

Universidad de las Ciencias Informáticas

Facultad 7



**Título: Registro de Indicadores y Conductas
Asistenciales para la Atención Primaria de la Salud.**

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

Autores: Yenisbel Guerrero Dacal

Yasser Ibarra Castro

Tutores: Ing. Alexander Luisovich Paneque

Asesor: Dr. Miguel Marín Díaz

Ciudad de La Habana, Junio 2008

“Año 50 de la Revolución”

“Todos y cada uno de nosotros paga puntualmente su cuota de sacrificio consciente de recibir el premio en la satisfacción del deber cumplido, conscientes de avanzar con todos hacia el Hombre Nuevo que se vislumbra en el horizonte. ”

A handwritten signature in black ink, appearing to be the initials 'J.E.' in a cursive style.

DECLARACIÓN DE AUDITORÍA

DECLARACIÓN DE AUTORÍA

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

Para que así conste firmamos la presente a los 18 días del mes de Junio de 2008.

Yenisbel Guerrero Dacal

Firma del Autor

Yasser Ibarra Castro

Firma del Autor

Ing. Alexander Luisovich Paneque

Firma del Tutor

DATOS DE CONTACTO.

Dr. Miguel Eusebio Marín Díaz: Graduado de Médico en el año 1989 en el ISCMH y Especialista de I Grado en Medicina General integral en 1994 en el ISCMH. Posee categoría docente de Profesor Asistente de la Facultad de Ciencias Médicas Comandante Manuel Fajardo donde forma parte de la Cátedra de Medicina General Integral y de Informática Médica. Es Profesor Asistente Adjunto de la UCI. Ha impartido la asignatura de Segundo Perfil de Salud en la Facultad 7 desde el curso 2005-2006. Ha presentado ponencias en eventos científicos nacionales e internacionales. Se desempeña como Experto Funcional del MINSAP del Proyecto APS en la Empresa Softel. Su dirección de correo electrónico es marin@infomed.sld.cu.

Ing. Alexander Luisovich Paneque Meschenkov: Profesor graduado de Ingeniería en Ciencias Informáticas en la UCI en el curso 2006-2007. Actualmente labora en la misma universidad desempeñándose como profesor adiestrado de la disciplina Práctica Profesional la cual pertenece al Dpto. de Ingeniería de Software de la Facultad 7. Ha participado en eventos científicos en el centro. Su dirección de correo es apaneque@uci.cu.

AGRADECIMIENTOS.

Queremos agradecer:

A nuestro Comandante en Jefe y la Revolución, por hacer realidad nuestros sueños, dándonos la oportunidad de formarnos en una Universidad de futuro.

A la UCI, por poner a disposición todos los materiales y medios necesarios en nuestra formación y permitirnos crecer como profesionales dignos de la Revolución Cubana.

A nuestros familiares y amigos por darnos el apoyo y la confianza durante toda nuestra vida.

A nuestras amistades que siempre confiaron en que todo iba a resultar ser lo mejor posible.

A nuestros profesores por su quehacer constante y diario en nuestra enseñanza.

A todos los compañeros del proyecto y del grupo, que de una u otra forma aportaron su granito de arena en la elaboración de esta tesis.

En especial, a nuestros padres por guiarnos ejemplarmente por el camino más certero.

DEDICATORIA.

*Dedico este trabajo a toda mi familia y en especial a las personas que más amo en esta vida,
mis padres: Luis y Migdalia.*

A mi tía Carmen por su guía incondicional, amor, respeto y confianza en todo este tiempo.

A mi hermano, por su inmensa paciencia y aguantar todas mis malcriadeces.

*A mi “Piquete”, con las que sé que puedo contar en todo momento y a las cuales no voy a olvidar
nunca.*

A mi compañero de tesis Yasser (mi “Pare”), al cual admiro y respeto.

A todos los compañeros de estudio que compartieron junto a mí esta etapa tan importante de mi vida.

*A todas esas personas, que a pesar de la distancia, seguirán ocupando un lugar especial
en mi corazón.*

Yenisbel Guerrero Dacal.

*Dedico este trabajo a mis padres, ellos son mi ejemplo a seguir. Siempre confiaron en mí y han hecho
todo lo posible por ayudarme a ser lo que soy, para que viera realizados mis sueños.*

A mis abuelos Paulina y Homero por sus consejos y cariño, a la memoria de mi abuela Lidia.

A Lázaro, Leo, Juaco, gracias por ser mis mejores amigos...

A mis hermanos del alma Lismary y Alain, sin excepción a todos mis tíos y primos.

*A mi novia, amiga y compañera, Naty, por haber tenido tanta paciencia conmigo, por estar ahí
siempre que te necesito, un beso.*

A los socios del grupo, al Yonha y a Yosle, a mi compañera de tesis, Yenisbel, cariñosamente Pare.

*En general, a las amistades que me dieron fuerzas para seguir adelante por malos que
fuesen los tiempos, los que siempre estuvieron ahí deseándome éxitos.*

A todos los que en algún momento me preguntaron ¿Y la tesis?

Yasser Ibarra Castro.

RESUMEN

En la actualidad se desarrollan componentes que forman parte del Sistema de Información para la Salud (SISalud). En este se necesita de un componente que controle la información relacionada con los Indicadores del trabajo realizado por los Equipos Básicos de Salud así como las Conductas Asistenciales, de prevención, rehabilitación y educación para la salud que se efectúan en el marco de la Atención Primaria.

El presente trabajo persigue como objetivo desarrollar una aplicación web, que permita gestionar la información de los Indicadores de Salud y Conductas Asistenciales, contar con un servicio de búsqueda para las aplicaciones integradas al SISalud e imprimir dicha información.

Para el cumplimiento del objetivo trazado se hace uso de la tecnología XML Web Service y la arquitectura orientada a servicios y basada en componentes (SOA-CBA) para aplicaciones distribuidas. Se utilizan herramientas como el Zend Studio, el MySQL-Front y el Rational Rose. Para la documentación de dicho trabajo se utilizó la metodología de desarrollo RUP, con el lenguaje de modelado UML. Estas son utilizadas por las facilidades que brindan y porque responden a las políticas del MINSAP de utilizar sistemas abiertos, tecnologías basadas en INTERNET, entre otras.

Con la creación de este registro se tendrá un mejor manejo de los datos a gestionar, permitirá contar con información más actualizada, precisa y centralizada acerca de los Indicadores de Salud y las Conductas Asistenciales.

DECLARACIÓN DE AUTORÍA.....	I
DATOS DE CONTACTO.....	III
AGRADECIMIENTOS	III
DEDICATORIA.....	IIV
RESUMEN	V
Introducción	1
Capítulo1. Fundamentación Teórica.....	6
1.1 Sistema Nacional de Salud (SNS).	6
1.1.1 Informatización del Sistema Nacional de Salud.	7
1.1.2 Infomed.	8
1.1.3 Registro informatizado de Salud (RIS).	8
1.1.4 El Sistema de Información para la Salud (SISalud).....	9
1.2 Sistemas Informáticos para brindar información clasificada de indicadores y conductas asistenciales de la salud.	9
1.3 Técnicas, Tecnologías, Metodologías a considerar	11
1.3.1 Internet: TCP/IP.....	11
1.3.2 Servicios web:	12
1.3.3 Aplicación web:.....	12
1.3.4 Servidor web:.....	13
1.3.5 HTTP:.....	13
1.3.6 Arquitectura en capas.	13
1.3.7 Arquitectura basada en componentes.....	13
1.3.8 Arquitectura orientada a servicios (SOA).	14
1.3.9 Entornos distribuidos. Modelo Cliente/Servidor.....	14
1.3.10 Metodologías de Desarrollo.	14
1.4 Lenguajes de programación Web.	16
1.5 Plataforma de Servicios (PLASER).....	18
1.6 Sistema de Gestión de Base de Datos.	18
1.7 Herramientas utilizadas.	19
Capítulo2. Características del Sistema.....	19

2.1 Situación problemática	21
2.2 Modelo de Dominio	22
2.2.1 Conceptos Fundamentales	22
2.2.2 Diagrama del modelo de dominio	24
2.3 Propuesta del Sistema	24
2.3.1 Especificación de Requerimientos de Software	25
2.3.2 Modelo de Casos de Uso del Sistema	28
Capítulo 3: Análisis y Diseño del Sistema	33
3.1 Modelo de Análisis	33
3.1.2 Diagrama de Clases del Análisis	34
3.1.2 Diagramas de iteración	36
3.2 Modelo del Diseño	42
3.2.1 Justificación del uso de Patrones	42
3.2.2 Definición de la estructura del diseño	45
3.2.3 Diagramas de Clases del Diseño	46
3.2.4 Descripción de las clases y atributos	52
3.2.5 Diagrama de clases persistentes	61
3.2.6 Modelo de Datos	62
3.2.7 Descripción de tablas y atributos	62
Capítulo 4: Implementación y Pruebas	64
4.1 Modelo de Implementación	64
4.1.1 Diagramas de Componentes	65
4.1.2 Diagrama de Despliegue	66
4.2 Descripción de los Métodos	67
4.5 Prueba	77
4.5.1 Métodos de Prueba	77
Conclusiones	83
Recomendaciones	84
Referencias Bibliográficas	85

Bibliografía.....	87
Anexos.....	89
Glosario de Términos.....	92

Introducción.

La salud en Cuba es un tema del cual todos los cubanos se enorgullecen, se conocen todas las acciones, tanto económicas como médicas, que lleva a cabo la Revolución para la protección y atención médica gratuita de todos los ciudadanos cubanos.

A pesar de las afectaciones que ha sufrido el pueblo cubano, los éxitos conquistados durante más de 40 años en el terreno de la salud pública son conocidos y reconocidos en todo el planeta. No existe un solo rincón del archipiélago al margen de sus programas. La atención médica, por ejemplo, no solo es gratuita, sino también masiva. Enfermedades que en otros países son endémicas, aquí se han erradicado por completo. Es que el Estado cubano le confiere una alta prioridad a este sector, en tanto constituye uno de los principales indicadores para medir el desarrollo de una nación.¹

Cuba presenta un sistema para la atención a la salud altamente desarrollado, el cual se puede comparar con aquellos de los países más industrializados donde los indicadores de salud como la tasa de mortalidad infantil y la esperanza de vida son similares al de nuestro país. Este nivel es sorprendente en un país prácticamente aislado, que es punto de mira ante una constante amenaza de invasión junto con el continuo embargo y agresión económica.²

El Sistema Nacional de Salud (SNS) cubano con el espíritu revolucionario de cambiar todo lo que debe ser cambiado viene realizando importantes reformas.

La Atención Primaria de Salud (APS) es la piedra angular del Sistema Nacional de Salud de Cuba. Se caracteriza por su cobertura universal, equidad, eficiencia, accesibilidad y la presencia de un escenario docente de excelencia. Dispone de médicos y enfermeras de la familia, organizados en equipos de trabajo, que junto a otros profesionales constituyen "los guardianes de la salud de nuestra población"³.

¹ cubasalud. (29 de 11 de 2006). Recuperado el 20 de 12 de 2007, de <http://cubasalud.blogia.com>

² George Eisen, M. (dic. de 1996). *scielo.sld.cu*. Recuperado el 20 de 12 de 2007, de http://scielo.sld.cu/scielo.php?pid=s0864-34661996000200003&script=sci_arttext

³ (s.f.). Recuperado el 20 de 12 de 2007, de <http://aps.sld.cu/bvs/E/sobreaps.html>

En la era de la información y el conocimiento, la rapidez de los cambios tecnológicos ha obligado a las bibliotecas médicas a enfrentar nuevos retos, tanto para el desarrollo de las funciones tradicionales como de las surgidas con el desarrollo de las nuevas tecnologías de información y comunicación (NTIC) y en particular de Internet.

Hoy en día, Cuba tiene un amplio uso de las Tecnologías de la Informática y las Comunicaciones (TIC), se avanza en la informatización de su sistema de salud pública. Persiguiendo ante todo, elevar la calidad y eficiencia del servicio de salud para todo el pueblo. Así como crear una plataforma que permita incrementar el intercambio entre los especialistas y la interconectividad entre las diferentes unidades hospitalarias.

Para desarrollar estas tareas complejas, se han tomado como apoyo sistemas ya desarrollados en diferentes instituciones del Sistema de Salud Pública, los cuales gestionaban la información de forma aislada y no garantizaban diferentes aspectos como la disposición de información única y confiable para la toma de decisiones en los diferentes niveles de dirección, la integridad de la información y la interconexión entre las diferentes aplicaciones para poder lograr un flujo de información lógico.⁴

Actualmente el proyecto de informatización de la APS se ha encontrado con un camino propicio y con factores colaboradores para su implementación, por ejemplo: Existe por parte del gobierno una voluntad política para desarrollar la Salud Pública.

El desarrollo alcanzado por la Salud Pública se debe a la amplia cobertura y calidad con que se cuenta a nivel de Atención Primaria y el creciente desenvolvimiento que nuestro país va alcanzando en el campo de la Informática y las Comunicaciones formarán un sector mucho más desarrollado.

Dentro del proyecto de informatización de la APS, juega un rol importante la Empresa Productora de Software para la Salud SOFTEL, vinculada a estudiantes y profesores de la Facultad 7 de la Universidad de Ciencias Informáticas (UCI). Estas entidades, utilizan una eficaz estrategia para el desarrollo de software en el país, con la organización del proceso productivo a través de una eficiente gestión de requerimientos.

⁴ Turrueles Tejeda Yosvani, González Díaz Maikel. (2007). Registro de Ubicación Geográfica y Registro de Localidades para el Sistema de Información para la Salud. La Habana.

En la misma participan, desde un inicio, médicos y trabajadores de la salud, vinculados directamente a la APS en calidad de expertos funcionales en estrecho vínculo con los especialistas de Informática, a través de una metodología con la que se logró describir en Lenguaje Unificado de Modelado (UML) los procesos que se propusieron para automatizar. La experiencia de este trabajo debe constituir el inicio de buenas prácticas en la producción de software con alta calidad y un ejemplo de normativa para los proyectos que deben irse abriendo de ahora en adelante en la Informatización del Sector de la Salud.

Dentro del marco de este proyecto se desarrolla el Sistema de Información para la Salud (SISalud), que permitirá combinar la información de diferentes áreas de manera sencilla, pero eficiente; utilizando una arquitectura orientada a servicios y basada en componentes (SOA-CBA). Este sistema está compuesto por diferentes módulos en los diferentes niveles de atención médica, y por el Registro Informatizado de Salud (RIS) que incluye los distintos nomencladores y codificadores nacionales que brindan servicios a las demás aplicaciones informáticas, ejemplos de estos son: el Registro de la Clasificación Internacional de Enfermedades y Problemas relacionados con la Salud (RCIE), el Registro de Personal de la Salud para la Atención Primaria (RPS), y el Registro de Equipos Médicos (REM) , Registro de Equipos no Médicos; todos con el objetivo de obtener una información integral en tiempo real que facilite la toma de decisiones en todos los niveles de dirección.

Surge así la necesidad de contar en el Registro de Actividades Diarias del Sistema Nacional de Salud con un registro que brinde la posibilidad de gestionar la información de los Indicadores de la salud del trabajo realizado por los Equipos Básicos de Salud así como las Conductas Asistenciales o Procedimientos Médicos, de prevención, rehabilitación y educación para la salud que se efectúan en el marco de la Atención Primaria.

Luego de haber analizado la situación existente respecto a este tema y por la importancia en el apoyo al mejoramiento del trabajo de nuestros profesionales de la Salud, se puede establecer como **Problema a resolver** la siguiente interrogante: ¿Cómo gestionar la información clasificada referente a los Indicadores Básicos y a las Conductas Asistenciales, que permita el proceso de recopilación de los indicadores de salud y de los procedimientos médicos llevados a cabo en el marco de la APS?

Se define como **Objeto de Estudio**: El proceso de gestión de la información del Sistema Nacional de Salud.

El **Campo de Acción** se enfoca en el proceso de gestión de la información clasificada referente a los Indicadores del trabajo realizado por los Equipos Básicos de Salud y a las Conductas Asistenciales dentro de la Atención Primaria de Salud.

Por lo que el **Objetivo General de la Investigación** es: Desarrollar una aplicación web que facilite la gestión de la información clasificada de las Conductas Asistenciales y los Indicadores de salud en la Atención Primaria.

Para ello se proponen las siguientes **Tareas de la Investigación**:

- 1- Realizar entrevistas a los clientes para determinar los requisitos funcionales y no funcionales.
- 2- Identificar elementos de integración con otros componentes o sistemas del SNS o fuera de éste.
- 3- Puntualizar los procesos asociados a la gestión de los indicadores y conductas asistenciales para la salud en la atención primaria.
- 4- Asimilar la Arquitectura definida por el MINSAP (Orientada a Servicios y Basada en Componentes (OBA_CBA)) para el desarrollo de sus aplicaciones: Plataforma de Servicios PLASER y Registro Informatizado de Salud (RIS).
- 5- Modelar los artefactos de los flujos de trabajo: Modelamiento del Negocio, Gestión de Requerimientos, Diseño e Implementación.
- 6- Implementar los métodos de la Capa de Negocio y Capa de Presentación del Registro de Indicadores y Conductas.

El presente documento está estructurado en cuatro capítulos como se presentan a continuación:

Capítulo 1. Fundamentación Teórica: Se describen los conceptos fundamentales, asociados al dominio del problema, se exponen los antecedentes tanto nacionales como internacionales. Se hace un análisis crítico y valorativo de las soluciones existentes y la que se propone. Además, se explican y justifican las técnicas, tendencias y tecnologías a considerar, en las que se apoya la solución del problema.

Capítulo 2. Características del Sistema: Se describe la situación problemática existente y el flujo actual de los procesos a través de un Modelo de Dominio, el cual facilita el análisis necesario para determinar las funcionalidades del sistema a desarrollar. Estas se describen detalladamente mediante la especificación de requerimientos, descripción de casos de uso y representación gráfica.

Capítulo 3. Análisis y diseño del sistema: En este capítulo se describen los patrones de diseño a utilizar en el desarrollo de la aplicación, se definen la estructura y los elementos del diseño, se muestran los diagramas de clases de los casos de uso del sistema utilizando estereotipo web. Se presenta el diagrama de clases persistentes y el modelo de datos. Conformándose, finalmente, el Modelo de Diseño, el cual constituye el punto de partida para la implementación del sistema.

Capítulo 4 Implementación y prueba: En este capítulo se describe la integración con otros componentes. Se muestra el modelo de implementación con el diagrama de componentes y el diagrama de despliegue. Se muestra una descripción detallada de los métodos más complejos en el desarrollo de la aplicación o agentes en el proceso de implementación, así como los estándares de diseño, codificación y tratamientos de errores utilizados.

Capítulo 1. Fundamentación Teórica.

El objetivo de este capítulo es abordar los distintos aspectos que se utilizan como soporte teórico del sistema que se pretende diseñar. Se explica la estructura del Sistema Nacional de Salud (SNS), el Registro Informatizado de la Salud (RIS) y el Sistema de Información para la Salud (SISalud) como la nueva etapa de la informatización del Sistema Nacional de Salud; así como el estado del arte de las tecnologías, metodologías y herramientas a utilizar en la presente investigación.

1.1 Sistema Nacional de Salud (SNS).

El sistema creado (Sistema Nacional de Salud) comenzó a realizar importantes reformas a partir de los años 60, como parte fundamental de las transformaciones del período revolucionario.

El Sistema Nacional de Salud es un sistema universal, gratuito, accesible, integral, es un sistema de alcance de todos los ciudadanos: en el campo y en la ciudad, de cualquier raza, de cualquier sexo, religioso o ateo, con participación comunitaria e intersectorial, con una concepción internacionalista.

El sistema de salud cubano tiene muchas características nuevas y únicas. Refleja a la sociedad socialista cubana como un todo. Es subsidiado por el Estado y se basa en las necesidades de la población y no en las relaciones de solvencia financiera.

La garantía de atención médica gratuita a toda la población cubana, aún en los lugares más apartados, se convirtió desde los primeros momentos del triunfo de la Revolución en uno de los paradigmas sociales fundamentales y hoy toda la población goza de atención médica primaria permanente y remisión a servicios gratuitos de alta tecnología.

Entre los logros del sistema están en el incremento del parto institucional, reducción de la mortalidad infantil, disminución del bajo peso al nacer e incremento de la esperanza de vida y la reducción de los egresos hospitalarios en todas las provincias. Dado el concepto de atención preventiva existen numerosos programas de estudios y pesquiasaje que tienen un carácter masivo y movilizan importantes recursos materiales y humanos, con el apoyo de las organizaciones sociales de las comunidades.⁵

⁵ Felipe, d. J. (08 de 01 de 2008). Cuba: Salud de privilegios en el Siglo XXI. Recuperado el 04 de 02 de 2008, de [www.cubasocialista.cu: http://www.cubasocialista.cu/texto/0008110salud.html](http://www.cubasocialista.cu/texto/0008110salud.html)

El SNS en Cuba está estructurado en tres niveles de dirección en correspondencia con los niveles administrativos del país, Nacional, Provincial y Municipal, representados por el MINSAP, que ha definido la incorporación de los últimos adelantos en el área de las tecnologías de la información y las comunicaciones, incluyendo el empleo de estándares internacionales para los productos relacionados con la salud.⁶

La atención primaria de salud(APS), comprende un conjunto de procederes y servicios de promoción, prevención, curación y rehabilitación, así como la protección de grupos poblacionales específicos y el abordaje de problemas de salud con tecnologías apropiadas de este nivel, dirigido a cada persona, sus familias, la comunidad y el medio ambiente. Tal atención primaria se relaciona estrechamente con la atención médica secundaria especializada y con la terciaria altamente especializada en los policlínicos y los hospitales, donde las avanzadas mediciones de diagnóstico se llevan a cabo junto con los complicados tratamientos quirúrgicos y médicos.⁷

La Atención Secundaria es brindada a nivel de las instituciones hospitalarias, generalmente son de carácter provincial, es decir es atendida toda la población de una provincia determinada. El paciente tiene acceso a esta atención a través de una remisión del personal médico de la atención primaria o sin ella, asistiendo directamente la persona necesitada de atención médica.

La Atención Terciaria es la atención que por su condición muy especializada, sólo se brinda en determinados centros, ejemplo: Instituto de Gastroenterología, Instituto de Cirugía Cardiovascular, Instituto de Neurocirugía, Instituto de Nefrología, entre otros o en centros hospitalarios y/o de investigación categorizados como centros de referencia nacional y en algunos casos de referencia internacional.

1.1.1 Informatización del Sistema Nacional de Salud.

La informatización del Sistema Nacional de Salud Pública (SNS) está dada por el conjunto de métodos, técnicas, procedimientos y actividades gerenciales dirigidas al manejo de la información en Salud, la cual comprende la información sobre el estado de Salud de la población, la información sobre el conocimiento

⁶ Marin Sanchez, J., & Acosta Montejo, C. (2007). *Registro de la Clasificación Internacional de Enfermedades y Problemas relacionados con la Salud (RCIE) del Sistema de Información para la Salud*. Ciudad de la Habana: Univesidad de las Ciencias Informáticas(UCI).

⁷ Ídem Referencia 5.

de las ciencias de la Salud y la información en general para la toma de decisiones, clínico-epidemiológicas, operativas y estratégicas.⁸

Este proceso busca lograr más eficacia y eficiencia, que permitan una mayor generación de riquezas y hagan sustentable el aumento sistemático de la calidad de vida de los ciudadanos.

Actualmente el Ministerio de Salud Pública (MINSAP) ha definido a la informatización como una de sus prioridades y ha convocando para ello a un grupo de instituciones propias del sector, del Ministerio de Informática y Comunicaciones y de otros organismos de la administración central del estado, para definir de conjunto la estrategia a desarrollar. En algunos casos se han tomado como punto de partida sistemas ya desarrollados en el país en el marco de aquella primera estrategia de desarrollo en 1997.⁹

1.1.2 Infomed.

La Red Telemática de la Salud INFOMED une todas las unidades y trabajadores del sistema sanitario nacional. Con un amplio uso de las Tecnologías de la Informática y las Comunicaciones (TIC), surgida en 1992, brinda servicios de mensajería electrónica, universidad virtual, biblioteca virtual, cursos a distancia, acceso a bases de datos de diferentes materias, entre otros, la cual constituye una infraestructura que favorece la informatización con nodos en todas las provincias y donde trabajan expertos de nivel internacional en la creación de productos y servicios electrónicos de información y el diseño de software, que son capaces de orientar proyectos con soluciones innovadoras para el sistema nacional de información de salud, y asesorar a los centros de información regionales en la introducción de las Nuevas Tecnologías de la Informática y las Comunicaciones (NTIC).¹⁰

1.1.3 Registro informatizado de Salud (RIS).

El Ministerio de Salud Pública de la República de Cuba (MINSAP), a partir del año 2003 ha definido como una de sus prioridades la informatización.

⁸ Delgado Ramos, A., & Vidal Ledo, M. (2006). Recuperado el 18 de 12 de 2007, de http://bvs.sld.cu/revistas/spu/vol32_3_06/spu15306.htm

⁹ Ídem Referencia 8

¹⁰ E. Marín Díaz, D. M. (2004). *Revista Habanera de Ciencias Médicas*. Recuperado el 20 de 01 de 2008, de http://www.ucmh.sld.cu/rhab/editorial_rev10.htm

El Registro Informatizado de la Salud es la solución propuesta por el MINSAP para la Informatización de la Salud Pública. Más que un software, es una plataforma de aplicaciones, abierta, con una interfaz de programación que permite incorporar nuevos módulos que sean compatibles con ella. Es portable a diferentes sistemas operativos, tanto en los servidores como en los clientes y replicable en otros entornos. Es una plataforma en constante desarrollo, que crece en la medida en que se implementan nuevos módulos.

Por definición el RIS es la solución informática integral para la Salud Pública, acorde con los objetivos de la informatización de la sociedad cubana. Constituido por un conjunto de aplicaciones independientes (módulos del sistema) que se interconectan según las necesidades del flujo de información. Es además la herramienta que permite a los usuarios autorizados combinar la información de los diferentes módulos que lo componen, para obtener una información integral en tiempo real para la toma de decisiones en los diferentes niveles de dirección, la docencia, investigación y la gestión en salud.¹¹

1.1.4 El Sistema de Información para la Salud (SISalud).

El Sistema de Información para la Salud (SISalud) es la propuesta que hace la Empresa de Soluciones Informáticas Softel para dar continuidad al proceso de informatización del Sistema Nacional de Salud (SNS). Esta nueva aplicación permitirá organizar e integrar los componentes ya desarrollados y los que se van a desarrollar como parte de la informatización del SNS. SISalud estará compuesto por el Registro Informatizado de Salud (RIS), el Sistema Automatizado de Atención Primaria (SIAP), Sistema Informatizado de Gestión Hospitalaria (SIGH) y el Sistema Informatizado de Atención Especializada (SIAE).¹²

1.2 Sistemas Informáticos para brindar información referente a los indicadores y conductas asistenciales de la salud.

Actualmente existen soluciones a nivel internacional y nacional que no resuelven el problema planteado. Entre ellas se encuentran:

¹¹ Delgado Ramos, D. A., Cabrera Hernández, I. M., & Juncal, D. V. (2006). *Registro Informatizado de Salud (RIS)*. Ciudad de la Habana: Ministerio de Salud Pública.

¹² Ídem Referencia 4.

Antecedentes Nacionales.

➤ Codificador Internacional de Enfermedades automatizado:

Trabajo de Diploma, Autora: Mariela Peris Cid

El objetivo fundamental del trabajo es la automatización del Codificador Internacional de Enfermedades.

Esta automatización trae consigo el diseño de dos bases de datos fundamentales: una que contiene la información de todos los diagnósticos que se conocen y otra con la información de los procedimientos quirúrgicos y no quirúrgicos existentes.¹³

Antecedentes Internacionales.

➤ Medical Control 2. Entre otras particularidades cuenta con:

- *Historia Clínica del paciente configurable para cada médico*

La HC incluye:

- Historial de procedimientos, recetas, exámenes, radiografías.
- **Agenda de citas con fotografía** (configurable para cualquier cantidad de médicos, asistentes, terapeutas o enfermeras.)
- **Administración del consultorio:** Lleva el control de pagos, abonos, cuentas por cobrar y facturas.
- Permite obtener **reportes** como Ingresos por procedimiento, por origen (aseguradora o filial del paciente), por asistente o terapeuta, por fechas, y muchos más al instante.
- **Resumen de H.C.:** Para cada paciente podrá visualizar un resumen por fechas de la historia clínica, incluyendo motivo de ingreso, la evolución de las consultas, los procedimientos realizados (terapias, cirugías, etcétera), las vacunas aplicadas, radiografías, imágenes o estudios digitalizados, documentos, etcétera.¹⁴

➤ DasiClinic:

¹³ Peris, M. (mayo-agosto de 1997). *Trabajos de Diploma*. Recuperado el 20 de enero de 2008, de Informatica Médica: http://bvs.sld.cu/revistas/aci/vol5_2_97/aci06297.htm

¹⁴ *JagarSoft*. (abril de 2008). Recuperado el enero de 2008, de JagarSoft: <http://www.jagarsoft.com/>

Desarrollado por Dasi Informática s.l., que es una empresa de servicios especializada en el desarrollo de aplicaciones informáticas.

DasiClinic proporciona una Completa gestión de agendas por profesional, Historia clínica del paciente, Control de tratamientos y procedimientos, Gestión de antecedentes médicos y patologías, Facturación, Presupuestos, Domiciliación de cuotas bancarias, Contabilidad, Creación y Gestión de Cuestionarios de Salud e Informes.¹⁵

Análisis Crítico de los Antecedentes.

Estas soluciones son muy globales, es decir ninguna puntualiza en el trabajo con los indicadores de la salud y los procedimientos médicos, además son aplicaciones de escritorio, con arquitecturas monolíticas, por tanto la información no se encuentra centralizada de manera que pueda ser accesible desde cualquier lugar. Además en los sitios de Salud Cubana, aunque la información esté centralizada, no permite actualizaciones, búsquedas, ni brinda servicios a las demás aplicaciones de la red de salud.

1.3 Técnicas, Tecnologías, Metodologías a considerar.

1.3.1 Internet: TCP/IP.

Internet nació a finales de los años 70 en los Estados Unidos. Es un amplísimo conjunto de ordenadores repartidos por todo el planeta, que se hallan interconectados entre sí a modo de una inmensa Red que se caracteriza por tener un ámbito mundial, a través de distintas líneas de intercambio de datos (las telefónicas, la Fibra Óptica (El Cable), Vía Satélite, teléfonos móviles, ordenadores portátiles, la Red Eléctrica, las Ondas de Radio, la Televisión).

Los ordenadores se entienden entre sí, independientemente de su configuración y sistema operativo, gracias a los protocolos como TCP/IP. Hay que tener en cuenta que en Internet se encuentran conectados ordenadores de clases muy diferentes y con hardware y software incompatibles en muchos casos, además de todos los medios y formas posibles de conexión. Aquí se encuentra una de las grandes

¹⁵ *Dasi Informática*. (2007). Recuperado el diciembre de 2007, de Dasi Informática: <http://www.dasi.es/content/view/2/4/lang.es/>

ventajas del TCP/IP, pues este protocolo se encargará de que la comunicación entre todos sea posible. TCP/IP es compatible con cualquier sistema operativo y con cualquier tipo de hardware.¹⁶

El constante incremento del número de host y del requerimiento de ancho de banda, la necesidad de contar con un nivel mayor de seguridad y un espacio de direcciones suficiente ha motivado un cambio en el protocolo TCP/IP evidenciado en el desarrollo y la aceptación de protocolo IPv6, como el protocolo para la nueva era de Internet. IPv6 brinda direcciones de 128 bits, así como encabezados flexibles, por lo tanto un número casi infinito de direcciones IP. El mismo se aplica sobre Internet2, el cual es un proyecto que nació entre las universidades de EEUU y tiende a extenderse a todo el mundo.

Algunos factores que deben resultar catalizadores del proceso, son los avances obtenidos en su adopción y desarrollo en Asia y Europa, la celebración de los Juegos Olímpicos en Beijing, China (bautizados como Olimpiada Digital, basada puramente en IPv6), y la exigencia de que todas las agencias federales norteamericanas estén corriendo el protocolo IPv6 en ese año.

Cuba planea la incorporación de estas nuevas tendencias dentro de unos años. Con estos cambios se podrán alcanzar una gran variedad de novedosos servicios como Teleinmersión, Laboratorios Virtuales o Bibliotecas Digitales. El módulo que se desarrolla esta basado en IPv4 debido a que el framework designado por la empresa Softel, PLASER, trabaja hoy en día sobre IPv4.

1.3.2 Servicios web: es una colección de protocolos y estándares que sirven para intercambiar datos entre aplicaciones. Las aplicaciones de software desarrolladas en distintos lenguajes de programación, y ejecutadas sobre cualquier plataforma, pueden utilizar los servicios web para intercambiar datos en redes de ordenadores como Internet.

1.3.3 Aplicación web: es un sistema informático que los usuarios utilizan accediendo a un servidor web a través de Internet o de una intranet. Las aplicaciones web son populares debido a la practicidad del navegador web como cliente ligero.

¹⁶ Alejandro Soto, M. (s.f.). Recuperado el 07 de 02 de 2008, de <http://usuarios.lycos.es/janjo/janjo1.html>

1.3.4 Servidor web: un servidor web es un programa que sirve para atender y responder a las diferentes peticiones de los navegadores, proporcionando los recursos que soliciten usando el protocolo HTTP o el protocolo HTTPS.

1.3.5 HTTP: Es un protocolo de transferencia de hipertexto (HyperText Transfer Protocol). Es un protocolo de comunicaciones utilizado por los programas clientes y servidores de WWW para comunicarse entre sí.

1.3.6 Arquitectura en capas.

Las aplicaciones modernas por lo general utilizan una arquitectura basada en capas (dos capas, tres capas, etc.).

El módulo RICAPS se desarrolló siguiendo una arquitectura de tres capas y una forma de comunicación donde las capas precisadas de superior a inferior son:

- *Capa de Presentación:* Contiene todos lo relacionado con las interfaces de interacción con los usuarios. En general incluye el manejo y vista de las ventanas, así como sus funcionalidades.
- *Capa de Negocio (o Acceso a Datos):* Contiene los elementos o métodos que automatizan los procesos del negocio que se ejecutan desde el cliente.
- *Capa de Datos (o Repositorio):* Contiene los datos persistentes de la aplicación agrupados en tablas de la base de datos.

1.3.7 Arquitectura basada en componentes.

La arquitectura de una aplicación basada en componentes consiste en uno o más componentes específicos de la aplicación (que se diseñan específicamente para ella), estos a su vez hacen uso de otros componentes que se conectan entre sí para suministrar los servicios que se necesitan en la aplicación. Un elemento básico de interconectividad lo constituye la interfaz donde cada componente debe describir de forma completa las interfaces que ofrece, así como las interfaces que requiere para su operación.

Esta técnica de software es importante para mejorar la calidad, disminuir los tiempos de desarrollo y gestionar la creciente complejidad de los sistemas.

1.3.8 Arquitectura orientada a servicios (SOA).

En las Arquitecturas Orientadas a Servicios, el elemento básico es el servicio. El concepto de Arquitectura Orientada a Servicios (Service-Oriented Architecture), define el uso de servicios para brindar soporte a los diferentes requerimientos de software por parte del usuario. Esta arquitectura de software permite la creación de procesos de negocio y realizar cambios de los mismos. La arquitectura SOA aporta algunos beneficios como son: Mejora en los tiempos de realización de cambios en procesos, fácil integración de tecnologías disímiles.

1.3.9 Entornos distribuidos. Modelo Cliente/Servidor.

La arquitectura Cliente-Servidor es un modelo para el desarrollo de sistemas de información en el cual las transacciones se dividen en procesos independientes que cooperan entre sí para intercambiar información, servicios o recursos. Se denomina cliente al proceso que inicia el diálogo o solicita los recursos y servidor al proceso que responde a las solicitudes. En este modelo las aplicaciones se dividen de forma que el servidor contiene la parte que debe ser compartida por varios usuarios, y en el cliente permanece sólo lo particular de cada usuario. Este modelo proporciona una gran ventaja y es que el servidor no necesita tanta potencia de procesamiento, se reduce el tráfico de red considerablemente ya que el cliente se conecta al servidor solo cuando es necesario, adquiere los datos que necesita y cierra la conexión dejando la red libre.

1.3.10 Metodologías de Desarrollo.

Lenguaje Unificado de Modelado (UML): UML es ante todo un lenguaje. Un lenguaje proporciona un vocabulario y reglas para permitir una comunicación. En este caso, este lenguaje se centra en la representación gráfica de un sistema.

Se pueden sintetizar las siguientes funciones:

- Visualizar: UML permite expresar de una forma gráfica un sistema de forma que otro lo puede entender.
- Especificar: UML permite especificar cuáles son las características de un sistema antes de su construcción.
- Construir: A partir de los modelos especificados se pueden construir los sistemas diseñados.

- Documentar: Los propios elementos gráficos sirven como documentación del sistema desarrollado que pueden servir para su futura revisión.¹⁷

Metodología RUP.

En un proyecto de desarrollo de software la metodología define Quién debe hacer Qué, Cuándo y Cómo debe hacerlo. Una metodología es un proceso. No existe una metodología de software universal. Las características de cada proyecto exigen que el proceso sea configurable.

La metodología RUP, llamada así por sus siglas en inglés **Rational Unified Process**, divide en 4 fases el desarrollo del software:

- Inicio, El Objetivo en esta etapa es determinar la visión del proyecto.
- Elaboración, En esta etapa el objetivo es determinar la arquitectura óptima.
- Construcción, En esta etapa el objetivo es llevar a obtener la capacidad operacional inicial.
- Transmisión, El objetivo es llegar a obtener el release del proyecto.

Cada una de estas etapas es desarrollada mediante el ciclo de iteraciones, la cual consiste en reproducir el ciclo de vida en cascada a menor escala. Los objetivos de una iteración se establecen en función de la evaluación de las iteraciones precedentes.

Vale mencionar que el ciclo de vida que se desarrolla por cada iteración, es llevada bajo dos disciplinas:

Disciplina de Desarrollo.

Ingeniería de Negocios: Entendiendo las necesidades del negocio.

Requerimientos: Traslado de las necesidades del negocio a un sistema automatizado.

Análisis y Diseño: Traslado de los requerimientos dentro de la arquitectura de software.

Implementación: Creando software que se ajuste a la arquitectura y que tenga el comportamiento deseado.

Pruebas: Asegurándose que el comportamiento requerido es el correcto y que todo lo solicitado está presente.

¹⁷ Orallo, E. H. (2008). *El Lenguaje Unificado de Modelado(UML)*. Recuperado el enero de 2008, de <http://www.disca.upv.es/enheror/pdf/ActaUML.PDF>

Disciplina de Soporte.

Configuración y administración del cambio: Guardando todas las versiones del proyecto.

Administrando el proyecto: Administrando horarios y recursos.

Ambiente: Administrando el ambiente de desarrollo.

Distribución: Hacer todo lo necesario para la salida del proyecto

Es recomendable que a cada una de estas iteraciones se les clasifique y ordene según su prioridad, y que cada una se convierte luego en un entregable al cliente. Esto trae como beneficio la retroalimentación que se tendría en cada entregable o en cada iteración.

Los elementos del RUP son:

- Actividades: Son los procesos que se llegan a determinar en cada iteración.
- Trabajadores: Son las personas o entes involucrados en cada proceso.
- Artefactos: Un artefacto puede ser un documento, un modelo, o un elemento de modelo.

Una particularidad de esta metodología es que, en cada ciclo de iteración, se hace exigente el uso de artefactos, siendo por este motivo, una de las metodologías más importantes para alcanzar un grado de certificación en el desarrollo del software.¹⁸

1.4 Lenguajes de programación Web.

PHP.

PHP, se conoce originalmente como Personal Home Pages, fue concebido en el otoño de 1994 por Rasmus Lerdorf. La primera versión salió en los comienzos de 1995.

Es un lenguaje de propósito general. PHP (HyperText Preprocessor) es un lenguaje "del lado del servidor" (esto significa que PHP funciona en un servidor remoto que procesa la página Web antes de que sea abierta por el navegador del usuario) especialmente creado para el desarrollo de páginas Web dinámicas.

¹⁸ Sanchez, I. M. (15 de abril de 2008). *Informatizate*. Recuperado el 15 de abril de 2008, de Informatizate: http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_software_07062004.html

Puede ser incluido con facilidad dentro del código HTML, y permite una serie de funcionalidades tan extraordinarias que se ha convertido en el predilecto de millones de programadores en todo el mundo. Combinado con la base de datos MySQL, es el lenguaje estándar a la hora de crear sitios de comercio electrónico o páginas Web dinámicas.

Java Script.

Es un lenguaje de programación del lado del cliente, porque es el navegador el que soporta la carga de procesamiento. Es el lenguaje de programación del lado del cliente más utilizado, debido a su compatibilidad con la mayoría de los navegadores modernos. Con JavaScript se pueden crear efectos especiales en las páginas y definir interactividades con el usuario. El navegador del cliente es el encargado de interpretar las instrucciones JavaScript y ejecutarlas para realizar estos efectos e interactividades, de modo que el mayor recurso, y tal vez el único, con que cuenta este lenguaje es el propio navegador.

XML.

El intercambio de Información siempre ha sido un grave problema cuando se utilizan lenguajes y sistemas operativos incongruentes, este problema solo se agudizó con la aparición de Internet; inclusive previa aparición de Internet fueron creados mecanismos para lograr el intercambio fluido de información entre diferentes sistemas, el primer método fue GML, posteriormente SGML y actualmente XML, todos estos mecanismos son llamados lenguajes de marcación o meta-lenguajes.

XML (Extensible Mark-up Language) es, más que un lenguaje de marcado, un meta-lenguaje, que permite definir lenguajes de marcado adecuados a usos determinados.

Es un subconjunto de SGML simplificado y adaptado a Internet.

XML no es una versión mejorada de HTML ni es un lenguaje para hacer mejores páginas Web. Es un estándar internacionalmente reconocido, que no pertenece a ninguna compañía y su uso es libre; permite el uso efectivo de Internet en diferentes alfabetos, y en diferente hardware (teléfonos celulares, etc.) Por demás XML no es difícil, es fácilmente procesable tanto por humanos como por software. Es legible y entendible por las computadoras.

XSLT (Extensible Stylesheet Lenguaje Transformation): es un lenguaje universal de transformación de documentos, éste permite definir las variadas presentaciones, además de ser muy rápido a la hora de procesar los documentos. La unión de XML y XSLT permite separar el contenido y su presentación.

1.5 Plataforma de Servicios (PLASER).

La Plataforma de Servicio (PLASER), es una plataforma sobre la que se pueden desarrollar aplicaciones XML Web Services. Está conformada principalmente por varias clases en PHP, una librería, que puede ser usada para que un componente se integre al Sistema de Información para la Salud (SISalud) garantizando la seguridad. Aunque PLASER solo soporta como llamada RPC (Remote Procedure Call) el protocolo SOAP, en futuras versiones se piensa en otros protocolos de transportes o incluso el acceso local a código a nivel de File System, de forma tal que para el programador es totalmente transparente si la invocación del proceso es remoto, local, por SOAP, directamente a código, etc. En su concepción se han utilizado estándares actuales y normas abiertas lo cual facilita la programación y similitud de los componentes.

Este sistema esta concebido completamente sobre Arquitectura Basada en Componentes y Orientada a Servicios, usando el paradigma de XML Web Services específicamente SOAP. PLASER desde el punto de vista estructural permite trabajar con cualquier base de datos que cumpla con la norma SQL ANSI 92; pero desde el punto de vista de implementación solo trabaja con las bases de datos soportadas por el componente DBX, ya que encapsula a dicho componente y lo utiliza para el acceso a bases de datos.

PLASER en su estructura encapsula varios componentes:

XSL: Extensión para transformaciones de ficheros XML.

Pear SOAP: Biblioteca para la comunicación SOAP.

DBX: Extensión para la abstracción del acceso a datos

1.6 Sistema de Gestión de Base de Datos.

MySQL es la base de datos relacional de código abierto más usada en el mundo. Código abierto significa que todo el mundo puede acceder al código fuente, es decir, al código de programación de MySQL. Todo el mundo puede contribuir para incluir elementos, arreglar problemas, realizar mejoras o sugerir

optimizaciones. MySQL ha pasado a ser una completa herramienta y ha conseguido superar a una gran cantidad de bases de datos comerciales. Su rápido desarrollo se debe a la contribución de mucha gente al proyecto, así como a la dedicación del equipo de MySQL. Gran portabilidad entre sistemas, es una de las principales ventajas que proporciona, pues puede trabajar en distintas plataformas y sistemas operativos, permite además que se realicen consultas a diferentes bases de datos de forma simultánea.

1.7 Herramientas utilizadas.

Rational Rose Enterprise Edition 2003 es utilizado para respaldar la documentación y UML (Unified Modeling Language) como lenguaje de modelado para la confección de los diagramas. Esta herramienta permite modelar complejas relaciones entre los objetos y es una de las más poderosas herramientas de modelado visual para el análisis y diseño de sistemas basados en objetos. Se utiliza para modelar un sistema antes de proceder a construirlo. Cubre todo el ciclo de vida de un proyecto: concepción y formalización del modelo, construcción de los componentes, transición a los usuarios y certificación de las distintas fases.

Stylus Studio 5.1 es utilizado para crear los ficheros XSL a través de escenarios XML. Stylus Studio 5.1 agrega una nueva y poderosa funcionalidad y facilidad de uso al galardonado Ambiente Integrado de Desarrollo XML que simplifica la programación en XML incrementando la productividad de los desarrolladores a través de la innovación.

Stylus Studio 5.1 permite la integración de Bases de Datos Relacionales con servicios Web, lo que habilita a las organizaciones para ofrecer una serie de servicios que van desde consultas hasta comercio electrónico a través de Internet.

Una de las características más poderosas de la Versión 5.1 es el Editor de Código Fuente entre Bases de Datos y XML. Stylus Studio permite a sus usuarios conectarse en forma directa a cualquier Base de Datos y utilizar la información contenida en éstas como XML.

Zend Studio 5.0 será usado para la edición del código PHP, esta herramienta es uno de los mejores IDE del momento. Este editor brinda la posibilidad de depurar el código y facilitar las diferentes tareas necesarias en el desarrollo de cualquier tipo de aplicación.

Servidor Web Apache 2.0: Este es un Servidor Web con capacidad de procesar páginas dinámicas php con acceso a base de datos MySQL. Es un claro ejemplo de lo que se puede conseguir con el código libre y el

modelo de desarrollo colaborativo del software abierto. La combinación de tres herramientas libres como Apache, PHP y MySQL, son el centro de millones de sitios web dinámicos.

EMS SQL Manager 2005: es una aplicación de alto desempeño para la administración y desarrollo de servidores de bases de datos MySQL. El programa trabaja con cualquier versión de MySQL desde la 3.23 hasta la 5.06 y soporta todas las últimas características de MySQL, incluyendo espacios de tablas, nombres de argumentos en funciones y más. Su interfaz gráfica es sumamente atractiva, de fácil uso, garantiza la administración y navegación rápida de bases de datos

Capítulo 2. Características del Sistema.

En este capítulo se dará una explicación de la situación problemática existente y de los conceptos fundamentales que se trataron en el desarrollo del modelo del dominio, modelo que sirve para especificar las condiciones, capacidades y cualidades que el sistema debe tener. Se presentan además los requisitos funcionales y no funcionales, así como modelo de casos de uso del sistema con las descripciones textuales de cada uno de los casos de uso.

2.1 Situación problemática.

Con el impacto social del creciente desarrollo de la informática, son cada vez más las instituciones y organizaciones que se inclinan por incorporar aplicaciones que gestionen su información, para de esta forma obtener una dinámica superior en sus respectivos procesos de negocio.

La incorporación de sistemas que brinden Información de la Salud en la gestión de de la información incrementa así la calidad de la información y se facilita la extracción de la misma.

Uno de los principales propósitos y objetivo primarios de un Sistema de Información de la salud es proveer de información a diferentes niveles en la estructura de la organización y estar informados del grado de importancia y eficacia de sus acciones.

En el mundo existen clasificaciones internacionales referentes a los indicadores de salud y los procedimientos médicos, esto resuelve que las estadísticas sean homogéneas de forma global y que el manejo de esta información sea más entendible y seguro.

La clasificación referente a estos indicadores se encuentra contenida en el sistema de información estadística complementaria (SIE-C) y la información consecuente con los procedimientos médicos en la clasificación internacional de enfermedades versión 9(CIE-9).

En nuestro país se cuenta con pocas soluciones informáticas en varios formatos, pero estos no son extensibles a toda la comunidad médica, existen libros y folletos que constan con dicha información, pero estos materiales no pueden ser reproducidos por nuestro país, es necesario comprarlos puesto que están amparados por el protocolo 2 de la Convención Universal del Derecho de Autor. Esto demuestra que hay poco acceso a dicha información por parte del personal médico y de la población, provocando un difícil estudio del tema para los estudiantes de la carrera de medicina.

Se le suma a todo esto que en la Atención Primaria de Salud la definición y clasificación de nuevos indicadores y conductas dependen en gran medida de la evaluación del entorno sanitario (capacidad de análisis) en que se desarrolla. Al ser esta actividad tan dinámica adquiere vital importancia la gestión de su información, sin embargo este proceso no cuenta con las ventajas de una solución para dicha gestión, ocasionando que el proceso sea mas lento y difícil pues constituye una considerable carga manual.

En el proceso de consultas a pacientes, realizadas en el marco de la APS, es necesario registrar todo el flujo de atención llevado a cabo por los EBS, desde el instante en que el paciente llega al consultorio. Todas estas operaciones son archivadas de forma manual razón por la cual la información tiende a volverse poco uniforme y la interpretación de esta se hace difícil.

Para dar solución a esta problemática se implementó el Registro Informatizado de Actividades Diarias, con el fin de hacer un procesamiento más eficiente y uniforme de la gestión de la información; donde se contemplarían datos tanto del paciente como de los profesionales que intervienen en este proceso. Además se necesita para satisfacer este proceso la obtención de la información clasificada de las conductas, accionar o procedimientos médicos y los indicadores de salud necesarios para dar seguimiento y comunicar la salud y el bienestar de los ciudadanos.

2.2 Modelo de Dominio.

Un modelo del dominio es una representación de las clases conceptuales del mundo real, no de componentes software. No se trata de un conjunto de diagramas que describen clases software, u objetos software con responsabilidades, el modelo del dominio podría considerarse como un diccionario visual de las abstracciones relevantes, vocabulario del dominio e información del dominio. Aprovechando las bondades de los diagramas UML para representar cosas y conceptos el diagrama del modelo del dominio se presenta en forma de diagrama de clases donde figuran los principales conceptos y roles del sistema analizado.

2.2.1 Conceptos Fundamentales.

Conceptos fundamentales del Registro de Indicadores y Conductas.

Indicadores: Los Indicadores de salud son instrumentos de evaluación que pueden determinar directa o indirectamente modificaciones dando así una idea del estado de situación de una condición. Son

definidos, según la Organización Mundial de Salud (OMS), como variables que sirven para medir los cambios. Un indicador es un término del cual se va a poder gestionar la información.

Ejemplos:

- Estado nutricional obeso o Bajo Peso.
- Dispositivo Intrauterino Implantado.
- Control de foco.
- Captación de embarazo.

Conductas Asistenciales: Son determinadas acciones ejecutadas o llevadas a cabo por el personal de salud al atender a un paciente, ya sea de prevención, rehabilitación y educación. Una Conducta Asistencial es un término del cual se va a poder gestionar la información.

Ejemplos:

- Examen mama.
- Gotas gruesas.
- Paciente ingresado en observación.

Categorías: Clasificación con que se describe a un paciente según parámetros determinados, ya sea por grupo de edades o un estado característico de la persona. Es útil para hacer pequeñas predicciones sobre su comportamiento o tratamiento. Una Categoría es un término del cual se va a poder gestionar la información.

Ejemplos:

- De 0 a 19 años.
- Mayores de 20.
- Embarazadas.

Usuarios: Es el conjunto de todas las personas que consultan la información de los Indicadores y Conductas Asistenciales y Categorías.

Usuarios (Nacionales, Provinciales, Municipales, Unidades de Salud): Son las personas que consultan la información de los Indicadores y Conductas Asistenciales y Categorías desde sus respectivos niveles (Nacional, Provincial, Municipal o a nivel de Unidades de Salud).

Especialista de la Dirección Nacional de la Atención Primaria de la Salud: Es la persona que gestiona la información de los Indicadores, Conductas Asistenciales y Categorías.

2.2.2 Diagrama del modelo de dominio.

El modelo del dominio o modelo conceptual, permite de manera visual mostrar al usuario los principales conceptos que se manejan en el dominio del sistema en desarrollo. . Esto ayuda a los usuarios, clientes y desarrolladores e interesados, a utilizar un vocabulario común para poder entender el contexto en que se emplaza el sistema.

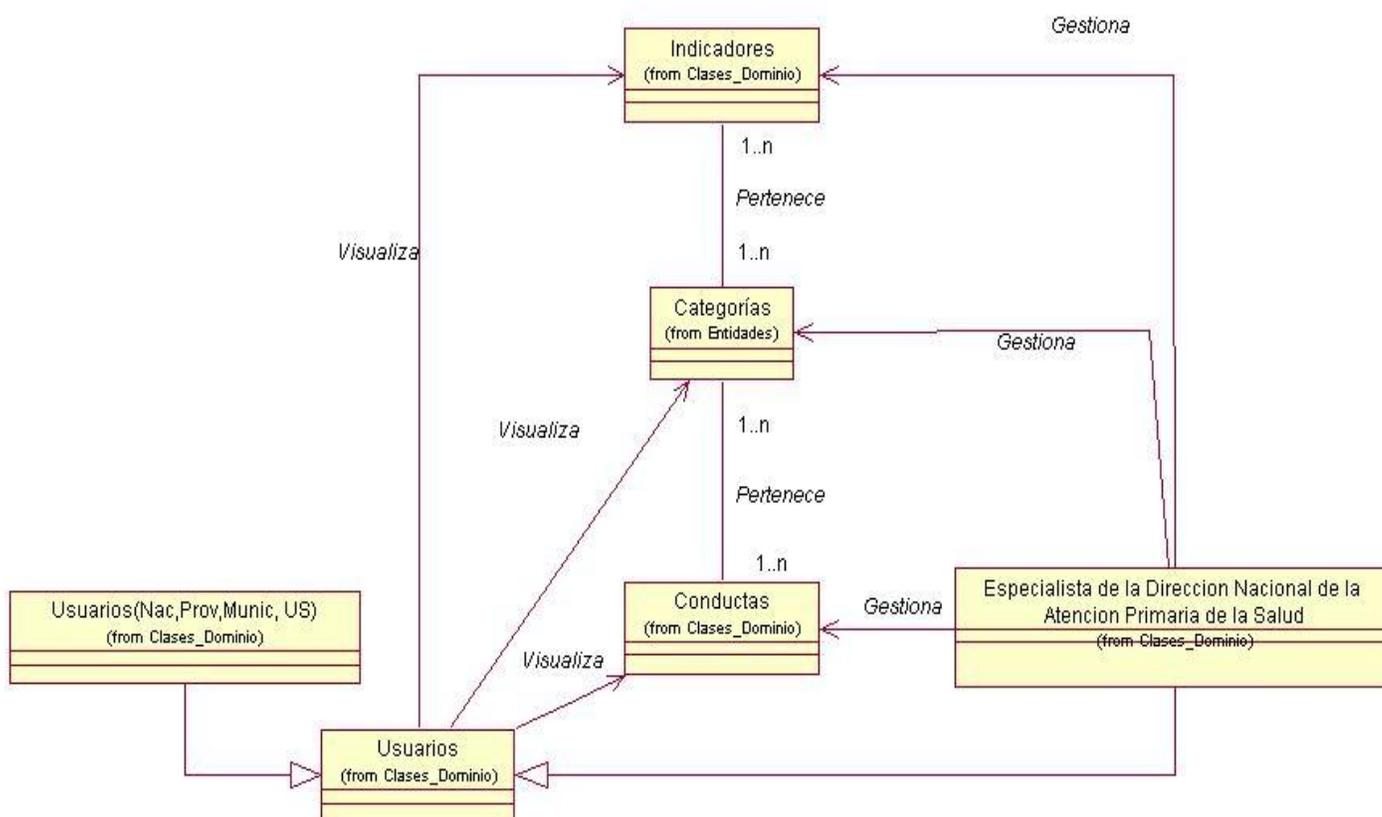


Figura 2.1 Diagrama del Modelo de Dominio del Registro de Indicadores y Conductas.

2.3 Propuesta del Sistema.

Para solucionar el problema planteado, se dispone crear el Registro de Indicadores y Conductas (RICAP).

Este Registro de Indicadores y Conductas gestiona información de las conductas asistenciales y los indicadores de la salud del trabajo realizado por los médicos. Las conductas asistenciales y los indicadores de salud están agrupadas de forma estratégica según las categorías correspondientes, las cuales también pueden ser gestionadas en RICAP.

2.3.1 Especificación de Requerimientos de Software.

Un Requerimiento de Software es la condición o capacidad que tiene que ser lograda o tenida por un sistema o componente de un sistema para satisfacer un contrato, estándar, u otro documento impuesto formalmente. Los requerimientos de Software se dividen en requerimientos funcionales y requerimientos no funcionales.

2.3.1.1 Requerimientos Funcionales.

Los requerimientos funcionales son capacidades o condiciones que el sistema debe cumplir sin alterar la funcionalidad del producto.

RF1 Gestionar la información de las **Conductas Asistenciales** de la atención primaria de la salud:

RF1.1 Insertar la información de las conductas asistenciales de la atención primaria de la salud.

RF1.2 Eliminar la información de las conductas asistenciales de la atención primaria de la salud.

RF1.3 Modificar la información de las conductas asistenciales de la atención primaria de la salud.

RF2 Gestionar la información de los **Indicadores** de la salud:

RF2.1 Insertar la información de los indicadores de la salud.

RF2.2 Eliminar la información de los indicadores de la salud.

RF2.3 Modificar la información de los indicadores de la salud.

RF3 Gestionar la información de las **Categorías**:

RF3.1 Insertar la información de las categorías.

RF3.2 Eliminar la información de las categorías.

RF3.3 Modificar la información de las categorías.

RF4 Listar la información de las **Conductas Asistenciales**.

RF5 Listar la información de los **Indicadores**.

RF6 Listar la información de las **Categorías**.

RF7 Imprimir información clasificada referente de los indicadores de la Salud y las Conductas Asistenciales.

RF8 Permitir búsquedas dinámicas de acuerdo a criterios seleccionados.

2.3.1.2 Requerimientos no Funcionales.

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener. Estas propiedades son las características que hacen al producto rápido o confiable, atractivo, usable.

Usabilidad.

RNF1: El sistema debe garantizar un acceso fácil y rápido, podrá ser usado por cualquier usuario que posea pocos conocimientos informáticos y de un ambiente Web en sentido general.

Rendimiento.

RNF2: El sistema debe tener una similitud en sus páginas y estar poco cargado, posibilitando que el sistema devuelva las respuestas de una manera eficiente, siendo más sencillo de entender y usar por el usuario.

Soporte.

RNF3: El personal que trabaja con el módulo debe contar con el nivel técnico requerido mediante adiestramiento de servicio.

Portabilidad.

RNF4: Permitir que el sistema se ejecute sobre el Sistema Operativo Linux, Windows 98 o superior.

Seguridad.

RNF5: Disponer de un mecanismo de seguridad basado en el modelo de Autenticación, Autorización y Auditoria (AAA).

Confiabilidad: La información manejada por el sistema está protegida de acceso no autorizado. El sistema debe prevenir posibles fallos y/o errores y presentar facilidades para una rápida recuperación en dichos casos.

Integridad: Que la información sea modificada (incluyendo su creación y borrado) sólo por personal autorizado. Se permitirá la creación de copias de respaldo que puedan restaurar el sistema en caso de fallo crítico o pérdida total de la información.

Disponibilidad: Los usuarios autorizados tendrán acceso a la información en todo momento, se debe lograr balancear la carga de acceso entre múltiples servidores, disminuyendo los tiempos de respuesta.

Apariencia o Interfaz Externa.

RNF6: La interfaz debe ser sencilla y amigable ya que el usuario no es experto en el uso de las aplicaciones Web.

Ayuda y Documentación en Línea.

RNF7: Disponer de instrucciones en una opción de ayuda.

Software.

RNF8: Los clientes tendrán acceso al registro a través de cualquier navegador Web. Recomendados Mozilla 1.5, Internet Explorer 5.0 o superior. El servidor debe tener PHP Versión 4.3.4 (aunque debe trabajar también con el 4.3.2), Biblioteca PEAR-SOAP 0.8RC3, Módulo XSLT (Sablotron) en PHP, Módulo DBX en PHP, Servidor de Base de Datos MySQL Versión 4 , Servidor HTTP (preferiblemente Apache) , Web Browser que soporte DHTML y CSS2.

Hardware.

RNF9: Requerimientos mínimos:

- 🏠 Ordenador Pentium o superior.
- 🏠 64 MB de Memoria RAM.
- 🏠 Monitor VGA o superior.
- 🏠 Teclado y Mouse.

- ⌚ Procesador 486DX / 66 MHZ o superior.
- ⌚ Disco duro de 20 GB.
- ⌚ Impresora de puntos.
- ⌚ Insumos. (Disquetes, CD RW, Papel continuo y cintas de impresora).
- ⌚ La PC de trabajo debe estar conectada a una Red de Área Local (LAN).
- ⌚ Conectividad con el nodo local de INFOMED.

2.3.2 Modelo de Casos de Uso del Sistema.

2.3.2.1 Definición de actores.

El actor del sistema no es parte de él, sino que representa el rol que juega una o varias personas, un equipo o un sistema automatizado que pueden intercambiar información con el mismo. Destacar que rol es la labor que se realiza frente al sistema. En la siguiente tabla se representan los actores del sistema propuesto:

Actor	Descripción
Especialista de la Dirección Nacional de la Atención Primaria.	Es el encargado de la gestión de la información (crear, modificar y eliminar), además de tener los permisos de un usuario normal de cualquiera de los niveles de la Atención Primaria (visualizador).
Usuario.	Todos los usuarios que soliciten RICAP desde cualquier lugar y en cualquier nivel. Es el encargado de listar, hacer búsquedas y recibir información de la aplicación. Se puede denominar al Sistema Externo.

Tabla 2.1 Definición de los actores del sistema del Registro de Indicadores y Conductas.

2.3.2.2 Diagrama de Casos de Uso.

Un diagrama de caso de uso del sistema es una especie de diagrama de comportamiento en el Lenguaje de Modelado Unificado donde no existe una forma de describir los mismos. El diagrama de casos de uso del sistema representa gráficamente a los procesos y su interacción con los actores del sistema. Estos procesos constituyen los casos de uso, los cuales se basan en las funcionalidades que debe cumplir el sistema.

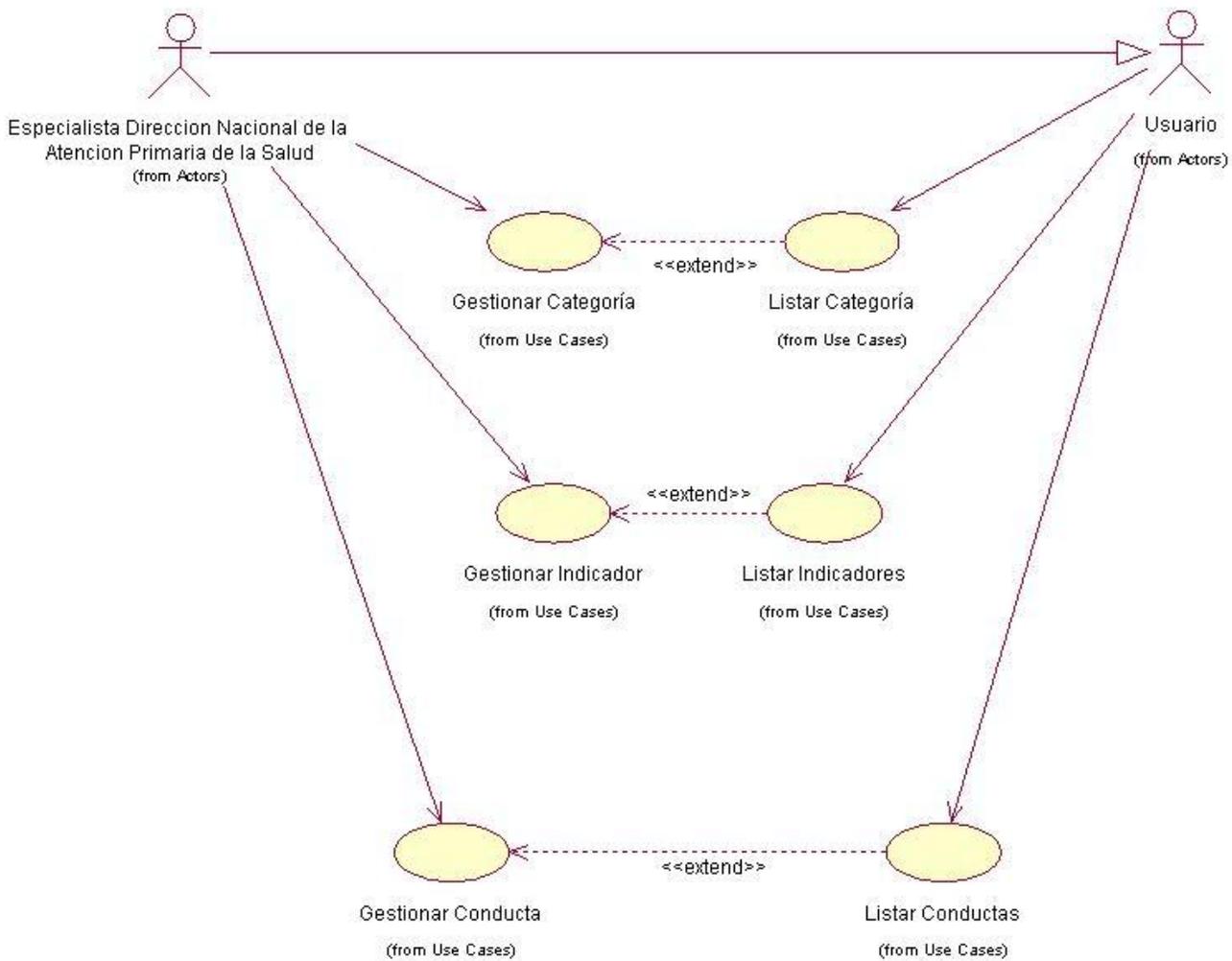


Figura 2.2 Diagrama de Casos de Uso del sistema del Registro de Indicadores y Conductas.

2.3.2.3 Descripción textual de los Casos de Uso.

CARACTERÍSTICAS DEL SISTEMA

CU-01	Gestionar Indicador
Actores:	Especialista de la Dirección Nacional de la Atención Primaria.
Descripción:	El caso de uso se inicia cuando el Especialista de la Dirección Nacional de la Atención Primaria selecciona de las Opciones de Menú, en la opción “Indicadores de Salud” que está dentro de la etiqueta “Gestionar” alguna de las opciones de submenú que se muestran. Al seleccionarla el sistema permite insertar, actualizar y eliminar un Indicador. Para actualizar o eliminar un Indicador primero debe buscarlo (extiende el CU Listar Indicadores). Una vez confirmada la inserción, actualización o eliminación por parte del usuario se actualiza la base de datos y finaliza el Caso de Uso.
Requisitos:	RF 2, RF5.
Prioridad:	Crítico

Tabla 2.2 Descripción del Caso de Uso Gestionar Indicador.

CU-02	Gestionar Conducta
Actores:	Especialista de la Dirección Nacional de la Atención Primaria.
Descripción:	El caso de uso se inicia cuando el Especialista de la Dirección Nacional de la Atención Primaria selecciona de las Opciones de Menú, en la etiqueta “Conductas o Accionar” que está dentro de la etiqueta “Gestionar” alguna de las opciones de submenú que se muestran. Al seleccionarla el sistema permite insertar, actualizar y eliminar una Conducta. Para actualizar o eliminar una Conducta primero debe buscarla (extiende el CU Listar Conductas). Una vez confirmada la inserción, actualización o eliminación por parte del usuario se actualiza la base de datos y finaliza el Caso de Uso.
Requisitos:	RF1, RF4.
Prioridad:	Crítico

Tabla 2.3 Descripción del Caso de Uso Gestionar Conducta.

CU-03	Gestionar Categoría
Actores:	Especialista de la Dirección Nacional de la Atención Primaria.

CARACTERÍSTICAS DEL SISTEMA

Descripción:	El caso de uso se inicia cuando el Especialista de la Dirección Nacional de la Atención Primaria selecciona de las Opciones de Menú, en la etiqueta “Categorías” que está dentro de la etiqueta “Gestionar” alguna de las opciones de submenú que se muestran. Al seleccionarla el sistema permite insertar, actualizar y eliminar una Categoría. Para actualizar o eliminar una Categoría primero debe buscarla (extiende el CU Listar Categoría). Una vez confirmada la inserción, actualización o eliminación por parte del usuario se actualiza la base de datos y finaliza el Caso de Uso.
Requisitos:	RF3, RF6.
Prioridad:	Crítico

Tabla 2.4 Descripción del Caso de Uso Gestionar Categoría.

CU-04	Listar Indicadores
Actores:	Usuarios
Descripción:	El Caso de Uso comienza cuando el usuario selecciona de las Opciones de Menú en la etiqueta “Buscar”, la opción “Indicadores de salud”, el sistema hace una búsqueda (según los criterios introducidos) y muestra en la página el listado de los Indicadores, mostrando su descripción y categoría.
Requisitos:	RF5.
Prioridad:	Crítico

Tabla 2.5 Descripción del Caso de Uso Listar Indicadores.

CU-05	Listar Conductas
Actores:	Usuarios
Descripción:	El Caso de Uso comienza cuando el usuario selecciona de las Opciones de Menú en la etiqueta “Buscar”, la opción “Conductas o Accionar”, el sistema hace una búsqueda (según los criterios introducidos) y muestra en la página el listado de las Conductas, mostrando su descripción y categoría.

Requisitos:	RF4.
Prioridad:	Crítico

Tabla 2.6 Descripción del Caso de Uso Listar Conductas.

CU-06	Listar Categorías
Actores:	Usuarios
Descripción:	El Caso de Uso comienza cuando el usuario selecciona de las Opciones de Menú en la etiqueta “Buscar”, la opción “Categorías”, el sistema muestra en la página el listado de las Categorías.
Requisitos:	RF6.
Prioridad:	Crítico

Tabla 2.7 Descripción del Caso de Uso Listar Categoría.

En el capítulo se mostraron las características del sistema a desarrollar. Se describen los principales conceptos del entorno y sus relaciones a través del modelo del dominio. Se definieron las condiciones o capacidades que debe cumplir el sistema, así como los actores del sistema, el diagrama de casos de uso y la descripción textual de cada uno de los casos de uso.

Capítulo 3: Análisis y Diseño del Sistema.

Introducción.

Con el presente capítulo se tiene el propósito de definir la estructura y elementos del análisis y diseño, describir los casos de uso en términos de clases y sus objetos, representándolos gráficamente en Diagramas de Clases, así como fundamentar los patrones empleados y estructurar el Modelo de Datos.

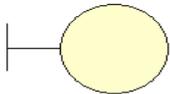
3.1 Modelo de Análisis.

El análisis consiste en obtener una visión del sistema, que se preocupa de ver qué hace, de modo que sólo se interesa por los requisitos funcionales.

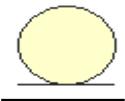
En la construcción del modelo de análisis se identifican las clases que describen la realización de los casos de uso, los atributos y las relaciones entre ellas. Con esta información se construye el Diagrama de clases del análisis. El Modelo de Análisis es el resultado de la actividad de analizar los Casos de Uso.

Tener una visión general de la propuesta de sistema es una de las principales ventajas de este modelo.

A continuación se explica cada estereotipo de las clases del análisis:



Clase Interfaz: Estas clases modelan la interacción Actor – Sistema. Representan ventanas, Formularios, comunicación con otros sistemas o dispositivos, interfaces de comunicaciones, interfaces de impresoras. Cada clase interfaz debe asociarse con al menos un actor, y viceversa.



Clase Entidad: Modelan la información persistente del sistema y el comportamiento asociado a una información. Las clases entidad suelen mostrar una estructura de datos lógica y contribuyen a comprender de qué información depende el sistema.



Clase Control: Coordinan el trabajo de las clases. Encapsulan el comportamiento de un CU. Los aspectos dinámicos del sistema se modelan con clases de control, debido a que ellas manejan y coordinan las acciones y los flujos de control principales, y delegan trabajo a otros objetos, es decir objetos de interfaz y de entidad.

3.1.2 Diagrama de Clases del Análisis

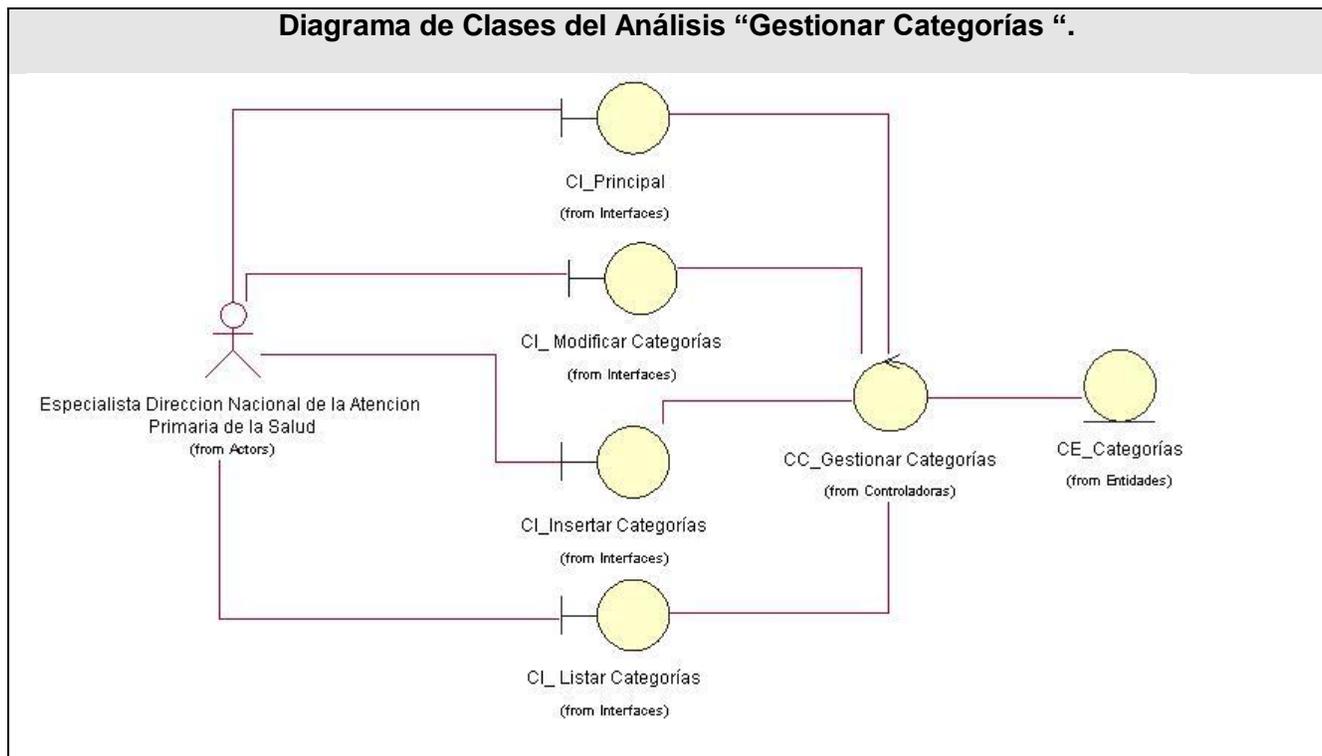


Tabla 3.1 Diagrama de Clases del Análisis “Gestionar Categorías”.

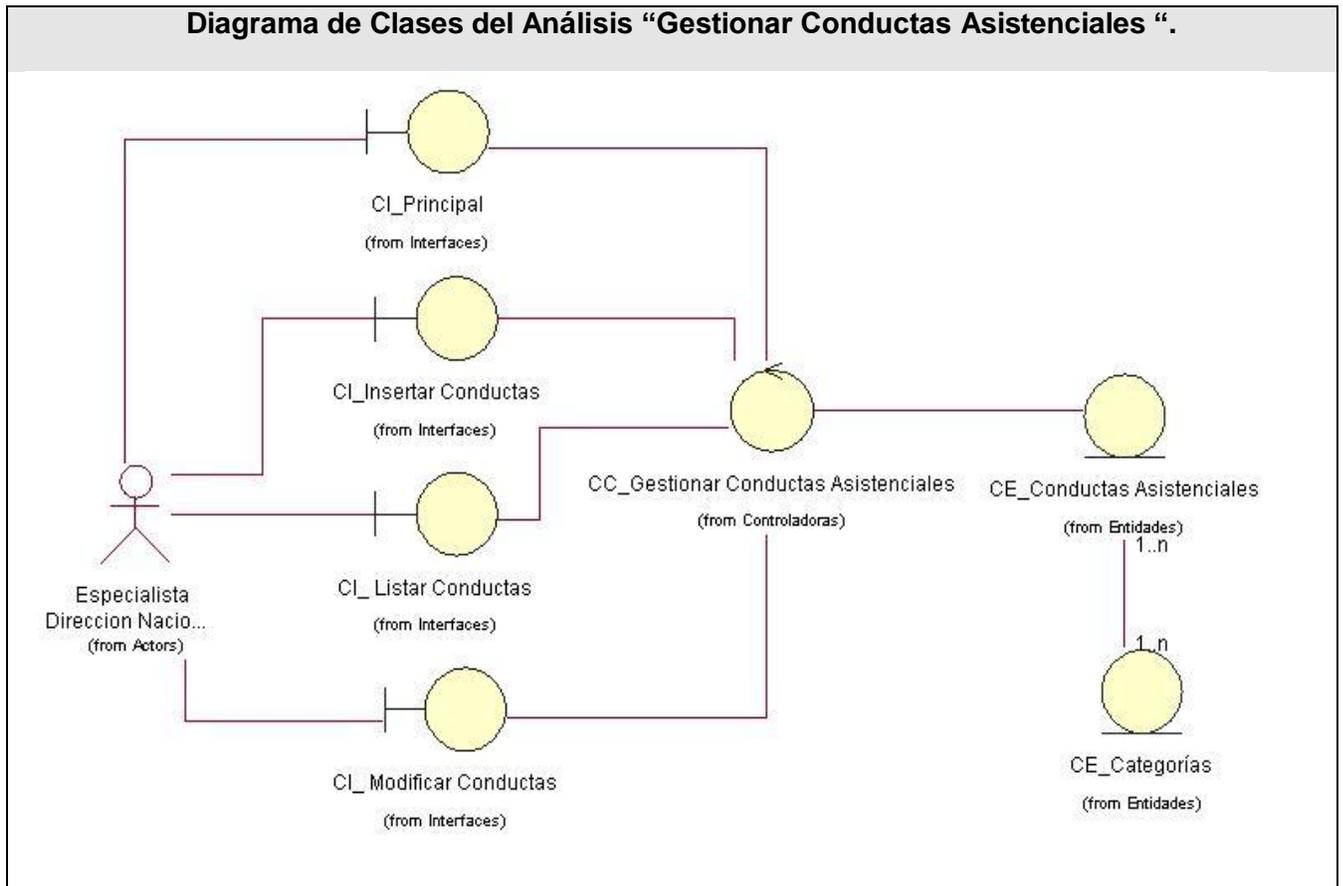


Tabla 3.2 Diagrama de Clases del Análisis “Gestionar Conductas Asistenciales”.

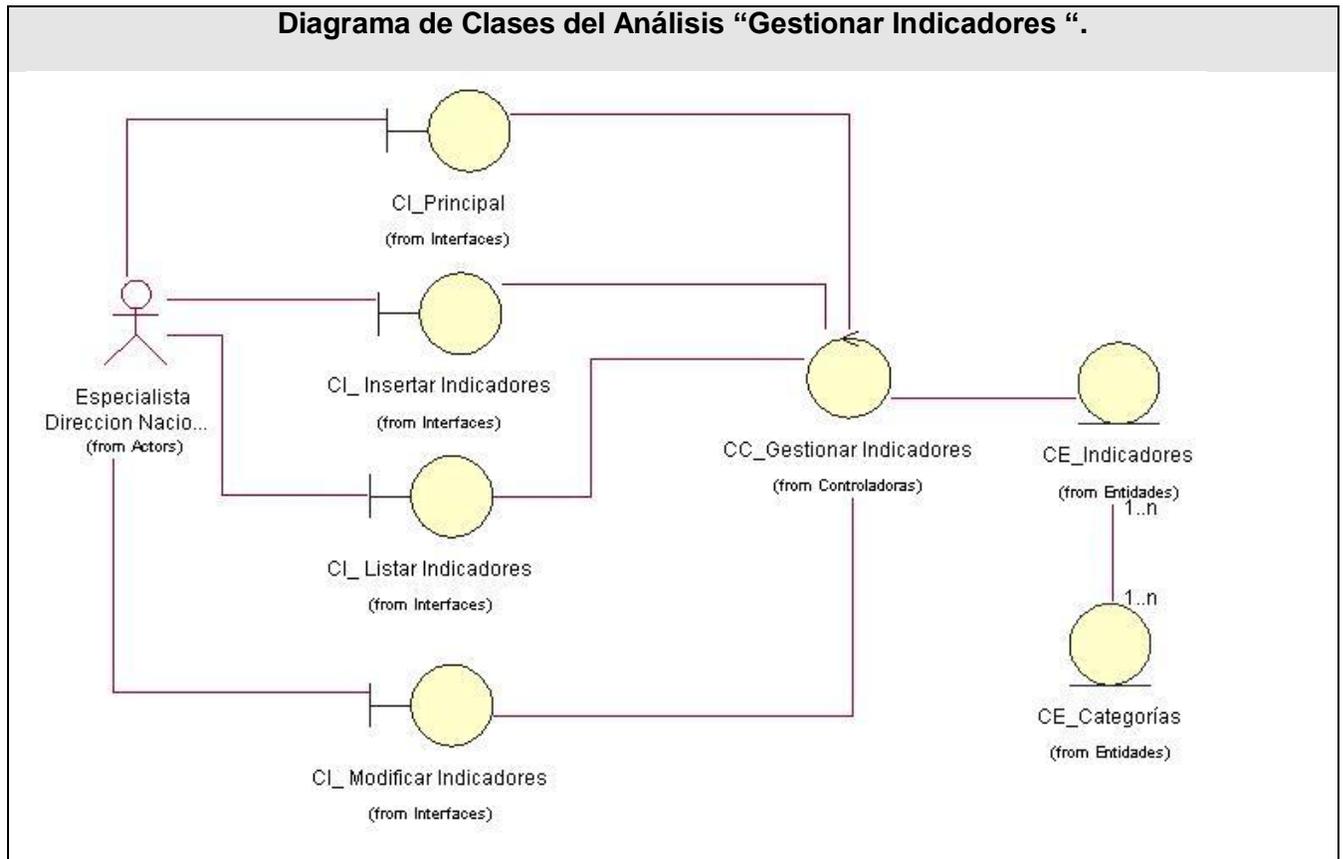


Tabla 3.3 Diagrama de Clases del Análisis “Gestionar Indicadores”.

3.1.2 Diagramas de iteración.

Los diagramas de interacción se encuentran divididos en diagramas de secuencia y diagramas de colaboración. Estos muestran las interacciones entre objetos o subsistemas mediante transferencia de mensajes, es decir, muestran la transferencia de información, con la expectativa de desencadenar una actividad. Además modelan aspectos dinámicos del sistema.

Los diagramas de colaboración son una forma alternativa de los diagramas de secuencia, explican gráficamente las interacciones entre los objetos organizadas entorno a los objetos y los enlaces entre ellos, por lo que se decidió realizarlos con objetivo fundamental de destacar la relación estructural entre los objetos que interactúan.

Diagrama de Colaboración del Análisis del Caso de Uso "Gestionar Categoría".

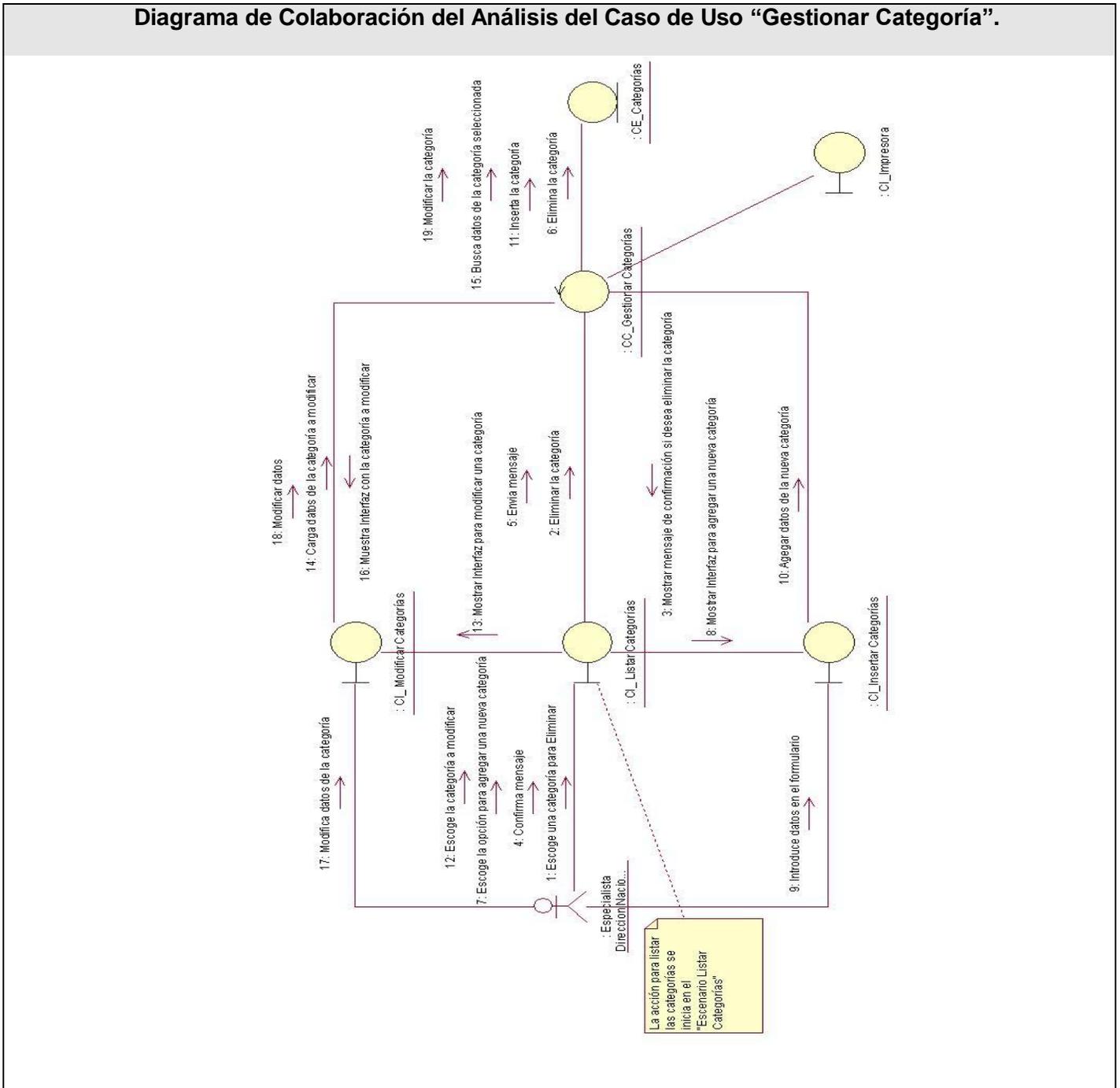


Tabla 3.4 Diagrama de Colaboración del Análisis del Caso de Uso "Gestionar Categoría".

Diagrama de Colaboración del Análisis del Caso de Uso "Gestionar Conducta".

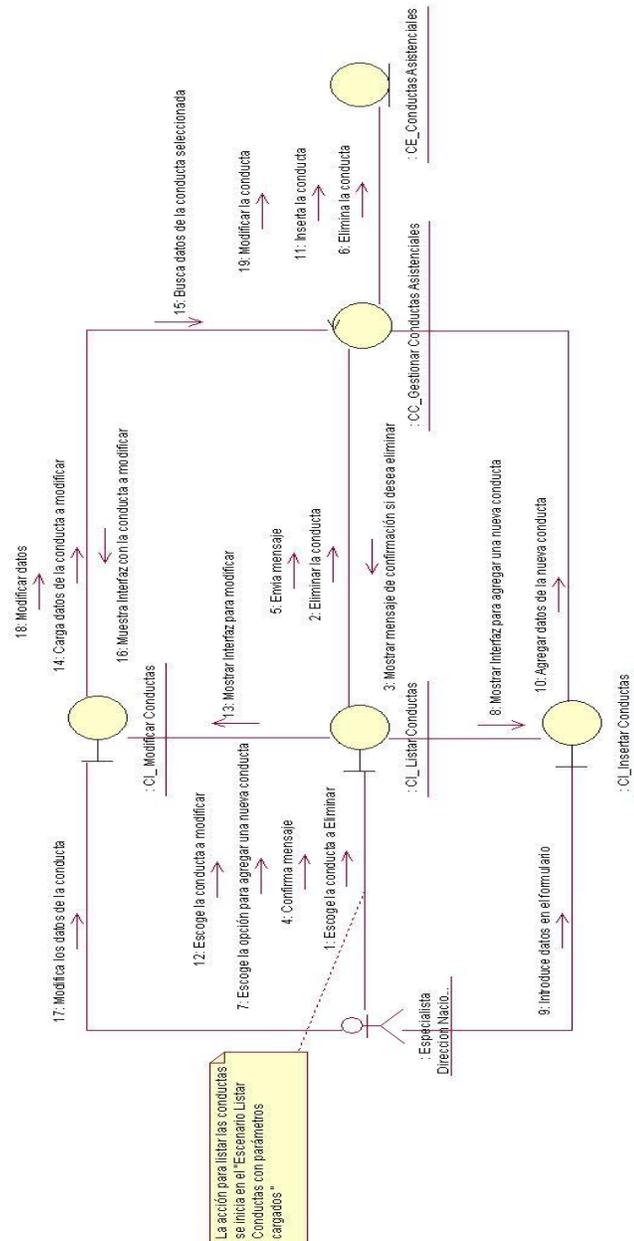


Tabla 3.5 Diagrama de Colaboración del Análisis del Caso de Uso "Gestionar Conducta".

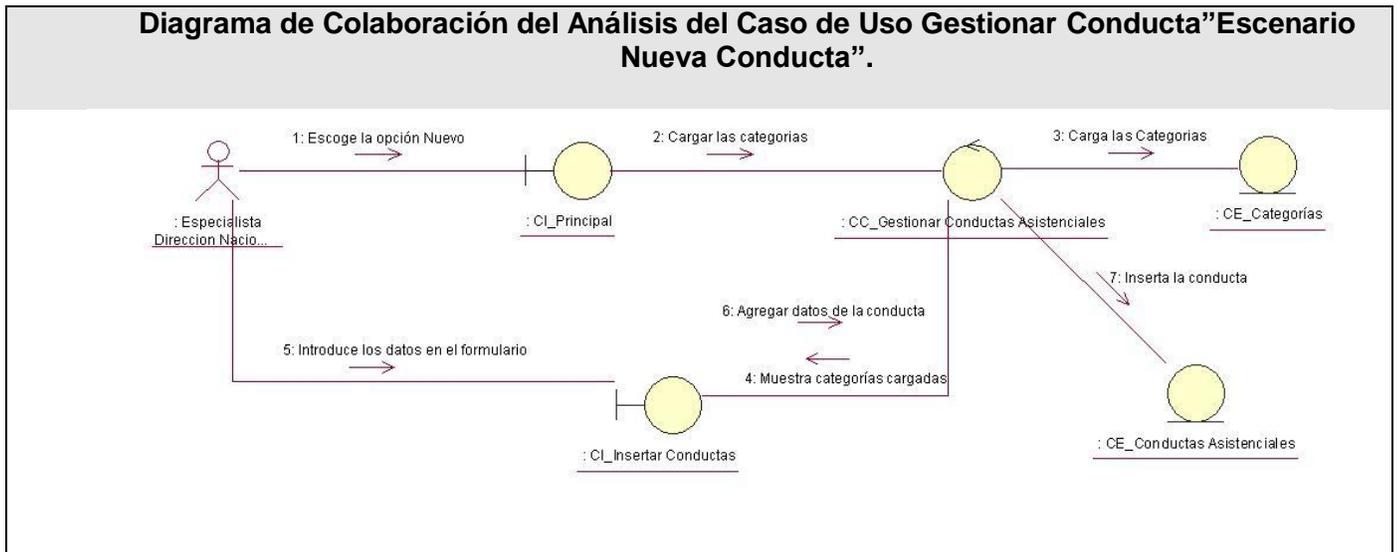


Tabla 3.6 Diagrama de Colaboración del Análisis del Caso de Uso Gestionar Conducta: "Escenario Nueva Conducta".

Diagrama de Colaboración del Análisis del Caso de Uso Gestionar Conducta "Escenario Listar Conductas con parámetros cargados".

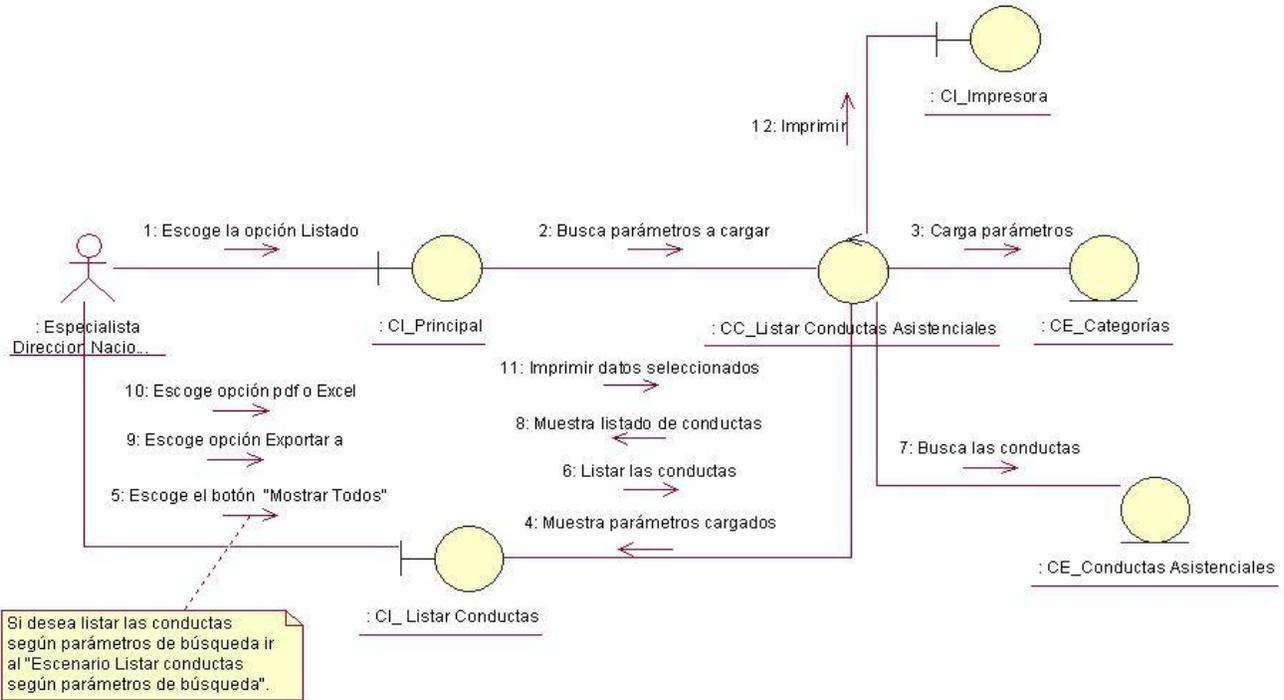


Tabla 3.7 Diagrama de Colaboración del Análisis del Caso de Uso Gestionar Conducta "Escenario Listar Conductas con parámetros cargados".

Diagrama de Colaboración del Análisis del Caso de Uso "Gestionar Indicador".

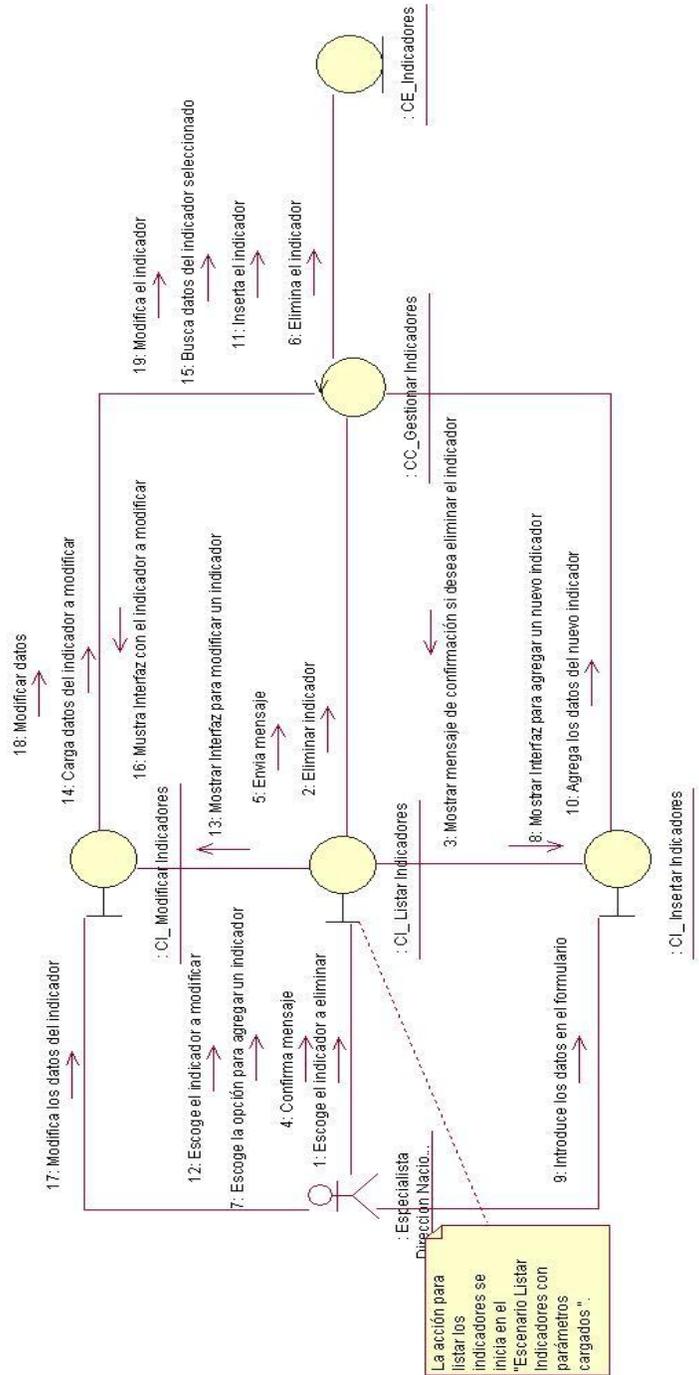


Tabla 3.8 Diagrama de Colaboración del Análisis del Caso de Uso "Gestionar Indicador".

3.2 Modelo del Diseño.

El diseño es un refinamiento del análisis que tiene en cuenta los requisitos no funcionales. Con el modelado del diseño se describe cómo el sistema será realizado a partir de la funcionalidad prevista y las restricciones impuestas (requerimientos). Contribuye a una arquitectura estable y sólida, y crear un plano del modelo de implementación.

Los artefactos del Modelo de Diseño son: Modelo de Despliegue, Descripción de la Arquitectura, Realización de Casos de Uso, Clase del Diseño, Subsistema de Diseño, Interfaz.

3.2.1 Justificación del uso de Patrones.

Un patrón es una solución a un problema. Para que una solución sea considerada un patrón, debe haber comprobado su efectividad resolviendo problemas similares en ocasiones anteriores, debe ser reusable, lo que significa que es aplicable a diferentes problemas de diseño en distintas circunstancias.

Los patrones de diseño son principios generales basados en la experiencia, son la base para la búsqueda de soluciones a problemas comunes en el desarrollo de software y otros ámbitos referentes al diseño de interacción o interfaces.

En el sistema desarrollado se utilizan patrones como: Modelo-Vista-Controlador, Fachada, Proxy, Alta Cohesión y Bajo Acoplamiento, estos ayudaron a diseñar satisfactoriamente en menos tiempo, a construir clases reutilizables, facilitaron la documentación y llevaron a cabo la definición de una arquitectura pequeña, simple y comprensible.

✓ **Modelo- Vista- Controlador.**

La lógica de un interfaz de usuario cambia con más frecuencia que los almacenes de datos y la lógica de negocio. Si realizamos un diseño que mezcle los componentes de interfaz y de negocio, cuando necesitemos cambiar el interfaz, tendremos que modificar trabajosamente los componentes de negocio, esto provoca un mayor riesgo de error y que se trabaje más.

El patrón de diseño modelo vista controlador desacopla la vista del modelo, con la finalidad de mejorar la reusabilidad. De esta forma las modificaciones en las vistas impactan en menor medida en la lógica de negocio o de datos.

Los Elementos del patrón son los siguientes:

- Modelo: datos y reglas de negocio.
- Vista: muestra la información del modelo al usuario.
- Controlador: gestiona las entradas del usuario.¹⁹

Para la solución (RICAP):

El modelo: Formado por la base de datos de RICAP.

El controlador: Métodos del negocio.

Las vistas: Formadas por los ficheros XSL y JavaScript, clientes PHP de la capa de Presentación y FachadaRICAP.

✓ **Bajo Acoplamiento.**

Este patrón de asignación de responsabilidades propone que debe haber pocas dependencias entre las clases. La idea de tener las clases lo menos ligadas entre sí que se pueda. De tal forma que en caso de producirse una modificación en alguna de ellas, se tenga la mínima repercusión posible en el resto de clases, potenciando la reutilización, y disminuyendo la dependencia entre las clases.

El diseño del módulo RICAP está presidido por la creación de dependencias escasas, un ejemplo se manifiesta en la definición de métodos del negocio que son poco dependientes de otros, lo que reduce el impacto del cambio y favorece la reutilización.

✓ **Alta Cohesión.**

La cohesión es la medida de la fuerza que une a las responsabilidades de una clase. Una clase con baja cohesión es aquella que hace muchas cosas no afines o muchas tareas, lo que trae como consecuencias dificultades para entender, reutilizar y conservarla. Son delicadas y las afectan constantemente los cambios.

En el patrón alta cohesión cada elemento de nuestro diseño debe realizar una labor única dentro del sistema, no desempeñada por el resto de los elementos y auto-identificable. Se dice que la información que almacena una clase debe de ser coherente y está en la mayor medida de lo posible relacionada con la clase

¹⁹ *Tecnología orientada a procesos de negocio.* (s.f.). Recuperado el diciembre de 2007, de Tecnología orientada a procesos de negocio: <http://www.proactiva-calidad.com/java/patrones/mvc.html>

RICAP se diseñó asignando responsabilidades de forma tal que la cohesión sea alta, por lo que los métodos del negocio tienen bien concretadas sus responsabilidades, sin recargar sus funcionalidades, proporcionando mayor eficiencia y que el tiempo de respuesta y ejecución no exceda lo estimado, esto genera por ende bajo acoplamiento.

✓ **Fachada.**

El patrón fachada trata de simplificar la interface entre dos sistemas o componentes de software ocultando un sistema complejo detrás de una clase que hace las veces de pantalla o fachada.

La idea principal es la de ocultar todo lo posible la complejidad de un sistema, el conjunto de clases o componentes que lo forman, de forma que solo se ofrezca un (o unos pocos) punto de entrada al sistema tapado por la fachada.

Una ventaja de usar una clase fachada para comunicar las dos partes o componentes, es la de aislar los posibles cambios que se puedan producir en alguna de las partes. Si cambias, por poner un ejemplo, el medio de comunicación o de almacenamiento de una de las partes, la otra, que por ejemplo hace la presentación, no tiene porque enterarse, y viceversa.²⁰

✓ **Proxy.**

El patrón proxy permite acceder a un recurso mediante un intermediario, con varios propósitos, tales como diferir la carga, controlar el acceso, hacer caché, etc.

Los recursos a los que se puede querer acceder mediante intermediario podrían ser archivos, conexiones de red, objetos muy grandes, etc. El proxy permite acceder desde múltiples lugares al mismo recurso, o darle múltiples usos, cuando hacer copias es muy costoso o imposible. En lugar de eso, se crean múltiples proxies que hacen de intermediario entre el mismo recurso y sus diferentes clientes.²¹

²⁰ *Java en Castellano*. (2007). Recuperado en diciembre de 2007, de Java en Castellano: http://www.programacion.net/java/articulo/joa_patrones4/#joa_patrones3_fachada

²¹ *Patrón Proxy*. (diciembre de 2007). Recuperado en enero de 2008, de http://mendozajug.com.ar/portal/components/com_mambowiki/index.php?title=Patr%C3%B3n_Proxy

3.2.2 Definición de la estructura del diseño.

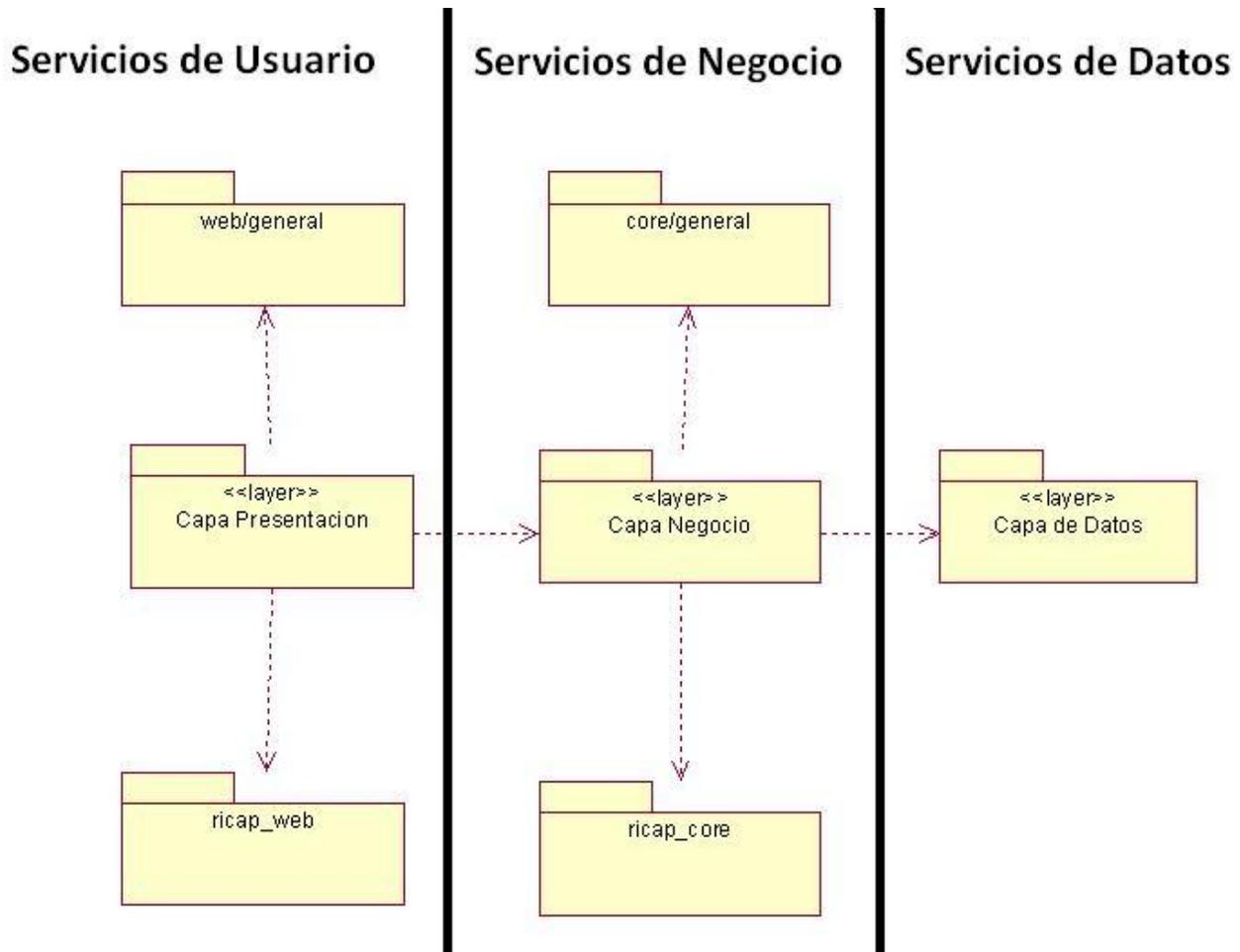


Figura 3.1 Diagrama de Subsistemas de Servicios.

Descripción de cada uno de los elementos que intervienen en la estructura del sistema:

Capa Presentación: Esta capa contiene la lógica de navegación y controla los eventos de las interfaces. Es la que interactúa con el usuario y hace transparente la complejidad del sistema a los mismos, al estar conformada por: DHTML, XML, PHP y XSL.

Capa Negocio: Esta capa contiene la lógica de negocio. Establece comunicación entre la capa de presentación y la de datos, recibiendo y respondiendo peticiones. Los métodos que la conforman reciben

solicitudes de la capa superior, se comunican con los ficheros de acceso a datos y actualizan o retornan información.

Capa de Datos: Contiene las tablas de la base de datos de RICAP, las que son gestionadas por la capa de negocio.

Web/general: Este paquete contiene la clase CFachada de la cual hereda la clase FachadaRICAP ubicada en la capa de presentación del módulo, la misma es la encargada de transformar las páginas xls en HTML mediante la función transforma, para esto incluye las clases PLASER_xml, PLASER_xslt y de acceder al negocio incluyendo a PLASER_Client.

Core/General: Este paquete contiene los ficheros utilizados por la capa de negocio para las validaciones y mensajes de error, así como los que brindan la posibilidad de generar ficheros en formato Portable Document Format (pdf) o Microsoft Office Excel (xls).

3.2.3 Diagramas de Clases del Diseño.

Los diagramas de clases muestran el diseño del sistema desde un punto de vista estático, a través de una colección de elementos declarativos, como clases, colaboraciones y sus relaciones.

RICAP es una aplicación web que se modelará con estereotipos web, lo que proporciona una mayor comprensión de las funcionalidades del sistema y mejor distinción de los atributos, operaciones y relaciones que están activas en el servidor y cuáles están activas cuando el usuario está interactuando con la página en el navegador cliente.

A continuación se explica cada estereotipo y cómo se utilizan en el diseño propuesto:



<<Server Page>>: Representa la clase que tiene código que se ejecuta en el servidor, la cual se encarga de construir (build) o generar el resultado HTML y/o realizar peticiones a la capa inferior.

<<Client Page>>: Es una página Web con formato XHTML. Mezcla de datos, presentación y lógica. Son interpretadas por el navegador. Sus atributos son las variables declaradas dentro del *script* que son accesibles para cualquier función dentro de la página. Cada página cliente es construida por una sola página de servidor.

<<FormHTML>>: Es una colección de elementos de entrada que están contenidos en la página cliente. Sus atributos son los elementos de entrada el formulario (input boxes, text areas, radio buttons, check boxes, hidden fields, entre otros). No tienen operaciones, el método para el paso de los parámetros es \$_POST y se comunican con las páginas servidores mediante *submit*.

Relaciones utilizadas entre las clases:

Desde	Hasta	Client Page	Form	Server Page
Client Page		<<link>>	---	<<build>>, <<Link>>
Form		---	---	<<Submit>>
Server Page		<<build>>, <<Link>>	<<Submit>>	<<Redirect>>. <<Call>>, <<include>>

Tabla 3.9 Relaciones entre las clases principales que conforman la extensión de UML para Web.

<<Build>>: Representa la relación existente entre las páginas cliente, que de forma general expresa cómo las páginas que se encuentran en el servidor construyen las páginas en el cliente. Es una relación direccional, donde una página servidor construye una o más páginas cliente.

<<Call>>: Se utiliza para llamadas a páginas servidoras que representan métodos de la capa de negocio.

<<Include>>: Una página servidor puede incluir a otra página del mismo tipo, pudiendo utilizar todas las funciones brindadas por esta última.

<<Link>>: Permite ir de una página cliente a otra página cliente.

<<Redirect>>: Una página servidora puede redireccionar el procesamiento a otra página, es decir, enviar información para que la otra ejecute la acción.

<<Submit>>: Envía los valores de un formulario a una página servidora.

A continuación se presentan algunos de los diagramas de clases del diseño:

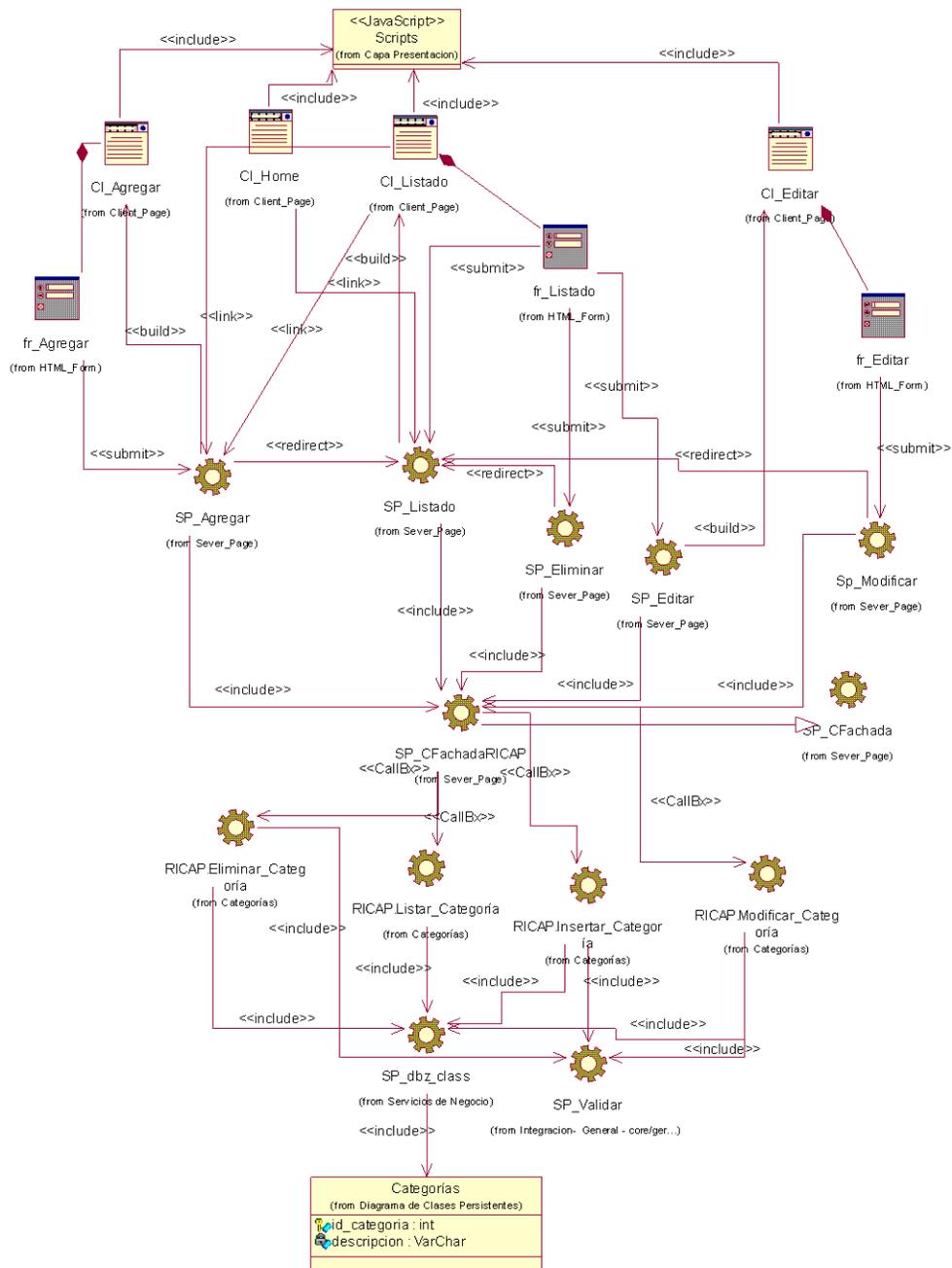


Figura 3.2 Diagrama de Clases del Diseño "Gestionar Categoría".

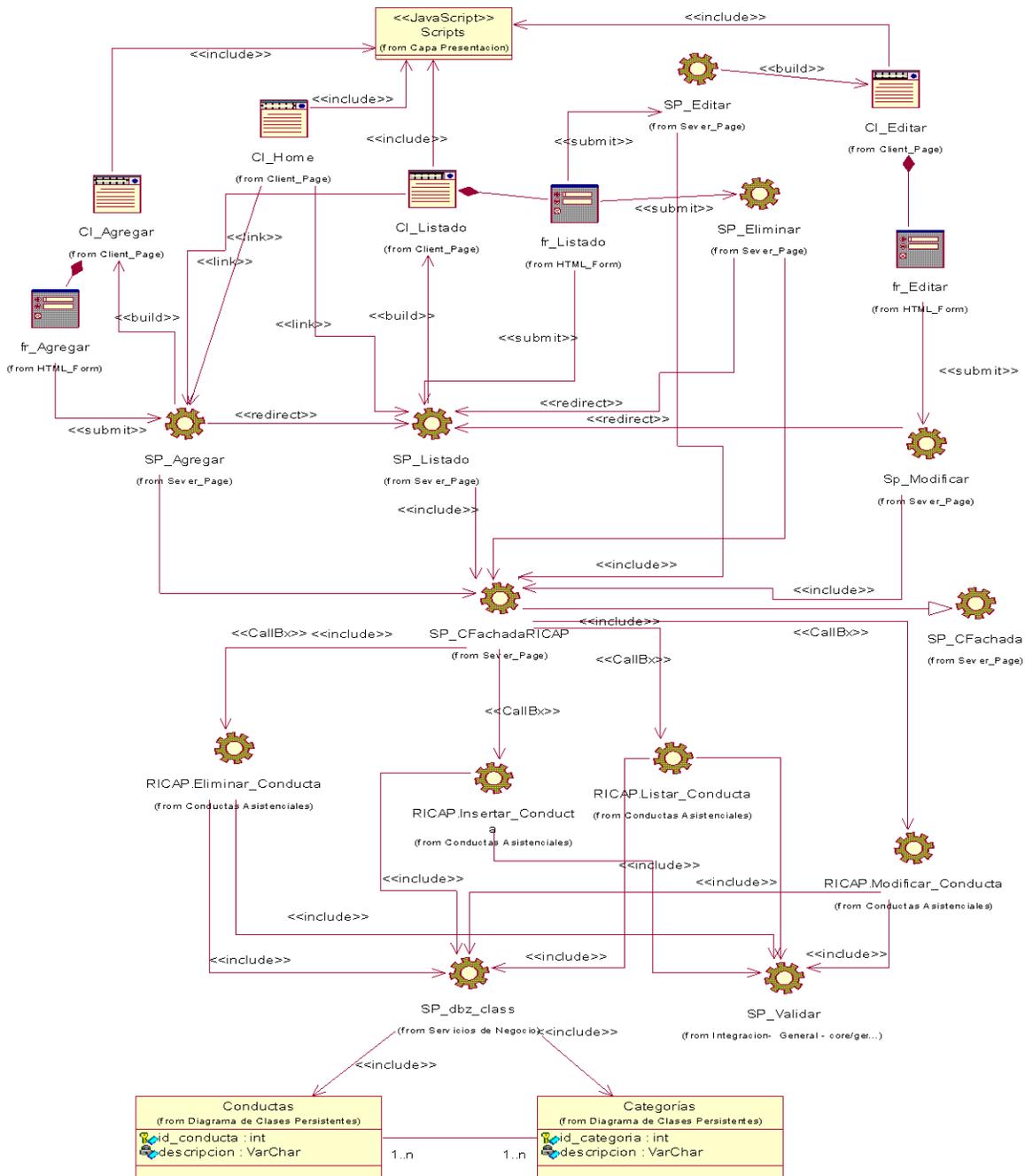


Figura 3.3 Diagrama de Clases del Diseño "Gestionar Conducta".

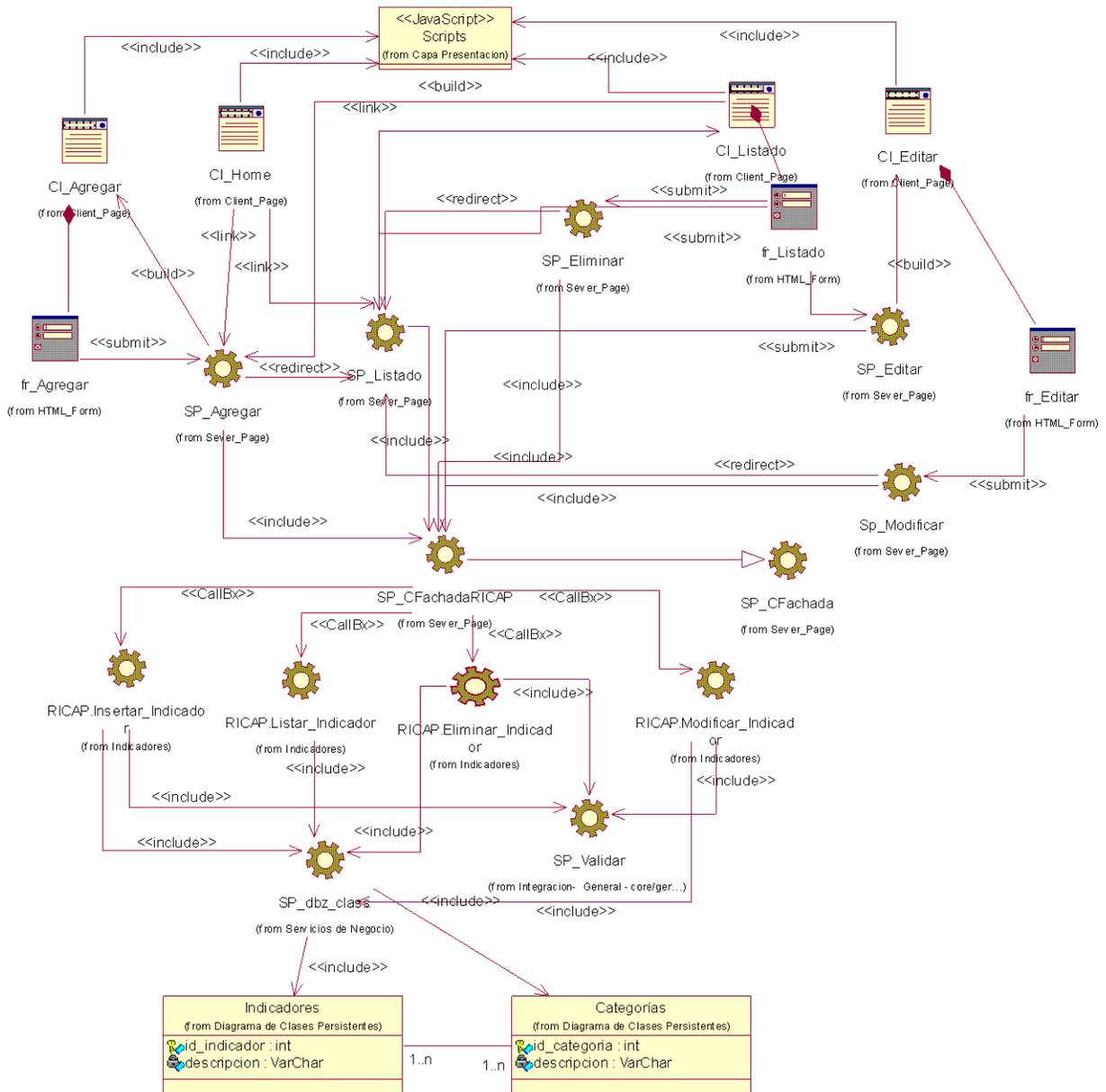


Figura 3.4 Diagrama de Clases del Diseño "Gestionar Indicador".

3.2.4 Descripción de las clases y atributos.

Una clase es la descripción de un conjunto de objetos que comparten los mismos atributos, operaciones, relaciones y semántica, definiendo como atributo a las propiedades o información detallada que posee una clase.

A continuación se describen las clases del diseño representadas en los diagramas mostrados anteriormente.

Páginas cliente.

Nombre: CI_listado
Tipo de clase: página cliente
Descripción General: La clase <i>CI_listado</i> es una página web que se ejecuta del lado del cliente sobre un browser o navegador web. Permite a los visualizadores o editor nacional la obtención de listado y búsquedas de información específica permitiendo generar documentos Portable Document Format (.pdf) y Microsoft Office Excel (.xls) para imprimir. A través de esta página se puede visualizar la información. El resultado de las búsquedas es paginada, permitiendo la movilidad por las diferentes páginas, inclusive ir directamente a la última. Es utilizada en los siguientes casos de uso: <ul style="list-style-type: none"> ✓ Listar Categorías. ✓ Listar Indicadores. ✓ Listar Conductas.

Tabla 3.10 Descripción de la página cliente CI_listado.

Nombre: CI_Editar
Tipo de clase: página cliente

<p>Descripción General: La clase <i>CI_Editar</i> es una página web que se ejecuta del lado del cliente sobre un browser o navegador web. Recibe los valores de la página cliente <i>CI_listado</i> .Permite al editor nacional modificar o visualizar información. Es utilizada en los siguientes casos de uso:</p> <ul style="list-style-type: none"> ✓ Gestionar Categoría. ✓ Gestionar Indicador. ✓ Gestionar Conducta.

Tabla 3.11 Descripción de la página cliente *CI_Editar*.

Nombre: <i>CI_Agregar</i>
Tipo de clase: página cliente
<p>Descripción General: La clase <i>CI_Agregar</i> es una página web que se ejecuta del lado del cliente sobre un browser o navegador web. Permite al editor nacional insertar información. Es utilizada en los siguientes casos de uso:</p> <ul style="list-style-type: none"> ✓ Gestionar Categoría. ✓ Gestionar Indicador. ✓ Gestionar Conducta.

Tabla 3.12 Descripción de la página cliente *CI_Agregar*.

Páginas servidoras

Nombre: <i>SP_CFachada</i>
Tipo de clase: página servidora
<p>Descripción General: La clase <i>SP_CFachada</i> es una clase que se encuentra dentro de PLASER, su función principal es la comunicación de sistemas complicados y brinda los puntos acceso entre estos. Se utiliza para separar la capa de presentación de la capa de acceso a datos.</p>

Tabla 3.13 Descripción de la página servidora *SP_CFachada*.

Nombre: SP_CFachadaRICAP
Tipo de clase: página servidora
Descripción General: La clase SP_CFachadaRICAP es una clase que hereda la clase SP_CFachada y la representa dentro del módulo. Cada módulo tiene la suya propia. Su principal función es modelar la comunicación con la capa de negocio.

Tabla 3.14 Descripción de la página servidora SP_CFachadaRICAP.

Nombre: SP_Listado
Tipo de clase: página servidora
Descripción General: La clase <i>SP_Listado</i> es una clase que se ejecuta del lado del servidor en la capa de presentación. Su responsabilidad es construir la página cliente <i>Cl_listado</i> . Aplica un documento (XSL) a otro documento XML para transformarlo y mostrarlo al usuario en formato XHTML para posteriormente recibir los parámetros de búsqueda. Es utilizada en los siguientes casos de uso: <ul style="list-style-type: none"> ✓ Listar Categorías. ✓ Listar Indicadores. ✓ Listar Conductas.

Tabla 3.15 Descripción de la página servidora SP_Listado.

Nombre: SP_Agregar
Tipo de clase: página servidora

Descripción General: La clase *SP_Agregar* es una clase que se ejecuta del lado del servidor en la capa de presentación. Su responsabilidad es construir la página cliente *Cl_Agregar*. Aplica un documento (XSL) a otro documento XML para transformarlo y mostrarlo al usuario en formato XHTML para posteriormente recibir los valores introducidos. Es utilizada en los siguientes casos de uso:

- ✓ Gestionar Categoría.
- ✓ Gestionar Indicador.
- ✓ Gestionar Conducta.

Tabla 3.16 Descripción de la página servidora *SP_Agregar*.

Nombre: *SP_Eliminar*

Tipo de clase: página servidora

Descripción General: La clase *SP_Eliminar* es una clase que se ejecuta del lado del servidor en la capa de presentación. Su responsabilidad es recibir los valores de la página cliente *Cl_listado* y posteriormente enviarlos hacia la capa de negocio al método del negocio para eliminar los datos involucrados. Una vez concluida la ejecución de sus responsabilidades devuelve una respuesta a la clase *SP_Eliminar*. Es utilizada en los siguientes casos de uso:

- ✓ Gestionar Categoría.
- ✓ Gestionar Indicador.
- ✓ Gestionar Conducta.

Tabla 3.17 Descripción de la página servidora *SP_Eliminar*.

Nombre: *SP_Editar*

Tipo de clase: página servidora

<p>Descripción General: La clase <i>SP_Editar</i> es una clase que se ejecuta del lado del servidor en la capa de presentación. Su responsabilidad es construir la página cliente <i>Cl_Editar</i>. Aplica un documento (XSL) a otro documento XML para transformarlo y mostrarlo al usuario en formato XHTML para posteriormente recibir los valores modificados. Es utilizada en los siguientes casos de uso:</p> <ul style="list-style-type: none"> ✓ Gestionar Categoría. ✓ Gestionar Indicador. ✓ Gestionar Conducta.
--

Tabla 3.18 Descripción de la página servidora SP_Editar.

Nombre: SP_Modificar
Tipo de clase: página servidora
<p>Descripción General: Recibe los valores de la clase <i>SP_Editar</i>, los valida y los envía hacia la capa de negocio al método de modificación de los datos involucrados. Una vez concluida la ejecución de sus responsabilidades devuelve una respuesta a la clase <i>SP_Modificar</i>. Es utilizada en el siguiente caso de uso:</p> <ul style="list-style-type: none"> ✓ Gestionar Categoría. ✓ Gestionar Indicador. ✓ Gestionar Conducta.

Tabla 3.19 Descripción de la página servidora SP_Modificar.

Nombre: RICAP.Listar_Categoría	
Tipo de clase: página servidora	
<p>Descripción General: La clase <i>Listar_Categoría</i> se ejecuta del lado del servidor en la capa de negocio. Su función es devolver al usuario un listado de las Categorías.</p>	
Parámetros de Entrada	
Nombre	Tipo

offset	int
cantidad	int
tipo_salida	int

Tabla 3.20 Descripción de la página servidora RICAP.Listar_Categoría.

Nombre: RICAP.Listar_Indicador	
Tipo de clase: página servidora	
Descripción General: La clase RICAP.Listar_Indicador es una clase que se ejecuta del lado del servidor en la capa de negocio. Su función es devolver al usuario un listado de los Indicadores según los parámetros de entrada.	
Parámetros de Entrada:	
categoria	int
descripcion	String
offset	int
cantidad	int
tipo_salida	int
Ordenar	String
Orden	String

Tabla 3.21 Descripción de la página servidora RICAP.Listar_Indicador.

Nombre: RICAP.Listar_Conducta	
Tipo de clase: página servidora	
Descripción General: La clase RICAP.Listar_Conducta es una clase que se ejecuta del lado del servidor en la capa de negocio. Su función es devolver al usuario un listado de las Conductas según los parámetros de entrada.	
Parámetros de Entrada:	
categoria	int
descripcion	String
offset	int
cantidad	int
tipo_salida	int

Ordenar	String
Orden	String

Tabla 3.22 Descripción de la página servidora RICAP.Listar_Conducta.

Nombre: RICAP.Modificar_Categoría	
Tipo de clase: página servidora	
Descripción General: La clase RICAP.Modificar_Categoría es una clase que se ejecuta del lado del servidor en la capa de negocio. Su función es modificar en la base de datos una Categoría según los parámetros de entrada.	
Parámetros de Entrada:	
descripción	string
id_seleccionado	int

Tabla 3.23 Descripción página servidora RICAP.Modificar_Categoría.

Nombre: RICAP.Modificar_Indicador	
Tipo de clase: página servidora	
Descripción General: La clase RICAP.Modificar_Indicador es una clase que se ejecuta del lado del servidor en la capa de negocio. Su función es modificar en la base de datos un Indicador según los parámetros de entrada.	
Parámetros de Entrada:	
descripción	string
id_indicador	int
arreglo_id	array

Tabla 3.24 Descripción de la página servidora RICAP.Modificar_Indicador.

Nombre: RICAP.Modificar_Conducta	
Tipo de clase: página servidora	
Descripción General: La clase RICAP.Modificar_Conducta es una clase que se ejecuta del lado del servidor en la capa de negocio. Su función es modificar en la base de datos una Conducta según los parámetros de entrada.	
Parámetros de Entrada:	
descripción	string
id_conducta	int

arreglo_id	array
------------	-------

Tabla 3.25 Descripción de la página servidora RICAP.Modificar_Conducta.

Nombre: RICAP.Eliminar_Categoría	
Tipo de clase: página servidora	
Descripción General: La clase RICAP.Eliminar_Categoría es una clase que se ejecuta del lado del servidor en la capa de negocio. Su función es eliminar de la base de datos la Categoría según el parámetro de entrada.	
Parámetros de Entrada:	
id_categoría	int

Tabla 3.26 Descripción de la página servidora RICAP.Eliminar_Categoría.

Nombre: RICAP.Eliminar_Indicador	
Tipo de clase: página servidora	
Descripción General: La clase RICAP.Eliminar_Indicador es una clase que se ejecuta del lado del servidor en la capa de negocio. Su función es eliminar de la base de datos del Indicador según el parámetro de entrada.	
Parámetros de Entrada:	
id_indicador	int

Tabla 3.27 Descripción de la página servidora RICAP.Eliminar_Indicador.

Nombre: RICAP.Eliminar_Conducta	
Tipo de clase: página servidora	
Descripción General: La clase RICAP.Eliminar_Conducta es una clase que se ejecuta del lado del servidor en la capa de negocio. Su función es eliminar de la base de datos la Conducta según el parámetro de entrada.	
Parámetros de Entrada:	
id_conducta	int

Tabla 3.28 Descripción de la página servidora RICAP.Eliminar_Conducta.

Nombre: RICAP.Insertar_Categoría
Tipo de Clase: página servidora

Descripción General: La clase RICAP.Insertar_Categoría es una clase que se ejecuta del lado del servidor en la capa de negocio. Su función es insertar en la base de datos los datos de la categoría.

Parámetros de Entrada	
Descripción	Tipo
descripcion	string
id_categoria	int

Tabla 3.29 Descripción de la página servidora RICAP.Insertar_Categoría.

Nombre: RICAP.Insertar_Indicador

Tipo de Clase: página servidora

Descripción General: La clase RICAP.Insertar_Indicador es una clase que se ejecuta del lado del servidor en la capa de negocio. Su función es insertar en la base de datos los datos del Indicador.

Parámetros de Entrada	
Descripción	Tipo
descripcion	string
categoria	array

Tabla 3.30 Descripción de la página servidora RICAP.Insertar_Indicador.

Nombre: RICAP.Insertar_Conducta

Tipo de Clase: página servidora

Descripción General: La clase RICAP.Insertar_Conducta es una clase que se ejecuta del lado del servidor en la capa de negocio. Su función es insertar en la base de datos los datos de la conducta.

Parámetros de Entrada	
Descripción	Tipo
descripcion	string
categoria	array

Tabla 3.31 Descripción de la página servidora RICAP.Insertar_Conducta.

3.2.5 Diagrama de clases persistentes.

En el diagrama de clases persistentes sólo aparecen las clases persistentes. Estas son las clases capaces de mantener su valor en el espacio y en el tiempo. A continuación se muestra el diagrama de clases persistentes del Registro de Indicadores y Conductas Asistenciales.

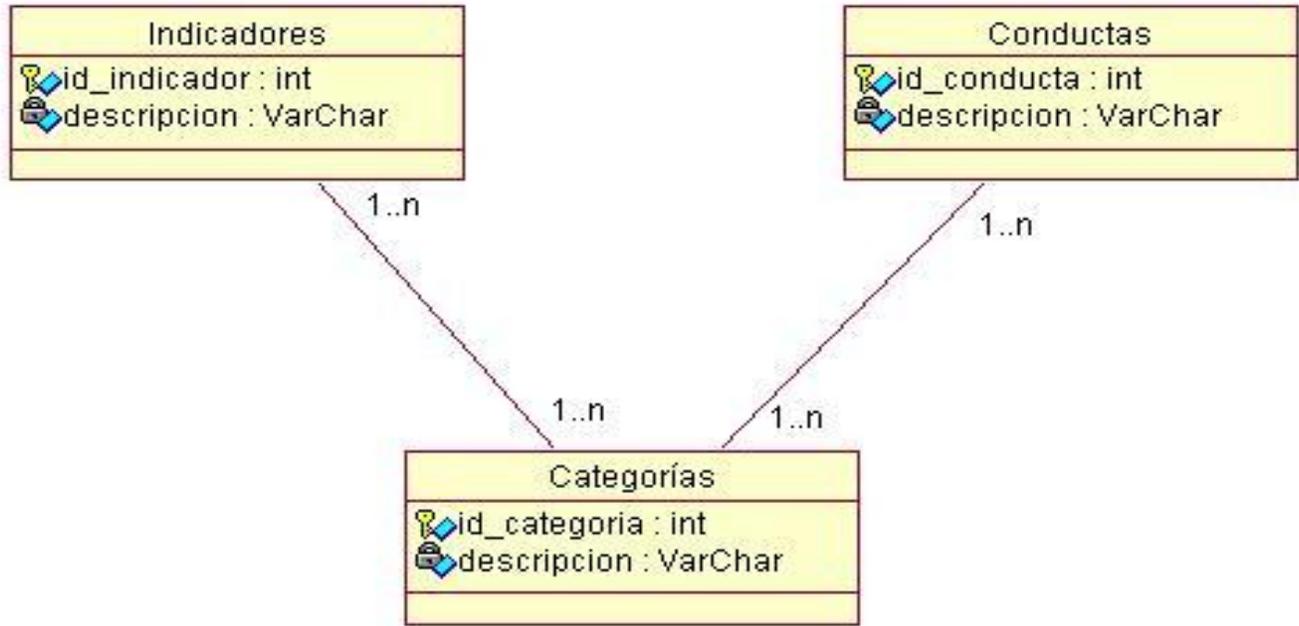


Figura 3.5 Diagrama de Clases Persistentes.

3.2.6 Modelo de Datos

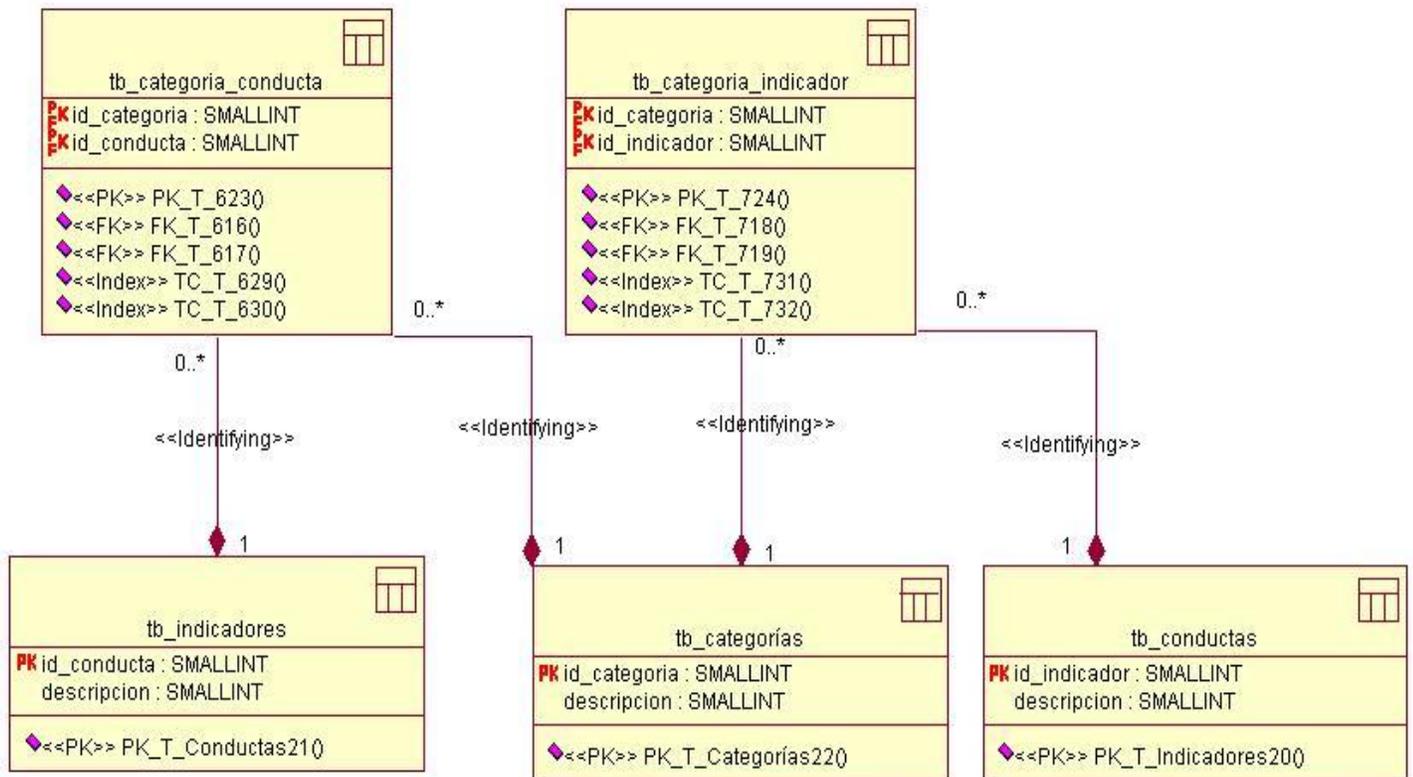


Figura 3.6 Diagrama de Modelo de Datos.

3.2.7 Descripción de tablas y atributos de la Base de Datos.

Nombre: tb_indicadores		
Descripción: Tabla que almacenará los datos de todos los Indicadores que existan.		
Atributo	Tipo	Descripción
id_indicador	INT	Identificador del Indicador.(PK)
descripcion	VARCHAR(100)	Nombre del Indicador.

Nombre: tb_conductas		
Descripción: Tabla que almacenará los datos de todas las Conductas que existan.		
Atributo	Tipo	Descripción

id_conducta	INT	Identificador de la Conducta.(PK)
descripcion	VARCHAR(100)	Nombre de la Conducta.

Nombre: tb_categorías		
Descripción: Tabla que almacenará los datos de todas las Categorías que existan.		
Atributo	Tipo	Descripción
id_categoria	INT	Identificador de la Categoría.(PK)
descripcion	VARCHAR(100)	Nombre de la Categoría.

Nombre: tb_categoria_indicador		
Descripción: Almacena la relación de las categorías de cada indicador, guardando sus identificadores.		
Atributo	Tipo	Descripción
id_categoria	INT	Identificador de la Categoría en la tabla tb_categorías. (PK)(FK).
id_indicador	INT	Identificador del Indicador en la tabla tb_indicadores. (PK)(FK)

Nombre: tb_categoria_conducta		
Descripción: Almacena la relación de las categorías de cada conducta, guardando sus identificadores.		
Atributo	Tipo	Descripción
id_categoria	INT	Identificador de la Categoría en la tabla tb_categorías. (PK)(FK).
id_conducta	INT	Identificador de la Conducta en la tabla tb_conductas. (PK)(FK)

En este capítulo se describió la estructura arquitectónica propuesta para la solución. Quedando fundamentados los patrones definidos para el modelado e implementación del sistema, contribuyendo a la reutilización y a una mayor comprensión del mismo. Se detallaron los elementos del diseño, escribiéndose las clases y sus relaciones; así como el modelo de datos, describiéndose cada una de las tablas de la base de datos. Con el diseño propuesto, se materializan con precisión los requerimientos definidos para el sistema, proporcionando la idea de lo que se quiere construir, sirviendo como guía a los desarrolladores para la futura implementación.

Capítulo 4: Implementación y Pruebas.

El propósito fundamental de este capítulo es describir el proceso de implementación en términos de componentes a partir del diseño. En el mismo se hará una breve descripción de los métodos, los estándares de diseño, codificación y tratamiento de errores a utilizar. También se definirán y se representarán los diagramas de despliegue y componente quedando conformado el modelo de implementación del sistema.

4.1 Modelo de Implementación.

Los diagramas de despliegue y componentes conforman lo que se conoce como el Modelo de Implementación al describir los componentes a construir y su organización y dependencia entre nodos físicos en los que funcionará a aplicación. Estos componentes incluyen: ficheros ejecutables, ficheros de código fuente, y todo otro tipo de ficheros necesarios para la implantación y despliegue del sistema. Este modelo describe la relación y organización de los componentes de acuerdo con los mecanismos de estructuración y modulación disponibles en el entorno de implementación y en el lenguaje o lenguajes de programación utilizados.

El modelo de implementación del módulo RICAP expone una organización en capas y jerarquías de paquetes y subsistemas de implementación que contienen componentes y sus relaciones, dividiendo al sistema en trozos más manejables, lo que posibilitará la reutilización, que se pueda implementen por separado y disminuirá el impacto que pueda traer consigo un cambio. A continuación se muestra una vista global de la estructura y organización de la implementación del sistema:

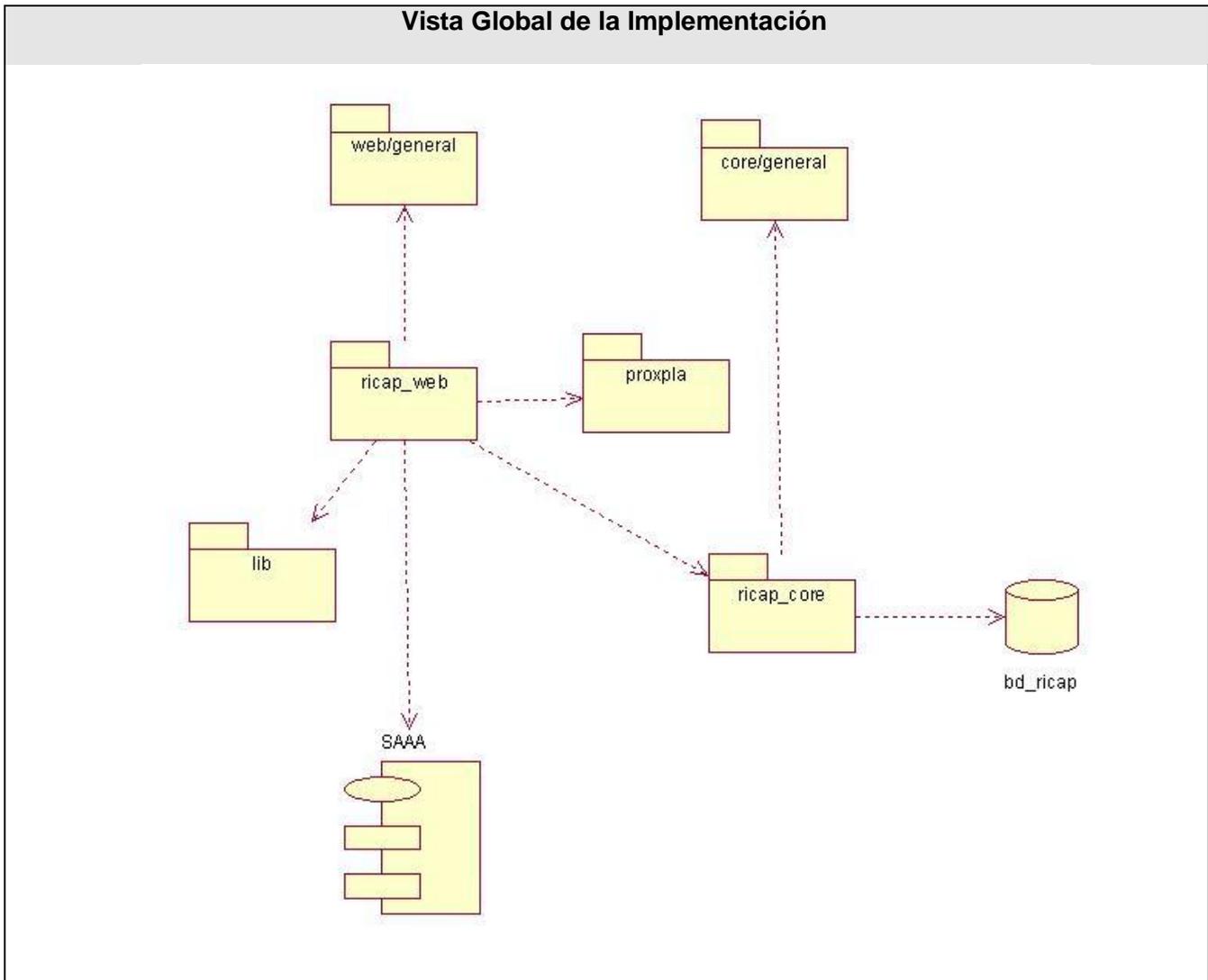


Figura 4.1 Vista Global de la Implementación.

4.1.1 Diagramas de Componentes.

Un componente es el empaquetamiento físico de los elementos de un modelo, como son las clases del modelo de diseño, los mismos son creados, modificados o eliminados en el proceso de implementación y constituyen la versión del producto. Los diagramas de componentes se usan para describir la organización y dependencias entre un conjunto de componentes.

El siguiente diagrama muestra diferentes componentes, separados por capas y sus relaciones, del Caso de Uso Listar Categorías.

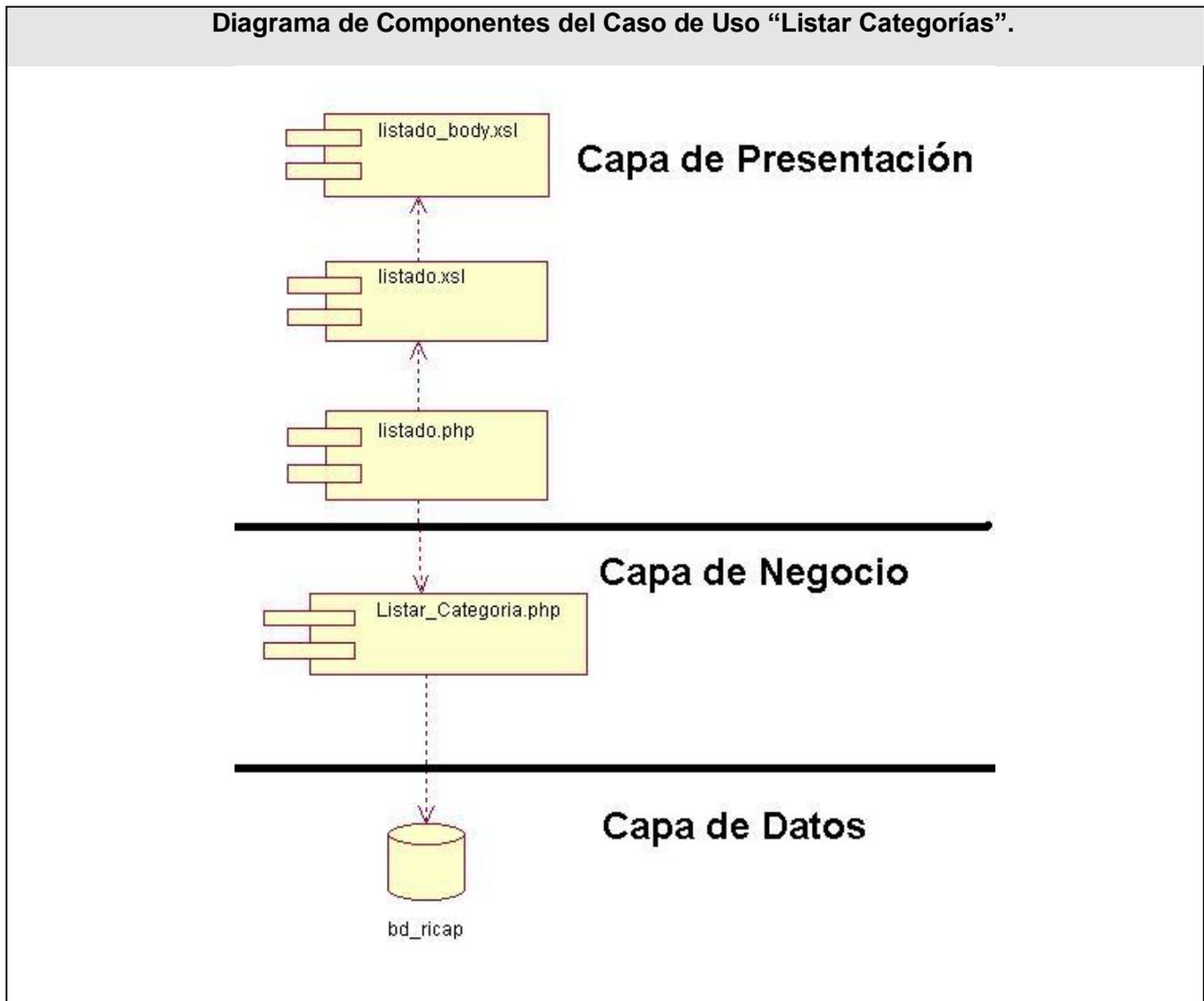


Figura 4.2 Diagrama de Componentes del Caso de Uso “Listar Categorías”.

4.1.2 Diagrama de Despliegue.

El diagrama de despliegue está compuesto por dispositivos, procesadores y protocolos. Permite comprender la correspondencia entre la arquitectura software y la arquitectura hardware. El diagrama de

despliegue describe la distribución física los componentes de software entre los distintos nodos de cómputo.

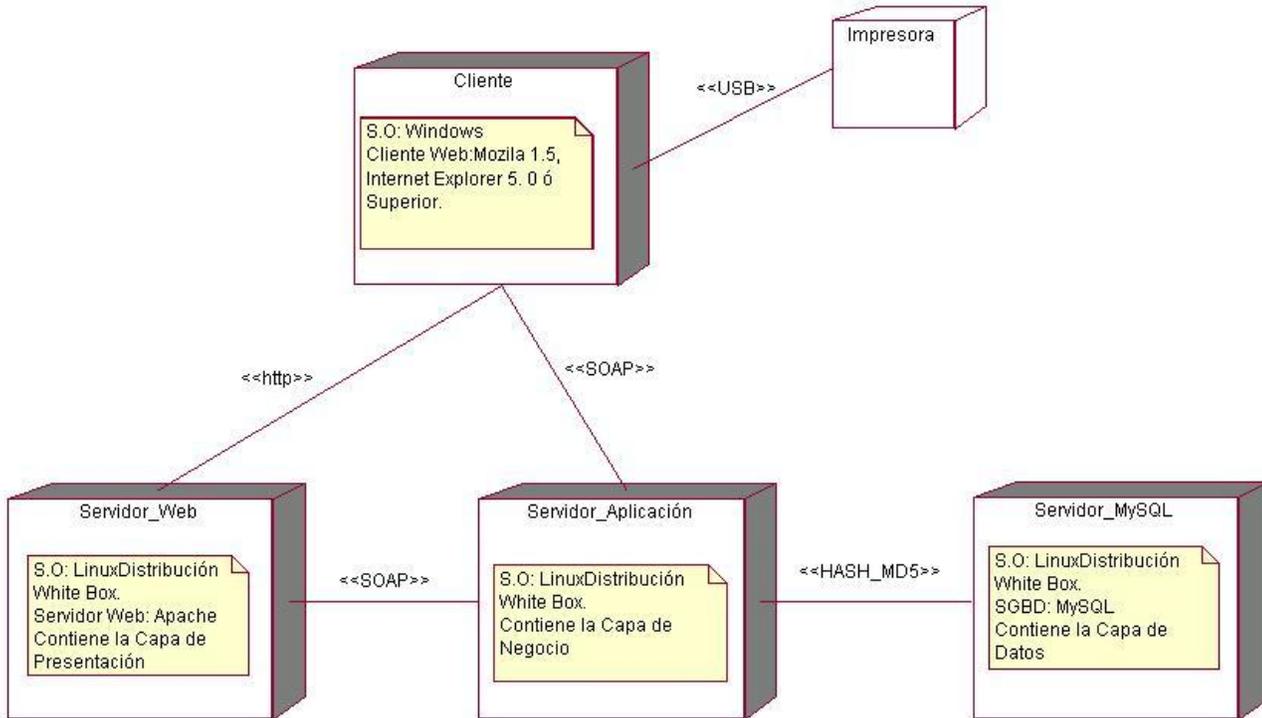


Figura 4.3 Diagrama de Despliegue.

4.2 Descripción de los Métodos.

A continuación se presenta la descripción de algunos de los métodos del negocio más complejos del Registro de Indicadores y Conductas Asistenciales para la Atención Primaria.

Nombre: Insertar_Indicador.php.	
Descripción General: Este método se ejecuta del lado servidor en la Capa de Negocio. Establece una conexión con el servidor de base de datos a través de la clase de acceso a datos dbz_class. Este método tiene la función de ejecutar las consultas necesarias para insertar la información de un indicador introducida por el funcionario de la Dirección Nacional de la Atención Primaria. Terminada la consulta SQL, se cierra la conexión con la base de datos para no sobrecargar el servidor.	
Parámetros de Entrada	
Descripción	Tipo
categoría	array
descripción	string

Tabla 4.1 Descripción del método Insertar_Indicador.php.

Nombre: Listar_Indicador.php.	
Descripción General: Este método se ejecuta del lado servidor en la Capa de Negocio. Establece una conexión con el servidor de base de datos a través de la clase de acceso a datos dbz_class. Este método tiene la función de ejecutar las consultas necesarias para retornar un listado con la información existente de los Indicadores según los parámetros de entrada. Este método permitirá conocer, dado un id de Categoría o un descripción del indicador, los datos referentes a los Indicadores que cumplan con los criterios de búsqueda especificados. Su implementación permite retornar documentos en formatos XML, Acrobat Reader (PDF) y Microsoft Excel (XSL). Terminada la consulta SQL, se cierra la conexión con la base de datos para no sobrecargar el servidor.	
Parámetros de Entrada	
Descripción	Tipo
id_categoría	int

descripción	string
-------------	--------

Tabla 4.2 Descripción del método Listar_Indicador.php.

4.3 Estándares de diseño, codificación y tratamiento de errores.

Con el objetivo de que exista una homogeneidad entre las aplicaciones que se encuentren integradas a los distintos sistemas informatizados de salud, en el proyecto APS se han definido una serie de estándares y tratamiento de errores para que los códigos fuentes de las aplicaciones y los mensajes que se emitan sean semejantes para sus futuras modificaciones.

Estándares de diseño.

Para el diseño de las páginas web del Módulo RICAP se siguieron las pautas y el mismo diseño que el Sistema de Información para la Salud (SISalud); esto trae como principal ventaja la uniformidad en la estructura de las páginas web de todos los módulos integrados al SISalud. Para lograr esta uniformidad se debe de aplicar una hoja de estilo en cascada (Cascading Style Sheets CSS), todas las páginas deben tener una resolución de 800 x 600 pixel, incluir una serie de imágenes estándares que define el SISalud y seguir una serie de pautas entre las que se encuentran:

- El menú principal siempre estará situado en una barra superior horizontal de solo 15 px de altura. No existirá barra vertical de menú situada a la izquierda de la página (como usualmente se hace) para ampliar el espacio de trabajo, pues estará reservado lo más amplio posible para la inserción de grandes tablas y formularios que constituyen la base fundamental de estas aplicaciones.
- El logo siempre estará ubicado en el extremo superior izquierdo de la página, es una imagen que cuenta con un ancho de 270 px y se corresponde con el nombre de cada módulo. Estará constituido por un juego tipográfico en Frankling Gothic Medium, y en el caso de las aplicaciones propias del Proyecto APS, estando especificado dentro del logo como una especie de genérico.
- Bajo el logo existirá una barra de ubicación dentro del sitio, funcionando como hipervínculo, que servirá como referencia para saber donde se encuentra el usuario o para acceder rápidamente a cualquiera de los niveles superiores de navegación dentro de los que se encuentra. Además se

encontrará destacado dentro del menú principal (con un destaque en el color secundario) en cual de los elementos del menú se encuentra el usuario en ese momento.

- La tipografía será siempre Tahoma, por su amplia legibilidad y por las facilidades conocidas que brinda para la lectura digital. El menú principal será a 7 ptos y los submenús a 6 ptos. Los demás puntajes se definirían en dependencia de las necesidades puntuales de cada pantalla.
- El espacio de trabajo comienza 33 px por debajo del menú. El espacio intermedio que queda es también con fondo blanco y está reservado para el texto de ubicación dentro del sitio (justificado a la izquierda) y para ubicar los botones propios de la pantalla (justificados a la derecha). Estos se organizarán en una o dos filas, de hasta cuatro botones (13 x 72 px) cada una. Los botones se corresponden también con los colores pautados.
- Entre los elementos comunes del menú principal se encuentran Inicio para regresar a la página inicial del módulo, Salir para desconectarse del sistema, y Otros Módulos para facilitar los enlaces a otros módulos necesarios. Son también comunes a casi todos los botones del menú principal Configurar para la configuración de codificadores, Cierre para la realización de cierre estadístico y Reportes para generar reportes de actividades u operaciones.
- Es común para todos los módulos el diseño de una serie de ventanas, en las que solo cambiarían los colores, en dependencia de cada uno. Son estas las ventanas de precaución, error, validación de datos, etc.
- En cuanto a los elementos de diseño del interior de las pantallas, es decir, de las tablas, formularios, etc., se definen los edit que se utilicen con una altura de 16 px y la separación entre estos y entre ellos y los bordes de tablas será de 8 px. Será de 8 px la separación entre el texto y el edit. Los textos de estos campos serán justificados siempre a la derecha, es decir, justificados a 8 ptos de cada edit.
- En el caso de tablas generadas por búsquedas, que ordenan una serie de elementos, y necesiten selección, se harán a través de checkboxes justificados a la izquierda de la tabla. Siempre habrá un checkbox en la fila de título, también a la izquierda, que facilite seleccionar todos. Es necesario

destacar que estas tablas pueden tener una cantidad grande de líneas generadas por la búsqueda, por lo que debe quedar pautado que hasta 25 resultados la tabla funcione con scroll, pero más de esta cantidad será entonces por paginado, al estilo de Google, con 25 resultados por página.

- Para mostrar un listado en una tabla se hace de esta forma. La primera fila de la tabla es de estilo **encabezadotabla**, la última es de estilo: **pietabla** y letra **Tahoma** de color negro (**#000000**), y en la parte de abajo a la derecha se ponen la cantidad de páginas.

☒ Listado - Categorías

Descripción	Código
<input type="radio"/> De 0 hasta 19 años	1
<input type="radio"/> Embarazadas	3
<input type="radio"/> Mayores de 20 años	4
<input type="radio"/> ancianos	5

1 - 4 / 4

- En el caso que el resultado de la búsqueda sea mayor que la cantidad deseada en una misma página se realizará de la siguiente forma:

🔍 Listado - Indicadores de Salud

🔍 **Buscar por:**

Descripción Categoría Embarazadas

Ordenar por Descripción

Exportar a **Página #** 1

Nombre	Categoría
<input type="radio"/> <u>Accidente en el hogar</u>	De 0 hasta 19 años Embarazadas Mayores de 20 años
<input type="radio"/> <u>Chequeo pre empleo</u>	De 0 hasta 19 años Embarazadas
<input type="radio"/> <u>Embarazada mayor de 35 años</u>	Embarazadas
<input type="radio"/> <u>Embarazada menor de 20 años</u>	Embarazadas
<input type="radio"/> <u>Embarazada no captada precozmente</u>	Embarazadas
<input type="radio"/> <u>Embarazada no captada precozmente tardía</u>	Embarazadas
<input type="radio"/> <u>Embarazadas con 11g hemoglobina</u>	Embarazadas
<input type="radio"/> <u>Embarazadas con ganancia de peso 8kg</u>	Embarazadas
<input type="radio"/> <u>Estado Nutricional Bajo Peso</u>	Embarazadas
<input type="radio"/> <u>OBESIDAD</u>	De 0 hasta 19 años Embarazadas Mayores de 20 años

1 - 10 / 11

Exportar a **Página #** 1

Existen detalles que serán definidos exclusivamente en cada módulo, ya que satisfacen a necesidades concretas de los mismos.

El interés general es mantener el diseño y la estructura del sitio lo más simple posible, la simplicidad es entendimiento del contenido, de la estructura, es facilidad para encontrar lo que se busca, es también velocidad de descarga.

Estándares de codificación.

Con el propósito de distribuir los esfuerzos y mejorar los rendimientos de la aplicación, se utilizó la programación orientada a servicios, creando clases genéricas que permiten la definición y distribución de las llamadas a los procedimientos de los módulos distribuidos, la definición de los métodos propios de cada módulo y el manejo de las bases de datos.

Actualmente se hallan estándares de codificación para la mayoría de los lenguajes existentes. El uso de ellos partiendo de las convenciones definidas permite una mejor comunicación entre los programadores creando las condiciones para la reusabilidad y el mantenimiento de los sistemas. Para definir el estilo de codificación a seguir en la aplicación se utilizó la notación estándar establecida para aplicaciones desarrolladas en PHP (PHP Coding Standard), que mayormente está basada en el estándar de código para aplicaciones en C++ (C++ Coding Standard).

- ✓ Las etiquetas de apertura y cierre del lenguaje serán de la forma `<?php ?>`, ya que siempre están disponible en cualquier configuración.
- ✓ Se harán uso de los arreglos predefinidos para el manejo de los valores enviados por el usuario `$_GET`, `$_POST`, `$_FILES` evitando el uso de `$_REQUEST`.
- ✓ Para nombrar las variables se seguirá la regla de escribir los identificadores con letras minúsculas y en español, utilizando como separador para las palabras el carácter “_” tratando de usar nombres sugerentes a la acción de la variable.
- ✓ Todos los campos id van a comenzar con el identificador (id) seguido del nombre del campo. Ejemplo `id _ indicador`.
- ✓ Los arreglos empezarán con el identificador array y las palabras no se separaran con el carácter “_”.
- ✓ En el caso de las clases se pondrá delante la letra C. Ejemplo: `CFachada` y en el de los métodos no se usarán abreviaturas y las palabras continuas deben comenzar con mayúsculas. Ejemplo: `ListarIndicadores`.
- ✓ Para comentar el código se utilizará, en el caso de una línea, al final de la misma el carácter `“//”` y seguido el comentario y en el caso de un bloque se utilizará los caracteres `“/* */”`.
- ✓ El idioma de las clases auxiliares como sesión y error, será el inglés para garantizar la homogeneidad con las programadas en este ámbito en el mundo, en el caso de los Servicios Web y la interface de administración se usará el español para esclarecer los objetivos de cada método o script a utilizar.

- ✓ Se utilizará el operador: ? para sentencias cortas, preferiblemente que ocupen una sola línea. La sentencia switch siempre tendrá la opción default y se evitará el uso de continue y break, ya que podrían perder la vista lógica del código fuente.
- ✓ El almacenamiento de la información será en scripts SQL para construir la base de datos e interactuar con ella desde las aplicaciones.
- ✓ Las palabras correspondientes a las sentencias SQL y sus parámetros deben ir en mayúsculas
Ej.: SELECT * FROM tb_Indicadores
- ✓ En las consultas de inserción se debe colocar siempre el nombre de los campos en los cuales desea escribir en la tabla.
- ✓ Los nombres de las tablas deben ir en minúsculas y cada palabra separada por línea abajo "_".
(Ejemplo: Id_nombre_tabla)
- ✓ En el caso de los XSL será con el mismo nombre que el fichero de la capa de presentación.
- ✓ Las páginas HTML se harán sin incluir código y todas las funciones JavaScript que se usarán se escribirán dentro de ficheros *js*.
- ✓ Para la capa de datos se tiene que nombrar la base de datos poniendo el identificador del proyecto APS seguido del carácter _ y del nombre del módulo. Ejemplo: bd_ricap
- ✓ Los controles seguirán el siguiente tratamiento:

Control	Prefijo	Ejemplo
Botón	Btn	btnAceptar
Etiqueta	Lbl	lblNombre
Lista/Menú	Mn	mnPrincipal
Campo de Texto	Txt	txtFecha
Botón de Opción	Opt	optSexo
Casilla de Verificación	Chx	chxBorrar

Tabla 4.3 Estándares para los controles

- ✓ Las tablas se identificarán con el acrónimo tb_indicadores, ejemplo: tb_conductas.
- ✓ Los campos de la base de datos se nombrarán igual que las variables.
- ✓ Cada módulo definirá un espacio de nombre (namespace), siguiendo la siguiente estructura

<http://APS/RICAP/NombreMetodo> .

Tratamiento de errores.

Se define como excepción a un evento que ocurre durante la ejecución del programa que impide el flujo normal de las sentencias y que requieren recursos específicos para su control. Las excepciones son una forma bastante legible para controlar los errores sin confundir el código con muchas instrucciones de control del error. Cuando se verifica un error se pone en marcha una excepción que, siendo recibida enseguida, permite gestionar un error.

Durante la ejecución, en las clases pueden provocarse errores de diferentes tipos y diversos grados de gravedad. Cuando se invocan métodos sobre un objeto, se puede encontrar con problemas internos de estado (valores incongruentes), detectar errores con los objetos o datos que manipula (como la dirección a un archivo o red), querer acceder sobre un archivo ya cerrado u otros problemas. Proporcionan una manera de verificar los errores y poder controlarlos si fuera el caso si abortar el código.

La correcta programación de excepciones significa diseñar los algoritmos pensando únicamente en la forma habitual en la que deben ejecutarse, manejando las situaciones extraordinarias a parte. De esta manera se consigue un diseño mucho más estructurado, legible, robusto y fácil de mantener.

Para depurar los errores se utilizará JavaScript. Por medio de este lenguaje serán informados la mayoría de los errores de la página, como apoyo a las validaciones de entrada de datos, garantizando que los datos introducidos por los usuarios sean validos, o les sea posible corregirlos.

Otros errores en la capa de negocio serán tratados devolviendo un SOAP_FAULT, cuyos elementos FaultCode, FaultString, FaultAutor se describen a continuación:

FaultCode:

Código de texto utilizado para indicar la clase de error, codificado de la siguiente manera.

Código del proyecto-código del modulo (:) número del método (.) número del error. Ejemplo: APS-RICAP:

1.5 que indica error 5 en el método 1 del módulo Registro de Indicadores y Conductas Asistenciales para Atención Primaria de la Salud perteneciente al Proyecto APS.

FaultString:

Una explicación del error asequible al humano (leíble). Debe tenerse en cuenta que este texto puede ser mostrado al operador final del sistema. Ejemplo:



FaultAutor:

Un texto que indica quién provocó el error, siempre será el nombre del método que eleva la excepción.

Los errores que pueden evitarse en la capa de presentación, serán tratados mediante funciones del lenguaje Java Script *del lado del cliente*, a través de mensajes de alerta.



Figura 4.4 Ejemplo de un mensaje de alerta.

También serán tratadas las validaciones a la hora de eliminar algún elemento del lado del cliente, mostrado en el navegador una ventana de si está seguro queriendo eliminar el elemento seleccionado. Si se acciona el botón *Aceptar* se procede a la acción, de lo contrario no se ejecuta.

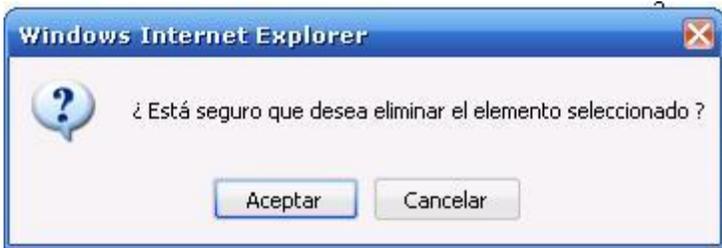


Figura 4.5 Ejemplo de un mensaje de alerta al eliminar un elemento.

4.5 Prueba.

Las pruebas es una actividad en la cual un sistema o componente es ejecutado bajo unas condiciones o requerimientos específicos, los resultados son observados y registrados, y una evaluación es hecha de algún aspecto del sistema o componente.

El Flujo Trabajo de Prueba tiene entre sus principales objetivos encontrar y documentar los defectos que puedan afectar la calidad del software, validar que el software trabaje como fue diseñado, validar y probar los requisitos que debe cumplir el software, validar que los requisitos fueron implementados correctamente.

Algunos de los métodos de pruebas mas importantes son: el método de caja negra y el de caja blanca. A la aplicación le realizaron las pruebas de caja negra.

4.5.1 Métodos de Prueba.

Pruebas de Caja Negra.

Las pruebas de caja negra se refieren a las pruebas que se llevan a cabo sobre la interfaz del software. O sea, los casos de prueba pretenden demostrar que las funciones del software son operativas, que la entrada se acepta de forma adecuada y que se produce un resultado correcto, así como que la integridad de la información externa se mantiene.

4.5.1.1 Modelo de Prueba

Nombre del caso de uso: Gestionar Indicador

Nombre del caso de prueba: "Insertar Indicador".

Entrada	Resultados	Condiciones
Se seleccionó la opción	Se insertó la información en la base	El usuario autenticado es el editor de la

Nuevo de la etiqueta Indicadores de salud en el menú y se llena el formulario.	de datos.	Dirección Nacional de la Atención Primaria.
--	-----------	---

Tabla 4.4 Caso de prueba "Insertar Indicador" del caso de uso Gestionar Indicador.

Nombre del caso de prueba: "Eliminar Indicador".

Entrada	Resultados	Condiciones
Después de mostrar el listado se seleccionó el indicador a eliminar y se da clic en el Botón Eliminar.	Se elimina en la base de datos el indicador, se actualiza la misma y se regresa a la página donde se muestra el listado.	El usuario autenticado es el editor de la Dirección Nacional de la Atención Primaria.

Tabla 4.5 Caso de prueba "Eliminar Indicador" del caso de uso Gestionar Indicador.

Nombre del caso de prueba: "Listar Indicador".

Entrada	Resultados	Condiciones
Se seleccionó la opción Listado de la etiqueta Indicadores de salud en el menú y se introducen los parámetros de búsqueda si es necesario.	Se mostró un listado de la información de la base de datos	El usuario autenticado es el editor de la Dirección Nacional de la Atención Primaria.

Tabla 4.6 Caso de prueba "Listar Indicador" del caso de uso Gestionar Indicador.

Nombre del caso de prueba: "Modificar Indicador".

Entrada	Resultados	Condiciones
Después de mostrar el	Se actualiza en la base de datos el	El usuario autenticado es el editor de la

listado se seleccionó el indicador a modificar dando clic en el vinculo de dicho indicador.	indicador y se regresa a la página donde se muestra el listado.	Dirección Nacional de la Atención Primaria.
---	---	---

Tabla 4.7 Caso de prueba “Modificar Indicador” del caso de uso Gestionar Indicador.

Nombre del caso de uso: Listar Indicadores.

Nombre del caso de prueba: “Listar Indicadores”.

Entrada	Resultados	Condiciones
En las Opciones de Menú escoge la etiqueta Indicadores de salud.	Se mostró un listado de la información de la base de datos	El usuario autenticado es un usuario de la Atención Primaria de la Salud

Tabla 4.8 Caso de prueba “Listar Indicador” del caso de uso Listar Indicadores.

Nombre del caso de uso: Gestionar Conducta.

Nombre del caso de prueba: “Insertar Conducta”.

Entrada	Resultados	Condiciones
Se seleccionó la opción Nuevo de la etiqueta Conductas o Accionar en el menú y se llena el formulario.	Se insertó la información en la base de datos.	El usuario autenticado es el editor de la Dirección Nacional de la Atención Primaria.

Tabla 4.9 Caso de prueba “Insertar Conducta” del caso de uso Gestionar Conducta.

Nombre del caso de prueba: “Listar Conducta”.

Se seleccionó la opción Listado de la etiqueta Conductas o Accionar en el menú y se introducen los parámetros de búsqueda si es necesario.	Se mostró un listado de la información de la base de datos	El usuario autenticado es el editor de la Dirección Nacional de la Atención Primaria.
--	--	---

Tabla 4.10 Caso de prueba “Listar Conducta” del caso de uso Gestionar Conducta.

Nombre del caso de prueba: “Modificar Conducta”.

Después de mostrar el listado se seleccionó la conducta a modificar dando clic en el vinculo de dicha conducta.	Se actualiza en la base de datos la conducta y se regresa a la página donde se muestra el listado.	El usuario autenticado es el editor de la Dirección Nacional de la Atención Primaria.
---	--	---

Tabla 4.11 Caso de prueba “Modificar Conducta” del caso de uso Gestionar Conducta.

Nombre del caso de prueba: “Eliminar Conducta”.

Después de mostrar el listado se seleccionó la conducta a eliminar y se da clic en el Botón Eliminar.	Se elimina en la base de datos la conducta, se actualiza la misma y se regresa a la página donde se muestra el listado.	El usuario autenticado es el editor de la Dirección Nacional de la Atención Primaria.
---	---	---

Tabla 4.12 Caso de prueba “Eliminar Conducta” del caso de uso Gestionar Conducta.

Nombre del caso de uso: Listar Conductas.

Nombre del caso de prueba: "Listar Conductas".

Entrada	Resultados	Condiciones
En las Opciones de Menú escoge la etiqueta Conductas o Accionar.	Se mostró un listado de la información de la base de datos	El usuario autenticado es un usuario de la Atención Primaria de la Salud

Tabla 4.13 Caso de prueba "Listar Conducta" del caso de uso Listar Conductas.

Nombre del caso de uso: Gestionar Categoría.

Nombre del caso de prueba: "Insertar Categoría".

Entrada	Resultados	Condiciones
Se seleccionó la opción Nuevo de la etiqueta Categorías en el menú y se llena el formulario.	Se insertó la información en la base de datos.	El usuario autenticado es el editor de la Dirección Nacional de la Atención Primaria.

Tabla 4.14 Caso de prueba "Insertar Categoría" del caso de uso Gestionar Categoría.

Nombre del caso de prueba: "Listar Categoría".

Se seleccionó la opción Listado de la etiqueta Categorías en el menú.	Se mostró un listado de la información de la base de datos	El usuario autenticado es el editor de la Dirección Nacional de la Atención Primaria.
---	--	---

Tabla 4.15 Caso de prueba "Listar Categoría" del caso de uso Gestionar Categoría.

Nombre del caso de prueba: "Modificar Categoría".

Después de mostrar el listado se seleccionó la categoría a modificar dando clic en el vinculo	Se actualiza en la base de datos la categoría y se regresa a la página donde se muestra el listado.	El usuario autenticado es el editor de la Dirección Nacional de la Atención Primaria.
---	---	---

de dicha categoría.		
---------------------	--	--

Tabla 4.16 Caso de prueba “Modificar Categoría” del caso de uso Gestionar Categoría.

Nombre del caso de prueba: “Eliminar Categoría”.

Después de mostrar el listado se seleccionó la categoría a eliminar y se da clic en el Botón Eliminar.	Se elimina en la base de datos la categoría, se actualiza la misma y se regresa a la página donde se muestra el listado.	El usuario autenticado es el editor de la Dirección Nacional de la Atención Primaria.
--	--	---

Tabla 4.17 Caso de prueba “Eliminar Categoría” del caso de uso Gestionar Categoría.

Nombre del caso de uso: Listar Categorías.

Nombre del caso de prueba: “Listar Categorías”.

Entrada	Resultados	Condiciones
En las Opciones de Menú escoge la etiqueta Categorías.	Se mostró un listado de la información de la base de datos	El usuario autenticado es un usuario de la Atención Primaria de la Salud

Tabla 4.18 Caso de prueba “Listar Categoría” del caso de uso Listar Categoría.

En el capítulo se mostraron los resultados obtenidos de la etapa de implementación del sistema. Como solución al diseño se presentó el modelo de implementación donde se conformaron los diagramas de componentes y despliegue para describir la distribución física del sistema y sus componentes. Se detallan los métodos de implementación más complejos, así como el tratamiento de errores, los principios de codificación y de diseño empleados en el proceso de implementación.

Conclusiones

Con el interés de dar cumplimiento a los objetivos y tareas trazados:

- ✓ Se realizó un estudio detallado del proceso de gestión de la información de los Indicadores y Conductas Asistenciales en el Sistema Nacional de Salud.
- ✓ Se investigó sobre sistemas existentes a nivel internacional y nacional, que pudieran resolver el problema planteado, realizando un análisis comparativo entre los mismos. Además se estudiaron las tendencias, tecnologías y herramientas a utilizar para el desarrollo de la solución propuesta.
- ✓ Se documentaron los flujos de trabajo: Modelamiento de Negocio, Requerimientos, Diseño e Implementación.
- ✓ Se diseñó una aplicación que cumple con los requerimientos trazados y con la arquitectura definida por el MINSAP.

Por todo lo expuesto anteriormente, se cumplieron los objetivos del trabajo logrando implementar una aplicación web, que permite búsquedas dinámicas, gestionar la información y brindar servicios a las demás aplicaciones integradas al SISalud.

Recomendaciones

Una vez cumplido con el objetivo general propuesto, se recomienda:

- ✓ Publicar los servicios web de RICAP para que puedan ser consumidos por cualquier aplicación informática que necesite de los mismos.
- ✓ Implementar la funcionalidad que permita supervisar que la información que consuman otros módulos del SISalud correspondiente a los indicadores de salud y conductas asistenciales no pueda ser eliminada.
- ✓ Incluir en la concepción del componente de avisos de SISalud los mensajes generados por actualizaciones que se realicen en el RICAP.

Referencias Bibliográficas

- [1]. cubasalud. (29 de 11 de 2006). Recuperado el 20 de 12 de 2007, de <http://cubasalud.blogia.com>
- [2]. George Eisen, M. (dic. de 1996). *scielo.sld.cu*. Recuperado el 20 de 12 de 2007, de http://scielo.sld.cu/scielo.php?pid=s0864-34661996000200003&script=sci_arttext
- [3] Recuperado el 20 de 12 de 2007, de <http://aps.sld.cu/bvs/E/sobreaps.html>
- [4] Turruelles Tejeda Yosvani, González Díaz Maikel. (2007). Registro de Ubicación Geográfica y Registro de Localidades para el Sistema de Información para la Salud. La Habana.
- [5] Felipe, d. J. (08 de 01 de 2008). Cuba: Salud de privilegios en el Siglo XXI. Recuperado el 04 de 02 de 2008, de [www.cubasocialista.cu: http://www.cubasocialista.cu/texto/0008110salud.html](http://www.cubasocialista.cu/texto/0008110salud.html)
- [6] Marin Sanchez, J., & Acosta Montejó, C. (2007). *Registro de la Clasificación Internacional de Enfermedades y Problemas relacionados con la Salud (RCIE) del Sistema de Información para la Salud*. Ciudad de la Habana: Univesidad de las Ciencias Informática(UCI).
- [7] Ídem Referencia 5.
- [8] Delgado Ramos, A., & Vidal Ledo, M. (2006). Recuperado el 18 de 12 de 2007, de http://bvs.sld.cu/revistas/spu/vol32_3_06/spu15306.htm
- [9] Ídem Referencia 8
- [10] E. Marín Díaz, D. M. (2004). *Revista Habanera de Ciencias Médicas*. Recuperado el 20 de 01 de 2008, de http://www.ucmh.sld.cu/rhab/editorial_rev10.htm
- [11] Delgado Ramos, D. A., Cabrera Hernández, I. M., & Juncal, D. V. (2006). *Registro Informatizado de Salud (RIS)*. Ciudad de la Habana: Ministerio de Salud Pública.
- [12] Ídem Referencia 4.
- [13] Peris, M. (mayo-agosto de 1997). *Trabajos de Diploma*. Recuperado el 20 de enero de 2008, de Informatica Médica: http://bvs.sld.cu/revistas/aci/vol5_2_97/aci06297.htm
- [14] *JagarSoft*. (abril de 2008). Recuperado el enero de 2008, de JagarSoft: <http://www.jagarsoft.com/>

REFERENCIAS BIBLIOGRÁFICAS

[15] *Dasi Informática*. (2007). Recuperado el diciembre de 2007, de Dasi Informática:
<http://www.dasi.es/content/view/2/4/lang.es/>

[16] Alejandro Soto, M. (s.f.). Recuperado el 07 de 02 de 2008, de <http://usuarios.lycos.es/janjo/janjo1.html>

[17] Orallo, E. H. (2008). *El Lenguaje Unificado de Modelado(UML)*. Recuperado el enero de 2008, de <http://www.disca.upv.es/enheror/pdf/ActaUML.PDF>

[18] Sanchez, I. M. (15 de abril de 2008). *Informatizate*. Recuperado el 15 de abril de 2008, de Informatizate: http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_software_07062004.html

[19] Tecnología orientada a procesos de negocio. (s.f.). Recuperado el diciembre de 2007, de Tecnología orientada a procesos de negocio: <http://www.proactiva-calidad.com/java/patrones/mvc.html>

[20] Java en Castellano. (2007). Recuperado en diciembre de 2007, de Java en Castellano:
http://www.programacion.net/java/articulo/joa_patrones4/#joa_patrones3_fachada

[21] Patrón Proxy. (diciembre de 2007). Recuperado en enero de 2008, de http://mendozajug.com.ar/portal/components/com_mambowiki/index.php?title=Patr%C3%B3n_Proxy

Bibliografía

1. ALBERTO, M. *Java en Castellano*, 2007. [Disponible en: http://www.programacion.net/java/articulo/joa_patrones4/#joa_patrones3_fachada
2. ALEJANDRO SOTO, M. *Protocolos TCP/IP*, 2008.
3. ARIEL DELGADO RAMOS, MARÍA VIDAL LEDO2. *Informática en la salud pública cubana Rev. Cubana Salud Pública* 2006.
4. *Atención Primaria de Salud*. 2006. [Disponible en: <http://aps.sld.cu/bvs/E/sobreaps.html>
5. COLECTIVO, A. *Arquitectura de Software para los componentes a emplear por el Sistema de Información para la Salud*. Ciudad de la Habana, SOFTEL, 2006.
6. CUBASALUD. *Salud Cubana*, 2006. [Disponible en: <http://cubasalud.blogia.com>
7. DASI INFORMÁTICA, S. L. *DasiClinic, Software Médico para la Gestión de su Clínica*, 2007. [Disponible en: <http://www.dasi.es/content/view/2/4/lang.es/>
8. DELGADO RAMOS ARIEL, ING. CABRERA HERNÁNDEZ MIRNA, & JUNCAL VIRGINIA. *Registro Informatizado de Salud (RIS)*. . Ciudad de la Habana, Ministerio de Salud Pública, 2005.
9. DR. MARÍN, D. M. *CONSIDERACIONES SOBRE EL PROYECTO DE INFORMATIZACION DE LA ATENCION PRIMARIA DE SALUD*
10. *Revista Habanera de Ciencias Médicas*, 2004.
11. GABRIEL CUERVO LUIS, AMÉRICA VALDÉS, & CLARK MARÍA LUISA *El registro internacional de ensayos clínicos Revista Panamericana de Salud Pública*, 2006.
12. GEORGE, E. *La atención primaria en Cuba: el equipo del médico de la familia y el policlínico. Revista Cubana de Salud Pública*, 1996.
13. HERNÁNDEZ ORALLO, E. *El Lenguaje Unificado de Modelado 2006*, 2006.
14. JAGARSOFT. *Medical Control 2*, 2008. [Disponible en: <http://www.jagarsoft.com/>
15. MARIN SANCHEZ, J., & ACOSTA MONTEJO. *Registro de la Clasificación Internacional de Enfermedades y Problemas relacionados con la Salud (RCIE) del Sistema de Información para la Salud*. . La Habana, Cuba, Universidad de las Ciencias Informáticas (UCI), 2007. p.
16. MENDOZA SANCHEZ, M. A. *Metodologías De Desarrollo De Software*, 2004a.
17. --- *Metodologías De Desarrollo De Software*, 2004b.
18. *Patrón Proxy*, 2007.

19. PÉREZ CRUZ, F. D. J. *Cuba: Salud de privilegios en el Siglo XXI* 2008. [Disponible en [Disponible en: <http://www.jagarsoft.com/>]
20. PERIS, C. M. Literatura gris. Trabajos de Diploma *Informática Médica*, 1997.
21. PRESSMAN. *Ingeniería de Software, un enfoque práctico*. La Habana, Editorial Félix Varela., 2004a. 2.
22. *Ingeniería de Software, un enfoque práctico, parte*. La Habana, Editorial Felix Varela, 2004b. 1.
23. RAMIRO, L. B. Tecnología orientada a procesos de negocio *Patrón "Modelo-Vista-Controlador"*, 2007.
24. TURRUEYES TEJEDA YOSVANIS, GONZÁLES DIAZ MAIKEL. *Registro de Ubicación Geográfica y Registro de Localidades para el Sistema de Información para la Salud*. La Habana, Cuba, Universidad de Ciencias Informáticas, 2007. p.

Anexo 1.2. Medical Control 2. Reportes

Impreso el Jueves, 06 de Octubre de 2005, 03:07 PM

Detalles de Ingreso

AARON VALDEZ GONZALEZ

El ingreso se realizó el día Jue 06/Oct/05 - Dr. José A. Martínez

Motivo de Ingreso

Consulta de control.

Antecedentes Perinatales

Producto 1 gesta, 31 años... control prenatal, llanto expont.

Alta con la madre, seno materno 3er mes...

Fórmula E-Nido, 4to mes..

Antecedentes Patológicos

Hospitalización: Gastroenteritis 1 6/12

Cirugías: No

Medicamentos: Negadas

Insectos: sancudos.

Casa-ambiente: hongos, jardín..

No fuman, hay perros adentro, y un gato en jardín.

Familia equitación.

Anexo 2. Dasi Clinic

FÁCIL

FIABLE

FLEXIBLE

TECNOLOGIES

DasiClinic

Dasi Informática S.L.

Contingut

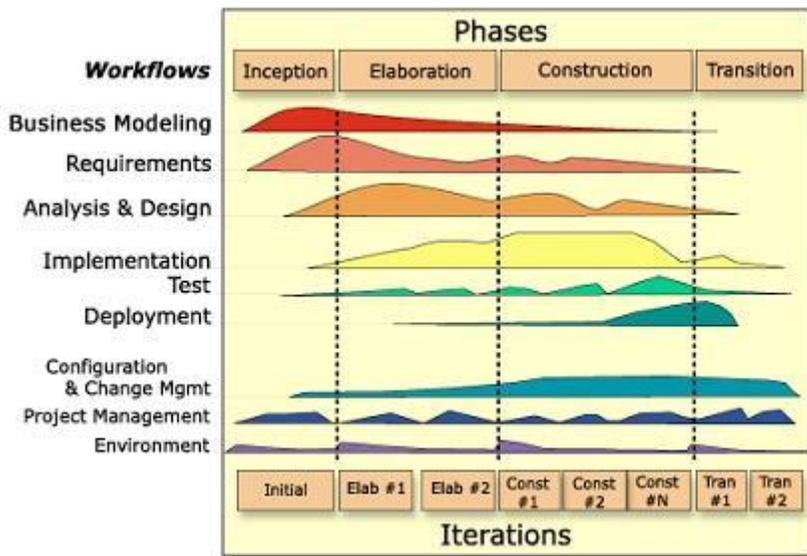
Veure vídeos

Demo

Especialitats

Preus

Anexo 3. Fases e Iteraciones de la Metodología RUP



Glosario de Términos

APS: Atención Primaria de Salud: Nivel asistencial que constituye la puerta de entrada del paciente al Sistema Nacional de Salud, donde debe darse solución alrededor del 90% de los problemas que afectan a la población. En este nivel se realizan acciones educativas, curativas y de Rehabilitación.

Aplicación Web: Es una aplicación informática que los usuarios utilizan accediendo a un servidor Web a través de un navegador o browser. Estas son muy populares debido a la habilidad para actualizar y mantener la información manipulada sin distribuir e instalar el software en miles de potenciales clientes.

Arquitectura: Conjunto de decisiones significativas acerca de la organización de un sistema de software, la selección de los elementos estructurales a partir de los cuales se componen el sistema. La misma se interesa no sólo por la estructura y el comportamiento, sino también por las restricciones y compromisos de uso, funcionalidad, funcionamiento, flexibilidad al cambio, reutilización, comprensión, economía y tecnología, así como por aspectos estéticos.

Base de Datos: Es un conjunto de datos que pertenecen al mismo contexto almacenados sistemáticamente para su posterior uso.

Caso de Uso: Descripción de un conjunto de secuencias de acciones, incluyendo variaciones, que un sistema lleva a cabo y que conduce a un resultado observable de interés para un actor determinado.

Componente: Parte física y reemplazable de un sistema que se ajusta a, y proporciona la realización de, un conjunto de interfaces.

Dependencia: Relación semántica entre dos elementos, en la cual un cambio en uno puede afectar al otro.

Diagrama: Presentación gráfica de un conjunto de elementos y sus relaciones.

Dominio: Área de conocimiento o actividad caracterizada por un conjunto de conceptos y terminología comprendidos por los practicantes de ese dominio.

IDE: En inglés Integrated Development Environment (entorno de desarrollo integrado).

INFOMED: Red telemática del Sistema Nacional de Salud (SNS) de Cuba que funciona como una división del Centro Nacional de Información de Ciencias Médicas (CNICM), y parte de la existencia de una red nacional especializada de centros de información.

Informática: Es la disciplina que estudia el tratamiento automático de la información utilizando dispositivos electrónicos y sistemas computacionales.

Informatizar: Proceso de aplicar sistemas o equipos informáticos al tratamiento de la información.

Metalinguaje: Es un lenguaje usado para hacer referencia a otros lenguajes.

MINSAP: Ministerio de Salud Pública en Cuba.

Paquete: Mecanismo de propósito general para organizar elementos en grupos.

PDF (del inglés, Portable Document Format, Formato de Documento Portátil): Es un formato de almacenamiento de documentos, desarrollado por la empresa Adobe Systems.

Proyecto: Esfuerzo de desarrollo para llevar un sistema a lo largo de un ciclo de vida.

RPC (del inglés, Remote Procedure Call): Es un protocolo que permite a un programa de ordenador ejecutar código en otra máquina remota sin tener que preocuparse por las comunicaciones entre ambos.

SISalud: Sistema de Información para la Salud, plataforma que brinda servicios web al Sistema Nacional de Salud.

SOFTTEL (Empresa de Soluciones Informáticas): Entidad del Ministerio de Informática y las Comunicaciones (MIC).

Servicio: Unidad de software que encapsula alguna funcionalidad de negocio y proporciona estas a otros servicios a través de interfaces públicas bien definidas.

Servicio Web: Colección de protocolos y estándares que sirven para intercambiar datos entre aplicaciones.

SOAP (del inglés, Simple Object Access Protocol, Protocolo de Acceso Simple a Objetos): Estándar que define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos XML. SOAP es uno de los protocolos utilizados en los Servicios Web.

Software: Conjunto de programas y procedimientos necesarios para hacer posible la realización de una tarea específica, en contraposición a los componentes físicos del sistema.

Software Libre: Es el software que, una vez obtenido, puede ser usado, copiado, estudiado, modificado y redistribuido libremente.

Subsistema: Agrupación de elementos, de los que algunos constituyen una especificación del comportamiento ofrecido por los elementos contenidos.