
6.1 Indica Cancelar, finaliza el CU.	5.1 Si falta algún campo por llenar muestra el mensaje “Este campo es requerido”.
Sección “Buscar y visualizar planilla”	
Acción del Actor	Respuesta del Sistema
	1. El sistema muestra los campos por los que se puede realizar la búsqueda. <ul style="list-style-type: none"> • Número de Expediente • Fecha de revisión • Certificado • Clase de Riesgo mayor • Conclusiones • Revisado por
2. Selecciona el campo y especifica el criterio de búsqueda.	3. El sistema busca todas las planillas que coincidan con el criterio especificados y muestra de cada una: <ul style="list-style-type: none"> • Número de Expediente • Fecha de revisión

	<ul style="list-style-type: none"> • Certificado • Clase de Riesgo mayor • Conclusiones • Revisado por
4. Selecciona ver los datos de una planilla.	5. El sistema muestra todos los datos de la planilla.
	6. El sistema da la opción de buscar otra planilla.
7. El Especialista DS indica que desea realizar otra búsqueda.	8. El sistema realiza las acciones descritas en el paso 1 de esta Sección.
Flujos Alternos Sección “Buscar y visualizar planilla”	
Acción del Actor	Respuesta del Sistema
7.1 Decide no realizar otra búsqueda, finaliza el CU.	
Sección “Calcular estadísticas”	
Acción del Actor	Respuesta del Sistema
	<p>1. El sistema muestra los datos:</p> <ul style="list-style-type: none"> • Norma • Fecha de revisión • Organismo certificador • Certificado o no (%) • Pertenciente la mismo certificador (%) <p>De cada una de las planilla existentes en el sistema y da la opcion de agrupar los resultados por cualquiera de los campos anteriores.</p>
2. El Especialista DS escoge un campo e indica agrupar.	3. El sistema realiza los cálculos necesarios y muestra los resultados.
	4. Brinda la opción de realizar otro cálculo.
5. Decide realizar otro cálculo de las estadísticas.	6. Realiza las acciones descritas en el paso 3 de esta sección.
Flujos Alternos Sección “Calcular estadísticas”	

Acción del Actor	Respuesta del Sistema
5.1 El Especialista DS decide no realizar otro cálculo, finaliza así el CU.	
Sección "Imprimir planilla"	
Acción del Actor	Respuesta del Sistema
	1. El sistema ejecuta las acciones descritas en la sección "Buscar y visualizar planilla".
2. Selecciona la planilla a imprimir e indica imprimir.	3. Muestra una vista previa de lo se desea imprimir. Muestra las opciones de impresión.
4. Selecciona las opciones de impresión e indica "Imprimir".	5. Imprime el documento y muestra un mensaje "Se ha impreso el documento"
	6. Brinda la posibilidad de realizar otra impresión.
7. Decide realizar otra impresión.	8. Ejecuta las acciones descritas en el paso 1 de esta sección.
Flujos Alternos Sección "Imprimir planilla"	
Acción del Actor	Respuesta del Sistema
	5.1 Si no se puede imprimir el documento muestra un mensaje "Falló la impresión del documento".
7.1 Decide no realizar otra impresión, finaliza el caso de uso.	
Poscondiciones	En dependencia de la acción realizada por el usuario: Se actualiza un planilla Se insertan datos en una planilla

Tabla 30 Descripción textual CU Gestionar planilla LCCF

Caso de Uso:	Gestionar planilla Plan de Auditoría
Actores:	Especialista DS (Inicia)
Resumen:	El CU se inicia cuando el Especialista DS decide realizar una de la

	<p>siguientes acciones:</p> <p>Registrar los datos en la planilla, introduce los datos y el sistema los guarda finalizando el CU.</p> <p>Actualizar datos en la planilla, el Especialista DS selecciona la planilla a modificar, inserta los datos, el sistema registra los cambios finalizando así el CU.</p> <p>Buscar y visualizar planilla, el Especialista DS escoge las opciones de búsqueda e inserta os criterios de búsqueda, el sistema muestra las coincidencia finalizando así el CU.</p> <p>Imprimir planilla, el Especialista DS escoge la planilla a imprimir, el sistema muestra las opciones de impresión e imprime el documento, finalizando así el CU.</p>
Precondiciones:	El Especialista DS debe estar autenticado.
Referencias	RF 6.1, RF 6.2, RF 6.3, RF 6.3.1, RF 6.4
Prioridad	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El Especialista DS va a al opción “Plan de la Auditoría”.	<p>2. El sistema muestra las opciones:</p> <p>a) Registrar datos en la planilla Plan de la Auditoría.</p> <p>b) Buscar/Actualizar datos en la planilla Plan de la Auditoría.</p> <p>c) Imprimir Plan de la Auditoría.</p>
3. El Especialista selecciona una opción.	<p>4. En dependencia de la opción seleccionada el sistema realiza una de las siguientes acciones.</p> <p>a) Registrar datos en la planilla Plan de la Auditoría: Ir a la sección “Registrar datos en la planilla”</p> <p>b) Buscar/Actualizar datos en la planilla Plan de la Auditoría: Ir a la sección “Actualizar datos en la planilla” y “Buscar y visualizar</p>

	planilla”. c) Imprimir Plan de la Auditoría: Ir a la sección “Imprimir planilla”.
Sección “Registrar datos en la planilla”	
Acción del Actor	Respuesta del Sistema
	1. El sistema muestra la interfaz de la planilla y habilita los campos para insertar los datos. (Ver Prototipo de Interfaz).
2. Inserta los datos necesarios e indica guardar.	3. Valida la completitud de los datos
	4. Guarda los datos y muestra un mensaje “El Plan de la Auditoría se ha salvado corretamente”.

Prototipo de Interfaz

Flujos Alternos Sección “Registrar datos en la planilla”

Acción del Actor	Respuesta del Sistema
	3.1 Si falta algún campo por llenar muestra el mensaje “Este campo es requerido”.

Sección “Actualizar datos en la planilla”

Acción del Actor	Respuesta del Sistema
	1. El sistema ejecuta las acciones descritas

	en la sección “Buscar y visualizar planilla”.
2. El Especialista DS selecciona la planilla a modificar.	3. Muestra los datos insertados y habilita los campos para modificarlos.
4. Actualiza los datos e indica Guardar.	5. Muestra un mensaje de confirmación.
6. Indica Aceptar.	7. Valida la completitud de los datos.
	8. Guarda los datos y muestra un mensaje “La planilla Plan de la Auditoría se ha modificado correctamente”.
Flujos Alternos Sección “Actualizar datos en la planilla”	
Acción del Actor	Respuesta del Sistema
6.1 Indica Cancelar, finaliza el CU.	7.1 Si falta algún campo obligatorio por llenar muestra un mensaje “Este campo es requerido”.
Sección “Buscar y visualizar planilla”	
Acción del Actor	Respuesta del Sistema
	1. El sistema muestra los campos por los que se puede realizar la búsqueda: <ul style="list-style-type: none"> • Código • Tipo Auditoría • Fabricante • Fecha desde • Fecha hasta
2. Selecciona el campo y especifica el criterio de búsqueda.	3. El sistema busca todas las planillas que coincidan con el criterio especificados y muestra de cada una: <ul style="list-style-type: none"> • Código • Tipo Auditoría • Fabricante • Fecha desde • Fecha hasta
4. Selecciona ver los datos de una planilla.	5. El sistema muestra todos los datos de la planilla.

	6. El sistema da la opción de buscar otra planilla.
7. El Especialista DS indica que desea realizar otra búsqueda.	8. El sistema realiza las acciones descritas en el paso 1 de esta Sección.
Flujos Alternos Sección “Buscar y visualizar planilla”	
Acción del Actor	Respuesta del Sistema
7.1 Decide no realizar otra búsqueda, finaliza el CU.	
Sección “Imprimir planilla”	
Acción del Actor	Respuesta del Sistema
	1. El sistema ejecuta las acciones descritas en la sección “Buscar y visualizar planilla”.
2. Selecciona la planilla a imprimir e indica imprimir.	3. Muestra una vista previa de la planilla que desea imprimir. Muestra también las opciones de impresión.
4. Selecciona las opciones de impresión e indica “Imprimir”.	5. Imprime el documento y muestra un mensaje “Se ha impreso el documento”.
	6. Brinda la posibilidad de realizar otra impresión.
7. Decide realizar otra impresión.	8. Ejecuta las acciones descritas en el paso 1 de esta sección.
Flujos Alternos Sección “Imprimir planilla”	
Acción del Actor	Respuesta del Sistema
	5.1 Si no se puede imprimir el documento muestra un mensaje “Falló la impresión del documento”.
7.1 Decide no realizar otra impresión, finaliza el caso de uso.	
Poscondiciones	En dependencia de la acción realizada por el usuario: Se actualiza un planilla Se insertan datos en una planilla

Tabla 31 Descripción textual CU Gestionar planilla Plan de Auditoría.

Caso de Uso:	Gestionar planilla IA
Actores:	Especialista DS (Inicia)
Resumen:	<p>El CU se inicia cuando el Especialista DS decide realizar una de la siguientes acciones:</p> <p>Registrar los datos en la planilla, introduce los datos y el sistema los guarda finalizando el CU.</p> <p>Actualizar datos en la planilla, el Especialista DS selecciona la planilla a modificar, inserta los datos, el sistema registra los cambios finalizando así el CU.</p> <p>Buscar y visualizar planilla, el Especialista DS escoge las opciones de búsqueda e inserta os criterios de búsqueda, el sistema muestra las coincidencia finalizando así el CU.</p> <p>Imprimir planilla, el Especialista DS escoge la planilla a imprimir, el sistema muestra las opciones de impresión e imprime el documento, finalizando así el CU.</p>
Precondiciones:	El Especialista DS debe estar autenticado
Referencias	RF 8.1, RF 8.2, RF 8.3, RF 8.3.1, RF 8.4
Prioridad	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El Especialista DS va a la opción Planilla Informe de la Auditoría.	2. El sistema muestra las opciones: <ul style="list-style-type: none"> a) Registrar datos en la planilla Informe de la Auditoría. b) Buscar/Actualizar datos en la planilla Informe de la Auditoría. c) Imprimir Informe de la Auditoría.
3. Selecciona una opción.	4. En dependencia de la opción seleccionada el sistema realiza una de las siguientes acciones: <ul style="list-style-type: none"> a) Registrar datos en la planilla

	<p>Informe de la Auditoría: Ir a la sección “Registrar datos en la planilla”.</p> <p>b) Buscar/Actualizar datos en la planilla Informe de la Auditoría: Ir a la sección “Actualizar datos en la planilla” y “Buscar y visualizar planilla”.</p> <p>c) Imprimir Informe de la Auditoría: Ir a la sección “Imprimir planilla”.</p>
Sección “Registrar datos en la planilla”	
Acción del Actor	Respuesta del Sistema
	1. El sistema muestra la interfaz de la planilla y habilita los campos para insertar los datos. (Ver Prototipo de Interfaz)
2. Inserta los datos necesarios e indica guardar.	3. Valida la completitud de los datos.
	4. Guarda los datos y muestra un mensaje “La planilla Informe de la Auditoría se ha insertado correctamente.”
Prototipos de Interfaz	

Registrar Datos código:SV.02-07

Tipo de auditoría: Inicial, Seguimiento, Especial

Fabricante a Auditar: [Campo de texto]

Fecha de la Auditoría in situ: Desde: [Campo de fecha] Hasta: [Campo de fecha]

Clase de Riesgo: [Campo de texto]

Objetivos de la Auditoría: Otorgar la inscripción o reinscripción del Fabricante, Comprobar el plan de acciones correctivas, Comprobar el cumplimiento del plan de calidad

Criterio de la Auditoría: NC ISO 13485:2005, ISO 14969:2004, Regulación ER11, Reglamento para la evaluación y registro de equipos médicos

Alcance de la Auditoría: Gerencia, Diseño, Realización del producto, Control de contrataciones, Control de documentos y registros, Procesos relacionados con el cliente, Acciones correctivas y preventivas, Documentación del productor

Equipo médico a auditar: [Campo de texto]

Guía de la Auditoría: [Campo de texto]

Cargo: [Campo de texto]

Idioma de la Auditoría: [Lista desplegable]

Instalaciones a visitar: [Campo de texto]

No conformidades de la Auditoría: [Campo de texto]

Conclusiones de la Auditoría: Como resultado de la Auditoría, el equipo auditor concluye lo siguiente: [Campo de texto]

Gerencia: Satisfactorio, No satisfactorio

Diseño y desarrollo: Satisfactorio, No satisfactorio

Acciones Correctivas y preventivas: Satisfactorio, No satisfactorio

Control de compras: Satisfactorio, No satisfactorio

Documentación del producto: Satisfactorio, No satisfactorio

Producción y control del proceso: Satisfactorio, No satisfactorio

Documentación y registro: Satisfactorio, No satisfactorio

Procesos relacionados con el cliente: Satisfactorio, No satisfactorio

En cuanto a las No Conformidades de la auditoría, se concluye que: [Campo de texto]

El equipo auditor propone: [Campo de texto]

En cuanto al comportamiento global del Sistema de Calidad, se concluye que: Satisfactorio, No satisfactorio

[Botón Registrar] [Botón Cancelar]

Flujos Alternos Sección “Registrar datos en la planilla”

Acción del Actor	Respuesta del Sistema
	3.1 Si falta algún campo por llenar muestra el mensaje “Este campo es requerido”.
Sección “Actualizar datos en la planilla”	
Acción del Actor	Respuesta del Sistema
	1. El sistema ejecuta las acciones descritas en la sección “Buscar y visualizar planilla”.
2. El Especialista DS selecciona la planilla a modificar.	3. Muestra los datos insertados y habilita los campos para modificarlos.
4. Actualiza los datos e indica Guardar.	5. Muestra un mensaje de confirmación.

6. Indica Aceptar.	7. Valida la completitud de los datos.
	8. Guarda los datos y muestra un mensaje “Se ha modificado la planilla Informe de la Auditoria correctamente”.
Flujos Alternos Sección “Actualizar datos en la planilla”	
Acción del Actor	Respuesta del Sistema
6.1 Indica Cancelar, finaliza el CU.	7.1 Si falta un campo por llenar muestra el mensaje “Este campo es requerido”.
Sección “Buscar y visualizar planilla”	
Acción del Actor	Respuesta del Sistema
	1. El sistema muestra los campos por los que se puede realizar la búsqueda: <ul style="list-style-type: none"> • Código • Tipo Auditoría • Fabricante • Fecha desde • Fecha hasta
2. Selecciona el campo y especifica el criterio de búsqueda.	3. El sistema busca todas las planillas que coincidan con el criterio especificados y muestra de cada una: <ul style="list-style-type: none"> • Código • Tipo Auditoría • Fabricante • Fecha desde • Fecha hasta
4. Selecciona ver los datos de una planilla.	5. El sistema muestra todos los datos de la planilla.
	6. El sistema da la opción de buscar otra planilla.
7. El Especialista DS indica que desea realizar otra búsqueda.	8. El sistema realiza las acciones descritas en el paso 1 de esta Sección.
Flujos Alternos Sección “Buscar y visualizar planilla”	
Acción del Actor	Respuesta del Sistema

7.1 Decide no realizar otra búsqueda, finaliza el CU.	
Sección “Imprimir planilla”	
Acción del Actor	Respuesta del Sistema
	1. El sistema ejecuta las acciones descritas en la sección “Buscar y visualizar planilla”.
2. Selecciona la planilla a imprimir e indica imprimir.	3. Muestra una vista previa de la planilla que desea imprimir. Muestra también las opciones de impresión.
4. Selecciona las opciones de impresión e indica “Imprimir”.	5. Imprime el documento y muestra un mensaje “Se ha impreso el documento”.
	6. Brinda la posibilidad de realizar otra impresión.
7. Decide realizar otra impresión.	8. Ejecuta las acciones descritas en el paso 1 de esta sección.
Flujos Alternos Sección “Imprimir planilla”	
Acción del Actor	Respuesta del Sistema
	5.1 Si no se puede imprimir el documento muestra un mensaje “Falló la impresión del documento”.
7.1 Decide no realizar otra impresión, finaliza el caso de uso.	
Poscondiciones	En dependencia de la acción realizada por el usuario: Se actualiza una planilla Se insertan datos en una planilla

Tabla 32 Descripción textual CU Gestionar planilla IA

Caso de Uso:	Gestionar planilla Registro de Asistencia
Actores:	Especialista DS (Inicia)
Resumen:	El CU se inicia cuando el Especialista DS decide realizar una de las siguientes acciones:

	<p>Registrar los datos en la planilla, introduce los datos y el sistema los guarda finalizando el CU.</p> <p>Actualizar datos en la planilla, el Especialista DS selecciona la planilla a modificar, inserta los datos, el sistema registra los cambios finalizando así el CU.</p> <p>Buscar y visualizar planilla, el Especialista DS escoge las opciones de búsqueda e inserta los criterios de búsqueda, el sistema muestra las coincidencias finalizando así el CU.</p> <p>Imprimir planilla, el Especialista DS escoge la planilla a imprimir, el sistema muestra las opciones de impresión e imprime el documento, finalizando así el CU.</p>
Precondiciones:	El Especialista DS debe estar autenticado
Referencias	RF 13.1, RF 13.2, RF 13.3, RF 13.4
Prioridad	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El Especialista DS va a la opción Registro de Asistencia.	2. El sistema muestra las opciones: <ul style="list-style-type: none"> d) Registrar datos en la planilla Registro de Asistencia. e) Buscar/Actualizar datos en la planilla Registro de Asistencia. f) Imprimir Registro de Asistencia.
3. Selecciona una opción.	4. En dependencia de la opción seleccionada el sistema realiza una de las siguientes acciones: <ul style="list-style-type: none"> d) Registrar datos en la planilla Registro de Asistencia: Ir a la sección "Registrar datos en la planilla". e) Buscar/Actualizar datos en la planilla Registro de Asistencia: Ir a la sección "Actualizar datos en la planilla" y "Buscar y visualizar

	planilla”. f) Imprimir Registro de Asistencia: Ir a la sección “Imprimir planilla”.
Sección “Registrar datos en la planilla”	
Acción del Actor	Respuesta del Sistema
	1. El sistema muestra la interfaz de la planilla y habilita los campos para insertar los datos. (Ver Prototipo de Interfaz)
2. Inserta los datos necesarios e indica guardar.	3. Valida la completitud de los datos.
	4. Guarda los datos y muestra un mensaje “La planilla Registro de Asistencia se ha insertado correctamente”.

Prototipo de Interfaz

El prototipo de interfaz muestra una ventana con el título "Insertar Registro de Asistencia". Dentro de la ventana, hay un campo de lista desplegable para "Código de auditoría", un campo para "Fecha:", y dos campos de texto para "Desde:" y "Hasta:". Abajo de esto, hay una tabla con cuatro columnas: "Nombre y Apellidos", "Cargo", "Firma Apertura" y "Firma Clausura". La tabla tiene tres filas vacías. En la parte inferior de la ventana, hay dos botones: "Aceptar" y "Cancelar".

Flujos Alternos Sección “Registrar datos en la planilla”	
Acción del Actor	Respuesta del Sistema
	3.1 Si falta algún campo por llenar muestra

	el mensaje “Debe llenar los campos obligatorios”.
Sección “Actualizar datos en la planilla”	
Acción del Actor	Respuesta del Sistema
	1. El sistema ejecuta las acciones descritas en la sección “Este campo es requerido”.
2. El Especialista DS selecciona la planilla a modificar.	3. Muestra los datos insertados y habilita los campos para modificarlos.
4. Actualiza los datos e indica Guardar.	5. Muestra un mensaje de confirmación.
6. Indica Aceptar.	7. Valida la completitud de los datos.
	8. Guarda los datos y muestra un mensaje “La planilla Registro de Asistencia se ha modificado correctamente”.
Flujos Alternos Sección “Actualizar datos en la planilla”	
Acción del Actor	Respuesta del Sistema
6.1 Indica Cancelar, finaliza el CU.	7.1 Si falta algún campo muestra un mensaje “Este campo es requerido”.
Sección “Buscar y visualizar planilla”	
Acción del Actor	Respuesta del Sistema
	1. El sistema muestra los campos por los que se puede realizar la búsqueda: <ul style="list-style-type: none"> • Código • Fecha apertura • Fecha clausura • Fabricante
2. Selecciona el campo y especifica el criterio de búsqueda.	3. El sistema busca todas las planillas que coincidan con el criterio especificados y muestra de cada una: <ul style="list-style-type: none"> • Código • Fecha apertura • Fecha clausura • Fabricante
4. Selecciona ver los datos de una	5. El sistema muestra todos los datos de la

planilla.	planilla.
	6. El sistema da la opción de buscar otra planilla.
7. El Especialista DS indica que desea realizar otra búsqueda.	8. El sistema realiza las acciones descritas en el paso 1 de esta Sección.
Flujos Alternos Sección “Buscar y visualizar planilla”	
Acción del Actor	Respuesta del Sistema
7.1 Decide no realizar otra búsqueda, finaliza el CU.	
Sección “Imprimir planilla”	
Acción del Actor	Respuesta del Sistema
	1. El sistema ejecuta las acciones descritas en la sección “Buscar y visualizar planilla”.
2. Selecciona la planilla a imprimir e indica imprimir.	3. Muestra una vista previa de la planilla que desea imprimir. Muestra también las opciones de impresión.
4. Selecciona las opciones de impresión e indica “Imprimir”.	5. Imprime el documento y muestra un mensaje “Se ha impreso el documento”
	6. Brinda la posibilidad de realizar otra impresión.
7. Decide realizar otra impresión.	8. Ejecuta las acciones descritas en el paso 1 de esta sección.
Flujos Alternos Sección “Imprimir planilla”	
Acción del Actor	Respuesta del Sistema
	5.1 Si no se puede imprimir el documento muestra un mensaje “Falló la impresión del documento”.
7.1 Decide no realizar otra impresión, finaliza el caso de uso.	
Poscondiciones	En dependencia de la acción realizada por el usuario: Se actualiza una planilla Se insertan datos en una planilla

Tabla 33 Descripción textual CU Gestionar planilla Registro de Asistencia.

Caso de Uso:	Generar aviso.	
Actores:	Reloj (Inicia)	
Resumen:	El caso de uso se inicia cuando se ejecuta la aplicación por alguno de los usuarios autenticados. Los reportes generales se generan de forma automática por el sistema. El caso de uso finaliza cuando los reportes son generados.	
Precondiciones:	Debe estar planificado el aviso.	
Referencias	RF 14.1, RF 14.2, RF 14.3	
Prioridad	Secundario	
Flujo Normal de Eventos		
	Acción del Actor	Respuesta del Sistema
	1. El usuario entra al sistema.	2. El sistema verifica que existan avisos pendientes. Muestra un mensaje "Debe revisar los avisos pues se vencen unos próximamente".
Poscondiciones	-	

Tabla 34 Descripción textual CU Generar aviso

Anexo 6: Descripciones de clases del diseño.

Nombre de la clase: RAPACActions	
Tipo : Controladores	
Descripción: Representa las acciones del módulo RAPAC	
Método	Descripción
executeGetGrid(\$request)	Obtiene los atributos de de la base de datos.
executeGetFabricante(\$request)	Obtiene el nombre de todos los fabricantes.
executeUpdate(\$request)	Actualiza campos en la base de datos.
executePlanificarNotificacion(\$request)	Inserta un nuevo aviso relacionado con un registro.
executeGetCodigo()	Obtiene el código de la auditoría.

Nombre de la clase: RPAActions	
Tipo : Controladores	
Descripción: Representa las acciones de un módulo RPA .	
Método	Descripción

executeGetGrid(\$request)	Obtiene los atributos de de la base de datos.
executeGetFabricante(\$request)	Obtiene el nombre de todos los fabricantes.
executeUpdate(\$request)	Actualiza campos en la base de datos.
executePlanificarNotificacion(\$request)	Inserta un nuevo aviso relacionada con un registro.
executeGetCodigo()	Obtiene el código de la auditoría.

Nombre de la clase: RPCActions	
Tipo : Controladores	
Descripción: Representa las acciones de un módulo RPC.	
Método	Descripción
executeGetGrid(\$request)	Obtiene los atributos de de la base de datos.
executeGetFabricante(\$request)	Obtiene el nombre de todos los fabricantes.
executeUpdate(\$request)	Actualiza campos en la base de datos.
executePlanificarNotificacion(\$request)	Inserta un nuevo aviso relacionada con un registro.

Nombre de la clase: IAActions	
Tipo : Controladores	
Descripción: Representa las acciones de un módulo IA.	
Método	Descripción
executeGetGrid(\$request)	Obtiene los atributos de de la base de datos.
executeGetFabricante(\$request)	Obtiene el nombre de todos los fabricantes.
executeUpdate(\$request)	Actualiza campos en la base de datos.
executeGetCriterio()	Obtiene los criterios de la auditoría.
executeGetCodigo()	Obtiene el código de la auditoría.
executeGetObjetivo()	Obtiene los objetivos de la auditoría.
executeGetAlcance()	Obtiene alcance de la auditoría.
executeInsertar(\$request)	Inserta datos en la base de datos.
executePrint(\$request)	Muestra la vista previa para impresión.
executeGetTipoAuditoria()	Obtiene el tipo de la auditoría.
executeGetClaseRiesgo()	Obtiene la clase de riesgo del fabricante.

Nombre de la clase: LCCFActions	
Tipo : Controladores	
Descripción: Representa las acciones de un módulo LCCF	
Método	Descripción
executeGetGrid(\$request)	Obtiene los atributos de de la base de datos.
executeGetFabricante(\$request)	Obtiene el nombre de todos los fabricantes.
executeUpdate(\$request)	Actualiza campos en la base de datos para fabricantes certificados.
executeInsertar(\$request)	Inserta datos en la base de datos para fabricantes no certificados.
executePrint(\$request)	Muestra la vista previa para impresión.
executeUpdate1(\$request)	Actualiza campos en la base de datos para

	fabricantes no certificados.
executeInsertarCert(\$request)	Inserta datos en la base de datos para fabricantes certificados.
executeCalculo1(\$request)	Calcula cantidades de los registros de la base de datos.

Nombre de la clase: PAActions	
Tipo : Controladores	
Descripción: Representa las acciones de un módulo PA.	
Método	Descripción
executeGetGrid(\$request)	Obtiene los atributos de de la base de datos.
executeGetFabricante(\$request)	Obtiene el nombre de todos los fabricantes.
executeUpdate(\$request)	Actualiza campos en la base de datos.
executeGetCriterio()	Obtiene los criterios de la auditoría.
executeGetObjetivo()	Obtiene los objetivos de la auditoría.
executeGetAlcance()	Obtiene alcance de la auditoría.
executeInsertar(\$request)	Inserta datos en la base de datos.
executePrint(\$request)	Muestra la vista previa para impresión.
executeGetTipoAuditoria()	Obtiene el tipo de la auditoría.
executeGetClaseRiesgo()	Obtiene la clase de riesgo del fabricante.

Nombre de la clase: RAActions	
Tipo : Controladores	
Descripción: Representa las acciones de un módulo RA.	
Método	Descripción
executeGetGrid(\$request)	Obtiene los atributos de de la base de datos.
executeGetFabricante(\$request)	Obtiene el nombre de todos los fabricantes.
executeUpdate(\$request)	Actualiza campos en la base de datos.
executeGetCodigo()	Obtiene el código de la auditoría.
executeInsertar(\$request)	Inserta datos en la base de datos.
executePrint(\$request)	Muestra la vista previa para impresión.
executeGetAsistencia()	Obtiene datos de la tabla asistencia.

Nombre de la clase: AvisosActions	
Tipo : Controladores	
Descripción: Representa las acciones de un módulo Avisos.	
Método	Descripción
executeGetGrid(\$request)	Obtiene los atributos de de la base de datos.
executeUpdate(\$request)	Actualiza campos en la base de datos.
executeDelete(\$request)	Elimina un aviso.

Nombre de la clase: PrincipalActions	
Tipo : Controladores	
Descripción: Representa las acciones de un módulo Principal.	

Método	Descripción
executeAviso(\$request)	Recalcula la fecha de los avisos y muestra el mensaje.

Anexo 7: Descripciones de las tablas de la base de datos.

Tabla: t_acciones_correctivas		
Descripción: Tabla que guarda los datos de las acciones correctivas		
Nombre atributo	Tipo de dato	Comentario
id_acciones_correctivas	INTEGER	Identificador de la tabla.
accion	VARCHAR	Acción correctiva que se inserta.

Tabla: t_alcance		
Descripción: Tabla que guarda el alcance de las auditorías		
Nombre atributo	Tipo de dato	Comentario
id_alcance	INTEGER	Identificador de la tabla.
alcance	VARCHAR	Alcance que se inserta.

Tabla: t_asistencia		
Descripción: Tabla que guarda la asistencia		
Nombre atributo	Tipo de dato	Comentario
id_asistencia	INTEGER	Identificador de la tabla.
nombreyA	VARCHAR	Nombre y apellidos.
cargo	VARCHAR	Cargo que ocupa.

Tabla: t_auditoría		
Descripción: Tabla que guarda los datos de auditoría		
Nombre atributo	Tipo de dato	Comentario
id_auditoría	INTEGER	Identificador de la tabla.
id_fabricante	INTEGER	Identificador del fabricante.
tipo_aud	VARCHAR	Tipo de auditoría.
fecha_desde	DATE	Fecha de inicio.
fecha_hasta	DATE	Fecha final.
equipo_auditor	VARCHAR	Equipo auditor.
idioma	VARCHAR	Idioma de la auditoría.
clase_riesgo	VARCHAR	Clase de riesgo del fabricante.

guia	VARCHAR	Guía de la auditoría.
cargo	VARCHAR	Cargo del guía de la auditoría.
codigo	INTEGER	Código de la auditoría.

Tabla: t_aviso		
Descripción: Tabla que guarda los datos de los avisos		
Nombre atributo	Tipo de dato	Comentario
id_aviso	INTEGER	Identificador de la tabla.
id_rpc	INTEGER	Identificador de la tabla.
id_rapac	INTEGER	Identificador de la tabla.
id_rpa	INTEGER	Identificador de la tabla.
id_fabricante	INTEGER	Identificador de la tabla.
fecha_auditoría	DATE	Fecha de la auditoría.
dias_antelacion	INTEGER	Días de antelación.
estado	VARCHAR	Estado.
fecha_antelacion	DATE	Fecha de antelación.

Tabla: t_cert		
Descripción: Tabla que guarda los datos de la planilla LCCF para el fabricante certificado. Tiene los atributos de la tabla t_lista_chequeo		
Nombre atributo	Tipo de dato	Comentario
id_cert	INTEGER	Identificador de la tabla.
id_lista_chequeo	INTEGER	Identificador de la tabla t_lista_chequeo.
norma	VARCHAR	Norma.
obs_norma	VARCHAR	Observaciones de la norma.
organismo_cert	VARCHAR	Organismo que certifica.
obs_organismo_cert	VARCHAR	Observaciones del organismo que certifica.
coincidencia	VARCHAR	Verdadero si coincide el nombre con el de la certificación.
obs_coincidencia	VARCHAR	Observaciones de la coincidencia.
alcance_inst	VARCHAR	Alcance de las instalaciones.
obs_alcance_inst	VARCHAR	Observaciones del alcance de las instalaciones.
alcance_equ	VARCHAR	Alcance de los equipos.
obs_alcance_equipo	VARCHAR	Observaciones del alcance de los equipos.
vigencia	VARCHAR	Vigencia de la certificación.
obs_vigencia	VARCHAR	Observaciones de la vigencia.

Tabla: t_clase_riesgo		
Descripción: Tabla que guarda los datos de las clases de riesgos		

Nombre atributo	Tipo de dato	Comentario
Id_clase_riesgo	INTEGER	Identificador de la tabla.
Clase_riesgo	INTEGER	Clase de riesgo.

Tabla: t_criterio		
Descripción: Tabla que guarda los datos de criterio		
Nombre atributo	Tipo de dato	Comentario
id_criterio	INTEGER	Identificador de la tabla.
criterio	VARCHAR	Criterio.

Tabla: t_cronograma		
Descripción: Tabla que guarda los datos de cronograma		
Nombre atributo	Tipo de dato	Comentario
id_cronograma	INTEGER	Identificador de la tabla.
participantes	VARCHAR	Participantes.
evento	VARCHAR	Evento.
actividad	VARCHAR	Actividad.
fecha_crono	DATE	Fecha del cronograma.
sesion	VARCHAR	Sesión.

Tabla: t_fabricante		
Descripción: Tabla que guarda los datos de fabricante		
Nombre atributo	Tipo de dato	Comentario
id_fabricante	INTEGER	Identificador de la tabla.
nombre	VARCHAR	Nombre del fabricante.
numero_exp	INTEGER	Número de expediente.

Tabla: t_ia		
Descripción: Tabla que guarda los datos de la planilla IA		
Nombre atributo	Tipo de dato	Comentario
idt_ia	INTEGER	Identificador de la tabla.
id_auditoría	INTEGER	Identificador de la tabla t_auditoría.
instalaciones_visitadas	VARCHAR	Instalaciones visitadas.
gerencia	VARCHAR	Conclusiones de Gerencia.
documentacion_p	VARCHAR	Conclusiones de documentación del productor.
acciones	VARCHAR	Acciones.

documentacion_r	VARCHAR	Conclusiones de documentación del productor.
diseño_d	VARCHAR	Conclusiones del diseño.
produccion_c	VARCHAR	Conclusiones de la producción.
control_compras	VARCHAR	Conclusiones del control de compras.
conc_noconform	VARCHAR	Conclusiones de las conformidades.
comport_global	VARCHAR	Conclusiones del comportamiento global.
propone	VARCHAR	Propuestas.
visto_bueno	VARCHAR	Nombre del que da el visto bueno.
fecha	DATE	Fecha de confección.

Tabla: t_lista_chequeo		
Descripción: Tabla que guarda los datos de la planilla LCCF		
Nombre atributo	Tipo de dato	Comentario
id_lista_chequeo	INTEGER	Identificador de la tabla.
id_fabricante	INTEGER	Identificador del fabricante.
fecha_rev	DATE	Fecha de revisión.
clase_riesgo	VARCHAR	Clase de riesgo del fabricante.
conclusiones	CHAR	Conclusiones.
revisado_por	VARCHAR	Nombre del que revisa.

Tabla: t_nocert		
Descripción: Tabla que guarda los datos de la planilla LCCF para el fabricante no certificado. Tiene los atributos de la tabla t_lista_chequeo		
Nombre atributo	Tipo de dato	Comentario
id_nocert	INTEGER	Identificador de la tabla.
id_lista_chequeo	INTEGER	Identificador de la tabla t_lista_chequeo.
dec_prod	VARCHAR	Declaración del productor.
obs_dec	VARCHAR	Observación de la declaración.
plan_calidad	VARCHAR	Plan de Calidad.
obs_plan	VARCHAR	Observaciones de Plan de Calidad.
constancia	VARCHAR	Constancia.
obs_constancia	VARCHAR	Observaciones de la constancia.

Tabla: t_noconformidad		
Descripción: Tabla que guarda los datos de la no conformidad		
Nombre atributo	Tipo de dato	Comentario
id_noconformidad	INTEGER	Identificador de la tabla.
nombre	VARCHAR	Nombre de la no conformidad.
requisito_incumplido	VARCHAR	Requisito incumplido.

Tabla: t_objetivo_a

Descripción: Tabla que guarda los datos de los objetivos de la auditoría		
Nombre atributo	Tipo de dato	Comentario
id_objetivo_a	INTEGER	Identificador de la tabla.
objetivo	VARCHAR	Objetivo.

Tabla: t_observaciones

Descripción: Tabla que guarda los datos de los observaciones		
Nombre atributo	Tipo de dato	Comentario
id_observaciones	INTEGER	Identificador de la tabla.
observaciones	VARCHAR	Observaciones.

Tabla: t_pa

Descripción: Tabla que guarda los datos de la planilla Plan de la Auditoría		
Nombre atributo	Tipo de dato	Comentario
id_pa	INTEGER	Identificador de la tabla.
id_auditoría	INTEGER	Identificador de la tabla t_auditoría.
equipo_auditar	VARCHAR	Equipo a auditar.
instalaciones_visitar	VARCHAR	Instalaciones a visitar.
fecha	DATE	Fecha.

Tabla: t_ra

Descripción: Tabla que guarda los datos de la planilla Registro de Asistencia		
Nombre atributo	Tipo de dato	Comentario
id_ra	INTEGER	Identificador de la tabla.
id_auditoría	INTEGER	Identificador de la tabla t_auditoría.
fechaa	DATE	Fecha de apertura.
fechac	DATE	Fecha de cierre.

Tabla: t_rapac

Descripción: Tabla que guarda los datos del RAPAC		
Nombre atributo	Tipo de dato	Comentario
id_rapac	INTEGER	Identificador de la tabla.
id_auditoría	INTEGER	Identificador de la tabla t_auditoría.
revisiónPlan	DATE	Fecha de revisión del plan.

fechaAu	DATE	Fecha de la auditoría.
fechaEnt	DATE	Fecha de entrega.

Tabla: t_rpa		
Descripción: Tabla que guarda los datos del RPA		
Nombre atributo	Tipo de dato	Comentario
id_rpa	INTEGER	Identificador de la tabla.
id_fabricante	INTEGER	Identificador de la tabla t_fabricante.
numero	INTEGER	Número.
fechaAu	DATE	Fecha de la auditoría.
causasAu	VARCHAR	Causa de la auditoría.

Tabla: t_rpc		
Descripcion: Tabla que guarda los datos del RPC		
Nombre atributo	Tipo de dato	Comentario
id_rpc	INTEGER	Identificador de la tabla.
id_fabricante	INTEGER	Identificador de la tabla t_fabricante.
numero	INTEGER	Número.
fechaent	DATE	Fecha de entrega del plan.
revision	DATE	Fecha de revisión.
audpro	INTEGER	Auditorias propuestas.
audre	INTEGER	Auditorias realizadas.

GLOSARIO DE TÉRMINOS

CCEEM: Centro para el Control Estatal de Equipos Médicos.

CICEM: Centro de Ingeniería Clínica y Electromedicina.

CMMI: Capability Maturity Model Integration.

CNE: Centro Nacional de Electromedicina.

CPE: Centro Provincial de Electromedicina.

CRP: Capacity Resource Planning.

CSS: Cascading Style Sheets. Hojas de estilo que facilitan la creación de páginas.

CU: Caso de Uso.

CUN: Caso de uso del negocio.

CUS: Caso de uso del sistema.

DA: Diagrama de Actividad.

DD: Departamento de Documentación.

DOM: (Document Object Model) Interfaz de programación de aplicaciones que proporciona un conjunto estándar de objetos para representar documentos HTML y XML.

DS: Departamento de Supervisión.

ECP: Equipo de Calidad del Proyecto.

EF: Expediente del fabricante.

ERP: Enterprise Resource Planning es una herramienta de estrategia, que equipa a la empresa con las capacidades y recursos necesarios para integrar y sincronizar las funciones aisladas en un proceso continuo de negocios en miras de ganar una competitividad en el turbulento ambiente de los negocios.

Estilo: El estilo es el conjunto de características o cualidad que diferencian y distinguen una forma de escribir o representar objetos.

GADE: Grupo de Automatización y Desarrollo de Electromedicina.

HTML: HyperText Markup Language (*Lenguaje de Marcas de Hipertexto*).

HTTP: (Hypertext Transfer Protocol). Se encarga de mantenerse a la espera de peticiones HTTP llevada a cabo por un cliente HTTP que solemos conocer como navegador. El navegador realiza una petición al servidor y éste le responde con el contenido que el cliente solicita.

IA: Informe de la Auditoría. Planilla del Departamento de Supervisión.

IEEE: Institute of Electrical and Electronics Engineers.

INFOMED: es el Portal de Salud Cubano y la red de personas e instituciones que comparten el propósito de facilitar el acceso a la información de salud en Cuba.

Internet: Es una red mundial de computadoras interconectadas con un conjunto de protocolos, el más destacado, el TCP/IP. Aparece por primera vez en 1960.

ISO: Internacional Standard Organization.

ITCLPA: Instrucciones para la confección de la Lista Personalizada de la Auditoría.

Iter: Iteración.

JDS: Jefe del Departamento de Supervisión.

LCCF: Lista de Chequeo de Calidad para fabricante. Planilla del Departamento de Supervisión.

LMC: Aplicación de los Lineamientos mínimos de calidad.

LPA: Lista Personalizada de la Auditoría.

Mapa de Navegación: Grafo de nodos y relaciones que establece la estructura organizacional del sitio, donde los nodos son páginas y las relaciones constituyen los links o vías de acceso.

MGIDS: Módulo de gestión de la información del Departamento de Supervisión del CCEEM.

MIC: Ministerio de la Informática y las Comunicaciones.

MINSAP: Ministerio de Salud Pública.

MLPA: Matriz de Lista Personalizada de la Auditoría.

MRP: Material Requirement Planning. Planificación de requerimientos de materiales, sistema de planificación de la producción y de gestión de stocks (o inventarios).

Multihilos: Un hilo de ejecución, en Sistemas Operativos, es similar a un proceso en que ambos representan una secuencia simple de instrucciones ejecutada en paralelo con otras secuencias. Los hilos permiten dividir un programa en dos o más tareas que corren simultáneamente, por medio de la multiprogramación.

MVC: Modelo Vista Controlador.

OBE: Organizaciones Básicas de Electromedicina.

PA: Plan de la Auditoría.

PAC: Plan de Acciones Correctivas.

PPQA: Aseguramiento de calidad del producto y proceso.

RAPAC: Registro de Auditorías y Planes de Acciones Correctivas.

RN: Regla del negocio.

RPA: Registro Programa de Auditorías.

RPC: Registro Planes de Calidad.

RRHH: Recursos Humanos.

RUP: Rational Unified Process. El Proceso Unificado es un proceso de desarrollo de software (Conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema de software).

Rv: Revisiones.

SACCEM: Sistema Automatizado Cubano para el Control de Equipos Médicos.

Script: Guión o conjunto de instrucciones. Permite la automatización de tareas creando pequeñas utilidades. Se ejecuta del lado del cliente.

SEI: Software Engineering Institute.

SIGEM: Sistema de Gestión para la Electromedicina.

SIGICEM: Sistema de Gestión para la Ingeniería Clínica y Electromedicina.

SNS: Sistema Nacional de Salud.

SOA: Arquitectura Orientada Servicios.

SoftlaQ: Herramienta Informática para el Procesos de Pruebas. Beta del módulo Gestión de No Conformidades.

TIC: Tecnologías de la informática y las comunicaciones.

UCI: Universidad de Ciencias Informáticas.

UML: Lenguaje de Modelado Unificado (Unified Modeling Language).

Web Services: Servicio Web. Programa que implementa el protocolo.

XP: Extreme Programming.

YUI: Yahoo UI Library, se puede traducir como "Librería de componentes de interfaz de usuario de Yahoo" es un conjunto de utilidades y controles escritos en JavaScript para el desarrollo rápido y sencillo de aplicaciones Web complejas.