



**UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS**  
**FACULTAD 7**

Trabajo de Diploma para Optar por el Título de  
Ingeniero en Ciencias Informáticas

**Título: Registro de Problemas de Salud de la Atención Primaria (RPSAP)  
del Sistema de Información para la Salud.**

**Autores:**

Johander León Garcés  
Yanies Maday Jiménez Pomares

**Tutores:**

Ing. Mirna Cabrera Hernández  
Prof. Auxiliar  
Dr. Denis Derivet Thaireaux  
Prof. Instructor

**Asesor:**

Dr. Julio Alberto Mora Salvador

**Ciudad de La Habana, Junio del 2007**  
**“Año 49 de la Revolución”**

# DECLARACIÓN DE AUTORÍA

---

## Declaración de Autoría

Declaramos que somos los únicos autores de este trabajo, en el cual hemos utilizado información y documentación que es propiedad de la empresa SOFTEL, lo cual está sujeto a un acuerdo de confidencialidad. Ponemos a disposición de la Universidad de Ciencias Informáticas (UCI) todo aquello que no comprometa dicho acuerdo.

Para que así conste firmamos la presente a los 22 días del mes de junio del año 2007.

---

Johander León Garcés  
Firma del Autor

---

Yanies Maday Jiménez Pomares  
Firma del Autor

---

Ing. Mirna Cabrera Hernández  
Firma del Tutor

---

Dr. Denis Derivet Thureaux  
Firma del Tutor

### Datos de Contacto

**Mirna Cabrera Hernández:** Graduada de Ing. Sistema Automatizado de Dirección Técnico Económico (SAD) en el año 1986 en el ISPJAE. Posee categoría docente de Profesor Auxiliar y cursa la maestría de Gestión de Proyectos Informáticos. Ha impartido la asignatura Gestión de Software en la Facultad 7 desde el curso 2005-2006. Ha presentado ponencias en eventos científicos nacionales e internacionales. Se desempeña como Líder del Proyecto APS en la Empresa Softel. Sus direcciones de correo electrónico son: [mirna@softel.cu](mailto:mirna@softel.cu) y [mirna@infomed.sld.cu](mailto:mirna@infomed.sld.cu).

**Denis Derivet Thaireaux:** Graduado de Medicina en el año 1994. Especialista 1er Grado en Medicina General Integral en 1999. Experto Funcional del MINSAP. Profesor Instructor. Imparte asignaturas del Perfil de Salud en la Facultad 7. Cursa la maestría de Gestión de Proyectos Informáticos y Bioinformática. Posee 12 años de experiencia. Ha presentado ponencias en eventos nacionales e internacionales. Sus direcciones de correo electrónico son: [denis@softel.cu](mailto:denis@softel.cu) y [derivet@infomed.sld.cu](mailto:derivet@infomed.sld.cu).

### Agradecimientos

*Tratar de nombrar a todas y cada una de las personas significaría ponernos al riesgo de omitir a alguien importante -todos lo son-, por eso expresamos aquí nuestro reconocimiento y gratitud a todos.*

*Gracias a nuestros compañeros de estudio y del proyecto APS.*

*Gracias a los especialistas de la Empresa SOFTEL.*

*Gracias a los profesores de la Universidad de las Ciencias Informáticas.*

*Finalmente, reciba el más sincero agradecimiento nuestra Revolución y nuestro Comandante en Jefe Fidel Castro Ruz, que han hecho realidad nuestros sueños.*

## Dedicatoria

*Dedicamos este trabajo a:  
Nuestras familias.  
Nuestros compañeros del Proyecto APS.  
Todas las personas que a lo largo de la  
carrera nos ayudaron con su apoyo  
incondicional, a estar más cerca de  
nuestras metas y objetivos profesionales.  
A Todos, Gracias...*

### Resumen

El presente trabajo tiene como objetivo, desarrollar una aplicación web que gestione de forma eficiente la información relacionada con los problemas de salud en el nivel de Atención Primaria, garantizando el proceso de dispensarización y diagnóstico, así como la estandarización y homogeneidad de la información en este nivel. Estos procesos presuponen diferenciar los problemas de salud de los pacientes en: riesgos, enfermedades y discapacidades.

Se realizó un estudio de los antecedentes de los sistemas automatizados que gestionan la información relacionada con los problemas de salud existentes, a nivel nacional e internacional. Se exponen las tecnologías y herramientas utilizadas como: Servidor Web Apache, Servidor de Datos MySQL, PHP, XSL, XML, XHTML, JavaScript, entre otras, que han permitido adoptar una arquitectura sólida y consistente; respondiendo a los objetivos trazados por el MINSAP. Se describen los estilos arquitectónicos usados: orientado a servicios, basada en componentes y modelo tres capas.

El sistema obtenido, facilita el manejo adecuado de la codificación de los problemas de salud; así como, la relación de códigos de problemas de salud de la atención primaria, con los códigos de estándares internacionales, permitiendo el entendimiento común entre ambos. Permite aumentar la calidad, oportunidad y consistencia de la información; además incrementa la efectividad y eficiencia de los procesos relacionados con la salud, incidiendo en el incremento continuo y sostenido de la calidad en la atención médica.

**Palabras Claves:** gestión, problemas de salud, dispensarización, diagnóstico, aplicación web, estandarización y homogeneidad.

## Tabla de Contenidos

|   |    |
|---|----|
| <b>Introducción</b> .....   | 1  |
| <b>Capítulo 1. Fundamentación Teórica</b> .....                               | 8  |
| Introducción.....   | 8  |
| 1.1 Marco Conceptual.....   | 8  |
| 1.2 Proceso de la Dispensarización de los Pacientes en Cuba.....              | 10 |
| 1.2.1 Grupos de Dispensarización.....   | 11 |
| 1.3 Proceso de Diagnóstico de los Pacientes en Cuba.....                      | 12 |
| 1.4 Estado del Arte de los Sistemas Informáticos en la Atención Primaria..... | 13 |
| 1.5 Aspectos Generales del Sistema Nacional de Salud.....                     | 20 |
| 1.5 Problema a Resolver y Situación Problémica.....                           | 23 |
| 1.6 Objeto de Automatización e Información Manipulada.....                    | 27 |
| 1.7 Tendencias y Tecnologías.....   | 28 |
| 1.7.1 Estilos Arquitectónicos.....  | 29 |
| 1.7.2 Metodologías.....   | 30 |
| 1.7.3 Lenguajes.....  | 30 |
| 1.7.4 Servidores.....   | 32 |
| 1.7.5 PLASER.....   | 33 |
| 1.7.6 XML - Web Services.....   | 34 |
| 1.7.7 Herramientas.....   | 34 |
| Conclusiones.....   | 36 |
| <b>Capítulo 2. Características del Sistema</b> .....                          | 37 |
| Introducción.....   | 37 |
| 2.1 Modelo de Dominio.....  | 37 |
| 2.1.1 Conceptos Fundamentales.....  | 38 |
| 2.1.2 Diagrama del Modelo de Dominio.....                                     | 39 |
| 2.2 Propuesta del Sistema.....  | 40 |
| 2.2.1 Especificación de Requerimientos de Software.....                       | 40 |
| 2.2.1.1 Requerimientos Funcionales.....                                       | 41 |
| 2.2.1.2 Requerimientos No Funcionales.....                                    | 44 |

# TABLA DE CONTENIDOS

---

|   |           |
|---|-----------|
| 2.2.2 Modelo de Casos de Uso del Sistema.....         | 51        |
| 2.2.2.1 Definición de Actores del Sistema. ....       | 52        |
| 2.2.2.2 Diagrama de Casos de Uso del Sistema. ....    | 53        |
| 2.2.2.3 Descripción Textual de los Casos de Uso. .... | 54        |
| Conclusiones.....                                     | 59        |
| <b>Capítulo 3. Diseño del Sistema</b> .....           | <b>60</b> |
| Introducción.....                                     | 60        |
| 3.1 Modelo de Diseño. ....                            | 60        |
| 3.1.1 Patrones de Diseño Utilizados.....              | 61        |
| 3.1.1.1 Modelo Vista Controlador.....                 | 61        |
| 3.1.1.2 Bajo Acoplamiento.....                        | 63        |
| 3.1.1.3 Alta Cohesión. ....                           | 64        |
| 3.1.1.4 Fachada.....                                  | 64        |
| 3.1.2 Definición de la Estructura del Diseño.....     | 65        |
| 3.1.3 Diagramas de Clases del Diseño.....             | 67        |
| 3.1.4 Descripción de las Clases y Atributos. ....     | 74        |
| 3.1.5 Diagrama de Clases Persistentes.....            | 81        |
| 3.5 Modelo de Datos. ....                             | 82        |
| 3.6 Descripción de tablas y atributos.....            | 84        |
| Conclusiones.....                                     | 89        |
| <b>Capítulo 4: Implementación</b> .....               | <b>90</b> |
| Introducción.....                                     | 90        |
| 4.1 Integración con Otros Sistemas.....               | 90        |
| 4.2 Modelo de Implementación.....                     | 91        |
| 4.2.1 Diagrama de Componentes. ....                   | 92        |
| 4.2.2 Diagrama de Despliegue.....                     | 93        |
| 4.3 Descripción de Métodos y Agentes. ....            | 95        |
| 4.4 Estándares Utilizados. ....                       | 98        |
| 4.4.1 Estándar de Diseño.....                         | 99        |
| 4.4.2 Estándar de Codificación. ....                  | 101       |
| 4.4.3 Estándar de Tratamiento de Excepciones.....     | 103       |

# TABLA DE CONTENIDOS

---

|  |            |
|--|------------|
| Conclusiones.....                      | 106        |
| <b>Conclusiones.....</b>               | <b>107</b> |
| <b>Recomendaciones .....</b>           | <b>108</b> |
| <b>Referencias Bibliográficas.....</b> | <b>109</b> |
| <b>Bibliografía .....</b>              | <b>113</b> |
| <b>Glosario de Términos .....</b>      | <b>114</b> |

## Introducción

La Atención Primaria de Salud fue concebida en 1978 durante la Conferencia Internacional, en la ciudad de Alma Ata, Kazajistán, que convocó a 134 países y 67 organizaciones internacionales. La conferencia otorgó reconocimiento internacional al concepto de Atención Primaria de Salud (APS), que se definió como: "la asistencia sanitaria esencial basada en métodos y tecnologías prácticas, científicamente fundados y socialmente aceptables, puesta al alcance de todos los individuos y familias de la comunidad mediante su plena participación y a un costo que la comunidad y el país puedan soportar, en todas y cada una de sus etapas de desarrollo con un espíritu de autorresponsabilidad y autodeterminación". [1]

Aunque se relaciona el surgimiento del concepto, a esta conferencia, hay que tener en cuenta que en algunos países de América ya existían conceptos y acciones relacionadas con la Atención Primaria, que condujeron a la promoción de importantes políticas sociales en toda la región.

Los principios establecidos en la mencionada conferencia, siguen siendo indispensables para tener una visión coherente de la salud mundial. Para lograr transformar esa visión, es necesario, reducir las disparidades en salud que actualmente existen entre las distintas regiones del mundo. Además es necesario distinguir, tanto las posibilidades, como los obstáculos, que han frenado los progresos en materia de salud; lo que significa trabajar con los países, especialmente con los más necesitados, afrontando las crisis sanitarias y construyendo a través de la experiencia y los logros del pasado, sistemas de salud sostenibles y equitativos. [2]

En la Cumbre del Milenio, realizada en enero de 2000, se establecieron los Objetivos de Desarrollo del Milenio, aprobados por todos los Estados Miembros de las Naciones Unidas. Aquí se definieron metas ambiciosas, donde la salud ocupa un lugar rector y de vital importancia para las futuras generaciones; tales como: reducir la pobreza al 50% para el 2015, eliminar la desnutrición aguda y crónica, reducir la mortalidad infantil y materna mediante el acceso universal a servicios básicos integrales de salud, así como contar con agua segura y disposición sanitaria de excretas, prevención y control de las enfermedades, con la activa participación de las comunidades organizadas. [3]

Unos años más tarde, la Organización Panamericana de la Salud (OPS) y la Organización Mundial de la Salud (OMS) reconocen a la Atención Primaria de Salud como la estrategia principal del sector salud para alcanzar dichos objetivos.

En las diferentes regiones del planeta, las formas de concebir y ejecutar los sistemas de la APS varían por sus características. Lo que refleja la existencia de importantes diferencias de un país a otro; pudiendo mencionar: la colaboración multisectorial reducida, la existencia de limitaciones financieras de los diferentes sectores de la salud y de los individuos que necesitan atención médica, las crisis y profundos cambios políticos, sociales y económicos. Además, de la tendencia general hacia la integración de los componentes de la atención de salud en un enfoque integral, el incremento de enfermedades crónicas, las dificultades con el acceso a medicamentos y los conflictos en el fortalecimiento y empleo de capacidades rectoras y distritales de planificación y gestión.

En los últimos decenios, a diferencia de las otras regiones, la mayor parte de los países de la América Latina y el Caribe, han trabajado por alcanzar la meta de salud para todos a través de la APS y sus sistemas de salud giran en sentido general sobre la misma base, siendo sus soluciones muy similares. Históricamente se conocen las experiencias de Costa Rica, Chile y principalmente de Cuba, donde, en los últimos 25 años, se han conseguido logros importantes en el mejoramiento de las condiciones de salud y de vida, gracias al desarrollo de la APS en forma complementaria con otros modelos de atención.

Cuba, a pesar de los recursos limitados que posee, ha logrado notables avances en materia de salud, potenciando y fortaleciendo los servicios primarios de atención que constituye el eje fundamental de las transformaciones, con un énfasis mayor en la promoción de la salud, la medicina familiar y los servicios de urgencia en la comunidad. Su objetivo principal es convertir los Policlínicos en centros de excelencia, cada vez más accesibles a la población, consolidando así el Sistema Municipal de Salud. El gobierno reglamenta firmemente el sistema de salud y está intentando reducir las inequidades, ofreciendo mejor acceso y protección para la atención de salud a su población, cumpliendo así con la visión del Comandante en Jefe, Fidel Castro Ruz cuando expreso: “la idea esencial es acercar los servicios primarios a los ciudadanos”. [4]

Durante los últimos 20 años un grupo de instituciones cubanas han desarrollado sistemas encaminados a lograr determinados niveles de informatización de la salud. Estas soluciones carecían de integración y de una definición generalizable, además de la no existencia en aquellos momentos de recursos tecnológicos necesarios para su ejecución en el Sistema Nacional de Salud. A partir de 1997 se concibe una primera estrategia de informatización como respuesta del sector de la salud a los lineamientos estratégicos para la informatización de la sociedad cubana, con la finalidad de coordinar esfuerzos para el desarrollo de este proceso en el SNS. [5]

En el marco de la informatización de la sociedad cubana, el sector de la salud se propone contar con una solución informática integral que dote al sistema de mayor grado de acceso a información unificada y confiable en tiempo real, que aporte la rapidez y fiabilidad necesaria para las modernas técnicas de administración y para la toma de decisiones en los diferentes niveles; además de permitir que la información llegue de forma rápida, actualizada y de manera eficiente a todos los niveles requeridos, posibilitando la integridad y el control de la información y sirviendo de referencia a otras áreas con similares problemas, para la solución de sus necesidades.

También se persigue el mejoramiento de la calidad de los servicios médicos a la población, tanto en Cuba como en otros países del área, facilitando la ejecución de muchos de los procesos que se realizan de forma manual y logrando almacenar la información de una forma más eficaz, ya que la misma se almacena en papel, lo que provoca el deterioro o la pérdida de información y que el trabajo sea mucho más complejo y engorroso.

Con la implementación a partir de 2003, de los programas de la Revolución, que incluye como prioridad la informatización de los servicios, se inicia un amplio proceso, orientado en primer lugar a la superación y desarrollo profesional, que a su vez se iría extendiendo con la automatización de los servicios médicos, la investigación, la información científico-técnica y el apoyo en la toma de decisiones, comenzando en los policlínicos y expandiéndose al resto del sistema. [6]

Para ello se convocan a un grupo de instituciones propias del sector, del Ministerio de Informática y Comunicaciones (MIC) y de otros organismos de la administración central del estado, para definir de conjunto la estrategia a desarrollar.

Entre estas instituciones se destaca la Empresa Softel, dedicada al desarrollo de aplicaciones informáticas para la salud, a la cual se le asigna la misión de informatizar los procesos del Sistema Nacional de Salud (SNS). A raíz de esta tarea se forman varios grupos de desarrollo para informatizar los tres niveles de atención definidos por el Ministerio de Salud Pública (MINSAP) y se definen las estrategias a seguir; asumiendo el diseño y puesta en marcha de manera gradual de un Sistema de Gestión de la Información y el Conocimiento de nuevas dimensiones: el Sistema de Información para la Salud (SISalud), que incluye los sistemas informatizados para el nivel de atención primaria, secundaria y especializada y los nomencladores que a nivel nacional deben definirse para garantizar el funcionamiento de los mismos de forma homogénea.

La Empresa Softel aportando profesionales con experiencia se vincula a los estudiantes y profesores de la Universidad de las Ciencias Informáticas (UCI), para aunar esfuerzos y de esta forma darle una correcta solución a las prioridades de informatización definidas. De esta integración surge, en la primera etapa de trabajo el Proyecto APS, con la misión de informatizar la Atención Primaria de Salud.

En el proceso de desarrollo de las aplicaciones informáticas asumido por el Proyecto APS, se define, la dificultad de la gestión de la información relacionada con los problemas de salud, para la clasificación en grupos homogéneos de los pacientes, que facilite el proceso de dispensarización y diagnóstico, procesos que presuponen diferenciar los problemas de un paciente en riesgos, enfermedades y discapacidades. Esta estructura, no se presenta en los sistemas elaborados mundialmente para la gestión de los problemas de salud en este nivel, por ser un elemento particular del Sistema Nacional de Salud Cubano.

Por otra parte los sistemas realizados anteriormente a nivel nacional e internacional, presentan complejidad en la definición de los nombres de los problemas de salud, que en ocasiones son registrados con muchas especificaciones y detalles que no son importantes en el entorno en que se manejan y que en diversas situaciones no se corresponden con la idiosincrasia del país, debido a que deben regirse por estándares internacionales que ofrece la OMS, como es el caso de la Clasificación Estadística Internacional de Enfermedades y Problemas Relacionados con la Salud en la Décima Revisión (CIE-10), que brinda una clasificación muy detallada y de difícil manejo, trayendo consigo que el trabajo del médico sea más complejo. Al mismo tiempo, al realizar los diagnósticos de salud sin regirse por estos estándares

impiden el establecimiento de un lenguaje común entre las organizaciones mundiales. En resumen las clasificaciones existentes no permiten utilizar de forma eficiente, fácil y sencilla los problemas de salud.

La situación descrita con anterioridad nos permite definir el **Problema a Resolver** de la siguiente forma: el sistema actual para la gestión de la información relacionada con los problemas de salud no garantiza de forma eficiente los procesos de dispensarización y de diagnóstico en la atención primaria, afectando la estandarización y homogeneidad de la información en este nivel de atención.

Como resultado de la problemática planteada, se espera obtener un producto que garantice la gestión de los problemas de salud de manera eficiente para la dispensarización de un paciente, así como para otros procesos propios de la atención primaria relacionados con el diagnóstico; permitiendo el manejo adecuado de la codificación y el renombramiento de los mismos acorde al país específico. Facilitando además la integración con otros módulos de la atención primaria, la reestructuración de los problemas de salud en riesgos, enfermedades y discapacidades y la forma de relacionar códigos de problemas de salud de la atención primaria con códigos de la CIE-10, para el entendimiento común y mundial.

El paciente será el principal beneficiado con este producto, ya que al garantizar la calidad, oportunidad y consistencia de la información, se incrementará la efectividad y eficiencia de los procesos relacionados con la salud, trayendo como consecuencia un incremento continuo y sostenido de la calidad en la atención médica.

Para el desarrollo de la investigación se plantea como **Objeto de Estudio**: el proceso de gestión de la información de la Atención Primaria en el Sistema Nacional de Salud.

Su **Campo de Acción** es: el proceso de gestión de la información relacionada con los problemas de salud de la Atención Primaria.

Por lo que el **Objetivo General de la Investigación** es desarrollar una aplicación web que gestione la información relacionada con los problemas de salud en el nivel de Atención Primaria y que garantice el proceso de dispensarización y diagnóstico en este nivel.

Se plantea como **Idea a Defender**: el desarrollo de una aplicación web que automatice la gestión de la información relacionada con los problemas de salud en el nivel de Atención Primaria, garantizará de forma eficiente el proceso de dispensarización y diagnóstico de los pacientes en este nivel, facilitando el procesamiento estadístico, estándar y homogéneo.

Para darle cumplimiento al objetivo planteado se proponen las siguientes **Tareas de la Investigación**:

1. Realizar entrevistas a los clientes para identificar los requerimientos funcionales y no funcionales.
2. Identificar la interacción con otros componentes del Sistema de Información para la Salud.
3. Describir los procesos asociados a la gestión de los problemas de salud.
4. Asimilar la Arquitectura definida por el MINSAP.
5. Modelar los artefactos de los flujos de trabajo: Modelamiento del Negocio, Gestión de Requerimientos, Diseño e Implementación.
6. Implementar los componentes del módulo y su interacción con otros sistemas.

El documento está estructurado por cuatro capítulos:

**CAPÍTULO 1. Fundamentación Teórica:** Contiene los aspectos esenciales para entender el entorno del problema a resolver. Se describen los conceptos fundamentales asociados al dominio del problema, el proceso de dispensarización y diagnóstico de los pacientes en Cuba, el estado del arte de los sistemas informáticos en la atención primaria que gestionan la información relacionada con los problemas de salud, los aspectos generales del Sistema Nacional de Salud, el problema a resolver, la situación problemática, el objeto de automatización, la información manipulada, las tendencias y las tecnologías.

**CAPÍTULO 2. Características del Sistema:** Presenta los argumentos principales de los procesos vinculados al problema a resolver, como son el Modelamiento del Negocio que contiene el Modelo del

Dominio y los conceptos fundamentales asociados a este y la Captura de Requerimientos, donde existen dos actividades principales: la especificación de requerimientos funcionales y no funcionales y el modelamiento del sistema.

**CAPÍTULO 3. Diseño del Sistema:** Muestra los elementos básicos del diseño del sistema, mediante la justificación del uso de patrones y los diagramas de clases del diseño, el diagrama de clases persistentes, el modelo de datos, así como la explicación de sus tablas y sus atributos y las descripciones de las clases más significativas del diseño.

**CAPÍTULO 4. Implementación:** Describe los componentes fundamentales de todo el proceso de implementación, argumentado la justificación de la integración con otros módulos, los métodos, agentes, estándares de diseño, codificación y excepciones, así como la representación general del diagrama de componente y de despliegue.

Además este documento contiene las conclusiones de la investigación, las recomendaciones a tener en cuenta para la continuidad del proyecto, las referencias bibliográficas y la bibliografía, así como el glosario de términos que proporcionará un mejor entendimiento del estudio realizado.

## Capítulo 1. Fundamentación Teórica

### Introducción.

La necesidad de disponer de una clasificación exclusiva para los problemas de salud de la atención primaria, desde hace algunas décadas es un problema que influye y determina en la sociedad contemporánea a nivel mundial. En varios países se realizan investigaciones y se trazan metas y líneas con el objetivo de resolver los problemas más notables en este nivel de atención.

La creación de redes de computadoras personales y otras tecnologías de la información y las comunicaciones entre instituciones médicas, así como su informatización, constituyen una solución a múltiples inconvenientes vinculados al sector de la salud. En muchas ocasiones, se percibe como una dificultad simple que se resuelve fácilmente, sin embargo, en la realidad, es un problema complejo que requiere aún, de proyectos de investigación científica multidisciplinaria, de desarrollo tecnológico y de mucho intercambio académico científico-tecnológico.

En el capítulo 1 se ofrecen los conceptos básicos que se van a tratar en todo el documento; una breve descripción del proceso de dispensarización y diagnóstico de los pacientes, así como la situación actual del proceso de la informatización en el nivel de atención primaria en Cuba y en el mundo, respecto a la gestión de la información de los problemas de salud. Se sintetiza el origen de los sistemas o aplicaciones que existen en este nivel de atención como solución a diferentes problemas y se describen los aspectos generales del Sistema Nacional de Salud. Se expone la situación problemática, el problema a resolver, el objeto de automatización y la documentación de la información que se maneja en todo el proceso de desarrollo. Por último, se hace una breve descripción de la arquitectura y se incluyen las metodologías, las tecnologías y las herramientas.

### 1.1 Marco Conceptual.

**Informatización del Sistema Nacional de Salud (SNS):** “Está dada por el conjunto de métodos, técnicas, procedimientos y actividades gerenciales dirigidas al manejo de la información en salud, la cual

# CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

comprende el estado de salud de la población, el conocimiento de las ciencias de la salud y la información en general para la toma de decisiones, clínico-epidemiológicas, operativas y estratégicas“. [7]

**Información:** “Conjunto organizado de datos, que constituyen un mensaje sobre un determinado ente o fenómeno”. [8]

**Gestión de Información:** “Actividades relacionadas con la obtención de la información adecuada, a un precio conveniente, en el tiempo y lugar apropiado, para tomar la decisión oportuna”. [9]

**Almacenamiento de la Información:** “Forma de guardar la información, de manera que la misma esté protegida y disponible en todo momento“. [10]

**Problemas de Salud:** “Cualquier queja, observación o hecho que el paciente y/o el médico perciben como una desviación de la normalidad, que ha afectado, afecta, o puede afectar la capacidad funcional del paciente. El problema de salud no se limita a una enfermedad en particular, no exige un nivel de daño o deterioro de la salud, sino, al menos, una preocupación del médico o del paciente de que pueda afectarse la salud por ese problema”. [11]

**Codificación de los Problemas de Salud:** “Proceso de convertir los términos de diagnósticos y de otros problemas de salud, de palabras a códigos alfanuméricos que permitan su fácil almacenamiento y recuperación”. [12]

**Aplicación Web:** “Un sistema informático que los usuarios utilizan accediendo a un servidor web a través de Internet o de una Intranet”. [13]

**Factor de Riesgo:** “Toda característica o circunstancia determinada de una persona o un grupo de personas que según los conocimientos que se poseen, asocia a los interesados a un riesgo anormal de sufrir un proceso patológico o de verse afectado desfavorablemente por tal proceso”. [14]

# CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

## 1.2 Proceso de la Dispensarización de los Pacientes en Cuba.

Hoy en día, a nivel mundial en la APS se llevan a cabo varios procesos indispensables para el control efectivo de la salud de los pacientes y existen tendencias fuertes de obtener clasificaciones de pacientes en grupos homogéneos (proceso de dispensarización de pacientes). En Cuba, el desarrollo de la Medicina Familiar, ha permitido mejorar con un enfoque más integral la atención médica de la población en el primer nivel de atención. Teniendo como acción importante la Dispensarización, que se ha perfeccionado como instrumento de abordaje en la gestión sanitaria y constituye un elemento fundamental en el trabajo del médico de familia.

La Dispensarización se define como: “un proceso organizado, continuo y dinámico de evaluación e intervención planificada e integral, con un enfoque clínico, epidemiológico y social, del estado de salud de las individuos y familias. Es un proceso coordinado y liderado por el Equipo Básico de Salud (EBS)”. [15]

La dispensarización permite al EBS y a otros profesionales de la salud conocer y accionar sobre los principales problemas de salud de los pacientes, tales como riesgos, enfermedades y discapacidades o deficiencias que afectan su calidad de vida y no permiten alcanzar indicadores de salud positiva.

Entre los propósitos de la dispensarización de un paciente se incluyen: “desarrollar un proceso de mejora continua del estado de salud de los individuos y familias, elevar la satisfacción de la población con los servicios de salud que brinda el sistema, determinar el estado de salud de individuos y familias, promover estilos de vida saludables en los individuos y familias, identificar e intervenir riesgos, enfermedades y otros daños a la salud individual y familiar, facilitar la intervención multidisciplinaria en los problemas de salud individual y familiar, aportar la información necesaria sobre la salud individual y familiar para el desarrollo del análisis de la situación de salud y mejorar la eficiencia en el trabajo del sistema de medicina familiar”. [16]

La Atención Dispensarizada se torna importante en la atención al paciente. Consiste en el registro, agrupamiento y control de las personas y familias. Esta acción facilita la observación y atención médica permanente y dinámica de individuos. Provee de grupos de personas con el objetivo de controlar riesgos, daños a la salud individual y colectiva y promover estilos de vida saludables.

# CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

En el país se utiliza para la clasificación de los pacientes en Grupos Dispensariales el siguiente algoritmo:

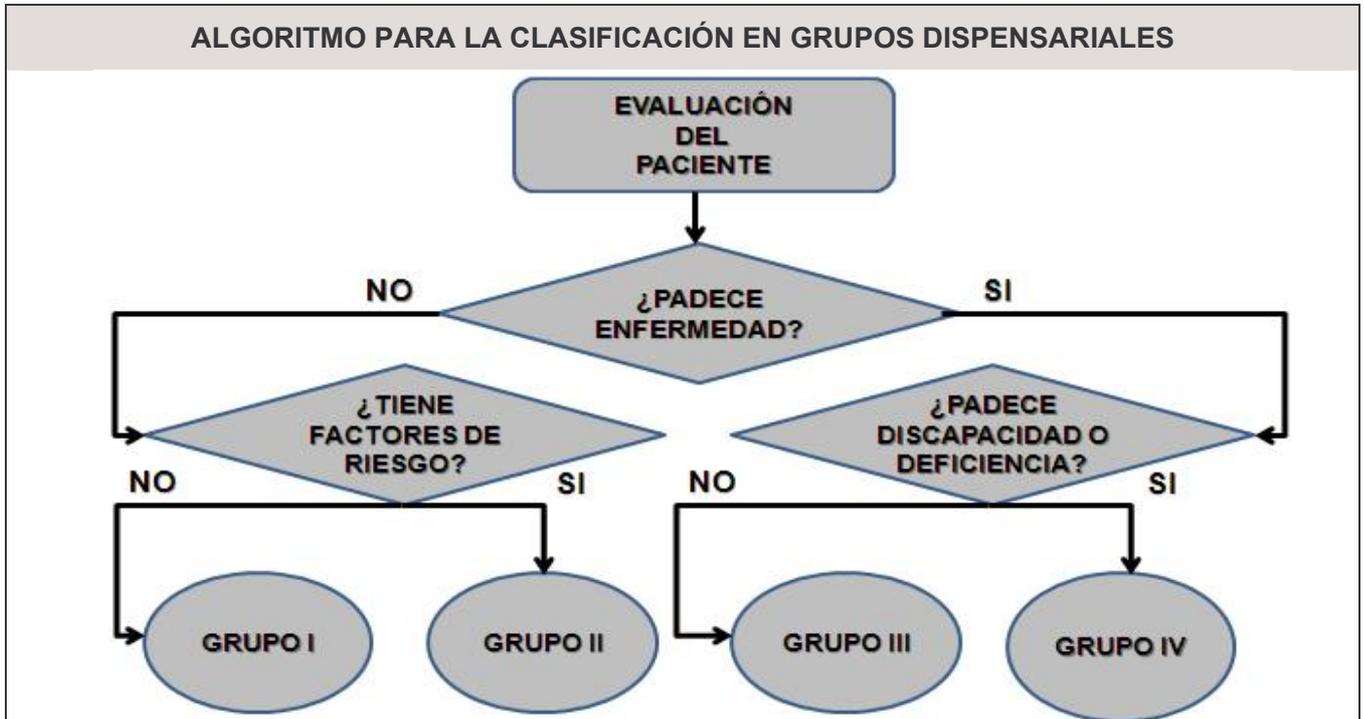


Figura 1.1 Algoritmo para la clasificación en grupos dispensariales

Este proceso de dispensarización surgido desde la práctica médica en la atención primaria, se plantea la búsqueda de mejores respuestas y soluciones para facilitar el trabajo de los profesionales de la salud y con esto mejorar la atención a los pacientes de forma individual y colectiva.

## 1.2.1 Grupos de Dispensarización.

Los individuos se clasifican en diferentes Grupos de Dispensarización, a los que se les brinda una atención activa y controlada periódicamente, cuya frecuencia depende de la edad y el grupo en el que se clasificó. Se definen estrategias y atención especializada para los grupos de acuerdo con sus características (clasificación). Los Grupos son:

**Grupo I.** Personas supuestamente sanas: "en este grupo se incluyen todas aquellas personas que no padezcan ninguna enfermedad ni estén en riesgo de padecerlas". [17]

# CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

**Grupo II.** Personas en riesgo o con factores de riesgo: “en este grupo se incluyen todas aquellas personas que no padezcan ninguna enfermedad pero tienen factores de riesgo para padecer enfermedades o accidentes que pueden repercutir en su salud”. [18]

**Grupo III.** Personas enfermas: “en este grupo se incluyen todas aquellas personas que padezcan alguna enfermedad, pero que no presentan discapacidad o deficiencia”. [19]

**Grupo IV.** Personas con deficiencias o con discapacidades: “en este grupo se incluyen todas aquellas personas que tienen discapacidad o deficiencia de enfermedades o accidentes”. [20]

## 1.3 Proceso de Diagnóstico de los Pacientes en Cuba.

La APS es el primer nivel de atención y contacto con la población que brinda la estructura del SNS, sin tener en cuenta nivel socioeconómico, raza, edad, sexo, estado de salud o enfermedad. Su principal recurso para detectar las enfermedades de los pacientes es a través del proceso de diagnóstico, que es de vital importancia para el equilibrio de la salud de los individuos.

En la medicina, el diagnóstico es el procedimiento por el cual se identifica una enfermedad, entidad nosológica, síndrome, o cualquier condición de salud-enfermedad. En términos de la práctica médica, el diagnóstico es un juicio clínico sobre el estado psicofísico de una persona; representa una manifestación en respuesta a una demanda para determinar tal estado. [21]

El diagnóstico clínico requiere tener en cuenta los dos aspectos de la lógica, es decir, el análisis y la síntesis, utilizando diversas herramientas como la anamnesis o interrogatorio, la exploración física y las indicaciones de exámenes complementarios. [22]

Existen dos tipos fundamentales de diagnóstico:

**Presuntivo:** Identificación de una enfermedad por sus signos o síntomas. Sin llegar a ser el diagnóstico definitivo. [23]

# CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

**Confirmado:** El diagnóstico que se confirma a través de algún proceder médico o por laboratorio. [24]

El diagnóstico es la base de la planeación de los servicios de salud, su objetivo principal es conocer la situación del proceso salud-enfermedad como fenómeno de una población en relación a: daños a la salud, servicios de salud, factores condicionados y riesgos. Permite el desarrollo de acciones planificadas para la protección a la salud y la prevención de brotes y epidemias. [25]

Los modernos procedimientos de diagnósticos permiten cada vez más un tratamiento temprano y efectivo de las enfermedades. Producen una amplia base de conocimientos que los gobiernos pueden utilizar para definir políticas, seleccionar prioridades y asignar recursos con racionalidad, a la vez, facilita niveles mayores de eficacia, eficiencia y equidad en la toma de decisiones y en la aplicación de intervenciones en los ámbitos individual y colectivo.

## 1.4 Estado del Arte de los Sistemas Informáticos en la Atención Primaria.

La gestión y el almacenamiento de la información en la medicina son dos de los principales procesos que tienen lugar hoy en día; los profesionales de la salud en diferentes países manifiestan preocupación por la imposibilidad de ejecutar correctamente estos procesos en diversas situaciones. Esta problemática ha estado vigente en la historia del pensamiento científico, ha sido objeto de estudio de disímiles especialistas y es considerada en la actualidad como una de las limitantes que imposibilita brindar servicios íntegros y eficaces a la sociedad.

Las organizaciones internacionales dedicadas a la salud están dando solución a estas dificultades creadas en distintas etapas; unos de los grandes pasos de avances para equilibrar estos problemas de la sociedad son las aplicaciones o sistemas informáticos.

Contar con un sistema informático que provea información adecuada y oportuna constituye el eje fundamental en la toma de decisiones y acciones a realizar para el cumplimiento de los objetivos y metas de una institución médica. El sistema informático debe contribuir a mejorar la calidad de atención, aumentar la productividad y disminuir costos. El manejo correcto de la información puede potenciar las

# CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

estrategias diagnósticas y terapéuticas, la administración del personal de la institución y los usuarios, la coherencia tecnológica y las inversiones de un centro médico.

A partir de 1950 se implantaron las primeras aplicaciones en los hospitales e institutos sobre máquinas pensantes; cálculos de radioterapia y bioquímicos desde 1952; diagnóstico diferencial desde 1954; cálculos muestrales, celulares, y cardio-respiratorios desde 1956 y 1957; estudios de aprendizaje y control de parasitismo desde 1958; análisis de electrocardiograma desde 1965; historia clínica computarizada desde 1966; instrucción médica, interrogatorio a paciente por computadora desde 1968; y consulta a expertos desde 1971. [26]

Los sistemas de información computarizados electromecánicamente en la APS se desarrollaron desde 1930-1940, y electrónicamente desde 1974. La informática e informatización de la APS aparecen a partir de 1985, como apoyo al médico en el diagnóstico, la terapia clínica y al programa de salud en la comunidad. [27]

Las primeras aplicaciones de la informática en la APS fueron los sistemas de recolección continua, la comunicación y la recuperación computarizada de los datos médicos de los pacientes y familiares en la APS. Esto llevó a estudiar la clasificación de enfermedades que había surgido en los hospitales para ajustarla a las necesidades del trabajo médico más ambulatorio en la APS. También, hizo trabajar con un universo de trabajo por área de residencia, en vez de la casuística hospitalaria generalmente por enfermedades. Los sistemas continuos sugirieron hacer investigaciones eventuales. [28]

Un elemento esencial a tener en cuenta es que en la actualidad el patrón de morbilidad ha cambiado, pues disminuyen las enfermedades infecciosas y aumentan las crónicas y los accidentes de tráfico. En definitiva, son más importantes las consecuencias de las enfermedades que la enfermedad en sí misma. Se hace necesario modificar las clasificaciones existentes para hacer más funcional el manejo de la información.

Dado que los datos generados deben ser formidables para un análisis estadístico adecuado y completo, es necesario un proceso de codificación en el cual se definan y estandaricen cada una de las entidades a medir mediante un vocabulario controlado. La codificación o clasificación de diagnósticos médicos surge a

# CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

partir de la intención de tener una terminología estandarizada que permita transformar el significado de un diagnóstico a una representación independiente del lenguaje, de modo que permita la disponibilidad de la información para propósitos de compatibilidad e integración. Esta codificación de datos permite su agrupación y comparación estadística, ya sea entre diferentes escenarios o en un mismo escenario pero en diferentes momentos.

Unas de las clasificaciones más importantes es la Clasificación Estadística Internacional de Enfermedades y Problemas Relacionados con la Salud (CIE-10), que nació como una clasificación de causas de mortalidad. El propósito de la CIE-10 es permitir el registro sistemático, el análisis, la interpretación y la comparación de los datos de mortalidad y morbilidad recolectados en diferentes países o áreas y en diferentes épocas, recoge con gran detalle las enfermedades existentes y las nuevas que surgen. La OMS diseñó su sistema de información estadística basada en esta clasificación, para promover la comparación internacional de la recolección, procesamiento, clasificación y presentación de estas estadísticas.

En la práctica, la CIE se ha convertido en una clasificación diagnóstica estándar internacional para todos los propósitos epidemiológicos generales y muchos otros de administración de salud. Actualmente se encuentra en su Décima Revisión, en la cual ha adquirido un nuevo nombre: Clasificación Estadística Internacional de Enfermedades y Problemas Relacionados con la Salud, cuenta con una estructura alfanumérica: "Asignación de una letra o grupo de letras a cada capítulo (antes llamados "Secciones")", lo que duplica la capacidad de la CIE-9. [29]

La CIE en su formato actual se hace insuficiente para resolver los problemas en el nivel de la APS, creando otras dificultades. En primer lugar, los países necesitan, renombrar los problemas de salud con nombres más comunes o relacionados con la idiosincrasia del país o lugar donde se utiliza la clasificación, por lo que la CIE se comporta de forma deficiente a la hora de utilizarla para este fin. Presenta una codificación con muchos detalles, que en diversas situaciones lo que hace es entorpecer y complicar el trabajo del médico.

Por otro lado en este nuevo período de cambios surge la necesidad de reestructurar los problemas de salud para su manejo más fácil y eficaz, teniendo en cuenta siempre la región y el lugar donde se quiere poner en práctica la clasificación. Las aplicaciones informáticas realizadas tanto a nivel nacional e

# CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

internacional con la CIE, no ofrecen una solución eficiente para resolver los problemas relacionados con la dispensarización y diagnóstico en la atención primaria.

Otra clasificación que surge para suplir el problema del cambio de morbilidad es la Clasificación de Deficiencias, Discapacidades y Minusvalías (CIDDM), creada por la OMS para codificar las consecuencias de las enfermedades. La misma fue aprobada en 1976 y su gran hallazgo fue identificar los tres niveles de las consecuencias de la enfermedad: impairment (deficiencia) como manifestación de la enfermedad; disability (discapacidad) como su objetivación; y handicap (minusvalía) como su socialización. [30]

En el 2002 después de concluido los trabajos de revisión de esta clasificación surge la nueva versión denominada "Clasificación Internacional del Funcionamiento, de la Discapacidad y de la Salud", conocida con las siglas CIF. [31] Ésta posee una estructura compleja que no cumple con las expectativas de Cuba, por lo que no resuelve los principales problemas en el nivel de Atención Primaria, al igual que la CIE-10. Actualmente no se conoce ninguna aplicación de esta clasificación.

A partir de 1972 el Comité de Clasificación de la Organización Mundial de Medicina Familiar (WONCA) establece los instrumentos necesarios para la investigación en medicina generalista. Unos años después crea la Clasificación Internacional en Atención Primaria (CIAP), su principal creación, basada en la Clasificación Internacional de Enfermedades (CIE). CIAP está disponible en más de veinte idiomas y se encuentra ya en su segunda edición (CIAP-2 o CIPSAP-2). Es el sistema más utilizado en España y su propósito es clasificar tanto los problemas de salud de las personas que se sienten enfermas, como de aquellos que considerándose sanos, buscan la evaluación y atención en el nivel primario. [32]

Varias organizaciones mundiales se han dado la tarea de implementar el CIPSAP-2, tal es el caso de la empresa SoftMédico, de Buenos Aires, Argentina, que ha diseñado varios sistemas basados en esta clasificación. De ellos se pueden mencionar el CIE-CIAP para quienes necesitan mantener su listado de diagnóstico, ya sea con la CIE-10 o la CIPSAP-2. Es una aplicación de libre distribución que requiere que el cliente posea Access XP para correr sin problemas, sin compatibilidad con otros gestores de Access anteriores. Además se encuentra el sistema CIE-CIAP FULL como versión más avanzada del CIE-CIAP. La misma presenta un mejor entorno de interfaz, haciendo más amigable la aplicación al usuario; también permite realizar búsquedas en diferentes revisiones de CIE (CIE-9 y CIE-10).

# CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

Otros sistemas que se han derivado de la Clasificación Internacional de la Atención Primaria es el ICPC-2© Master DataBase, el mismo ha sido financiado y auspiciado por la WONCA y está hecho en inglés. Presenta una característica muy notable que no había presentado ningún sistema hasta el momento, esta propiedad es la de relacionar los problemas de salud de la APS con la CIE, mejora que hace más potente el uso de esta herramienta en la APS.

Aunque la Clasificación Internacional en la atención primaria y las aplicaciones informáticas que implementan dicha clasificación, constituye un gran paso de avance para los países y las regiones que tienen incluida la atención primaria en su infraestructura de salud; en Cuba la realidad es distinta, por tener un sistema de salud con características especiales, donde se utiliza una clasificación dividida en riesgos, enfermedades y discapacidades para su buen desempeño en este nivel, que no está presente en los sistemas informáticos realizados con esta concepción a nivel internacional.

Existen otras alternativas para la codificación de los problemas de salud en la atención primaria. En Argentina se tomó en consideración las necesidades planteadas por el Ministerio de Salud de la Nación y algunas jurisdicciones, la Comisión Nacional de Clasificación de Enfermedades (CNCE) dependiente de la Dirección de Estadística e Información de Salud del Ministerio de Salud de la Nación y la Organización Panamericana de la Salud, realizó la Clasificación Estadística de Problemas de la Salud para la Atención Primaria (CEPS-AP), cuya primera edición data de agosto de 2001 y contó con el apoyo técnico y financiero de la OPS y OMS. [33]

La CEPS-AP se estructuró a partir de la Clasificación Estadística Internacional de Enfermedades y Problemas Relacionados con la Salud, 10ma. Revisión (CIE-10), editada en español por la Organización Panamericana de la Salud (OPS), de manera que sea compatible con ella. Al igual que la CIE-10, permite clasificar enfermedades y traumatismos con un diagnóstico formal y otros problemas y razones para entrar en contacto con los servicios de salud. Condensó las 2036 categorías de la CIE-10 en 268 categorías, tratando que resulte más breve y de manejo más simple; su objetivo es clasificar los problemas de salud atendidos en la atención primaria de una manera más breve y simple que la CIE-10. [34]

# CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

Con esta Clasificación se trata de cubrir la visión para Argentina, de los problemas de salud de primer contacto. No pretende ser aplicable en los niveles de atención de mayor complejidad. Consta de un único volumen que contiene la Lista Tabular de Categorías, el Índice Alfabético y orientaciones para el uso de la clasificación.

En la actualidad no se conocen aplicaciones o sistemas informáticos que implemente esta clasificación. A pesar de que esta codificación cumple sus objetivos en su marco de acción, la misma no sirve para ser utilizada en el proceso de dispensarización de un paciente y en otras actividades asociadas al diagnóstico en el nivel de atención primaria en Cuba, ya que no consta de una estructura dividida por riesgos, enfermedades y discapacidades, surgida en el entorno cubano y utilizada en la mayoría de los procesos que se llevan a cabo en la APS.

En los últimos años con el advenimiento de las investigaciones, se despierta, un creciente interés por contar con sistemas o aplicaciones de clasificación que permitan obtener grupos homogéneos en cuanto a la necesidad de recursos asistenciales. Cuba no se queda atrás en este sentido y desde hace varias décadas empezó a buscar soluciones y alternativas para resolver este problema, como consecuencia surgen varias clasificaciones para facilitar el trabajo del médico en el primer punto de contacto con el paciente.

La clasificación de riesgos surgió hace más de 20 años, se usa en Cuba después de haber sido analizada por un grupo de expertos y aprobada para la APS; parte de la organización de la Atención Primaria con el Programa del Médico y Enfermera de la Familia, como parte de una de las actividades fundamentales de este tipo de atención que es la dispensarización.

La clasificación de las enfermedades surge como punto de partida desde el mismo proceso de dispensarización y vigilancia de la morbilidad y mortalidad en Cuba, utiliza los términos médicos más comunes y son las más frecuentes en la población cubana.

El uso de la clasificación de las discapacidades se oficializó a raíz del estudio de las discapacidades que se llevó a cabo en Cuba en los años 2002-2003. Esta propuesta se efectuó por un grupo de expertos cubanos y fue la que el MINSAP aprobó para el estudio de las discapacidades.

# CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

Estas clasificaciones son las que se utilizan en Cuba para llevar a cabo la dispensarización y diagnóstico de un paciente en la APS. Fueron aprobadas por la Dirección Nacional de la Atención Primaria (DNAPS) mediante investigaciones y reuniones de varios especialistas.

El problema de la informatización de las actividades que cotidianamente se realizan de forma manual y en papel en los policlínicos comunitarios y Equipos Básicos de Salud (EBS), se comenzó a abordar desde mediados de los 1980's en los Policlínicos Docentes Lawton y Plaza. [35] La más completa solución para el tratamiento de las estadísticas del policlínico y los consultorios, con el fin de administrar los servicios de la APS, se implantó experimentalmente por parte del Centro para el Desarrollo Informático en la Salud Pública (CEDISAP) a partir de 1999, en una red del departamento de estadística del Policlínico Universitario Vedado. [36]

A partir del 2002 al 2003 se diseñaron nuevos proyectos de investigación y desarrollo para introducir los avances digitales mundiales en los servicios de APS, tomando en cuenta las experiencias exitosas de los países más avanzados en el uso de las Tecnologías de la Información y Comunicación (TICs) en la salud. [37]

En el 2002 en el Policlínico Comunitario Docente Capdevila se programó un Sistema de Dispensarización de los Pacientes en los Consultorios de Médicos de Familia (SIDAPS) con apoyo del CECAM, entre otros trabajos académicos realizados para la APS. Aunque este fue un intento digno de reconocer, no cumplió las expectativas para el cual se creó, por ser una aplicación monolítica, que carece de integración y generalización, por tal razón fracasaron las ideas de informatizar el proceso de dispensarización de un paciente. [38]

Se puede afirmar que desde 1986 al 2006 el interés por el control estadístico de la población, sus problemas de salud, y los servicios de los consultorios y policlínicos para uso gerencial de salud, constituyen los primeros 20 años de trabajo informático en la APS cubana. No obstante, se comenzó a conformar varios proyectos informáticos y un proyecto de informatización más completo y actualizado bajo el impacto de Internet. [39]

# CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

En el 2003 se decide centralizar el desarrollo y completamiento del sistema para la informatización de la APS en un Proyecto de Sistema de Información Único Gerencial del SNS, integrado por los niveles de Atención Secundaria de Salud, Atención Terciaria de Salud e instancias administrativas, con la colaboración mutua de la Empresa Softel, Copextel y la Universidad de Ciencias Informáticas. [40]

A pesar de que se realizan varios intentos en Cuba y a nivel mundial para darle una solución correcta a la gestión y clasificación de los problemas de salud en la atención primaria, tanto para el proceso de dispensarización de un paciente como para el de diagnóstico, las soluciones en el mundo informático se quedan truncadas ante la incapacidad de resolver los problemas más concretos y vitales dentro de este nivel. Entre las necesidades más apreciables se encuentran: el manejo fácil de la codificación de problemas de salud, la integración de sistemas de la atención primaria con otros sistemas de salud, la división de los problemas de salud surgidos por la idiosincrasia de cada país (en el caso de Cuba por riesgos, enfermedades y discapacidades), la forma de relacionar códigos de problemas de salud de la atención primaria con códigos de la CIE para el entendimiento común y mundial y el renombramiento de los problemas de salud más inherentes a la nación, para el uso más eficiente en la sociedad contemporánea.

## **1.5 Aspectos Generales del Sistema Nacional de Salud.**

El Ministerio de Salud Pública (MINSAP) es el “Organismo rector del Sistema Nacional de Salud. Encargado de dirigir, ejecutar y controlar la aplicación de la política del Estado y del Gobierno en cuanto a la Salud Pública, el desarrollo de las Ciencias Médicas y la Industria Médico Farmacéutica”. [41]

El Sistema Nacional de Salud se estructura en tres niveles que se corresponden con la estructura político-administrativa del país. El nivel nacional está representado por el Ministerio de Salud Pública que es el órgano rector con funciones metodológicas, normativas y de coordinación y control, al cual se le subordinan directamente los centros universitarios, institutos de investigaciones, centros hospitalarios de asistencia médica altamente especializados, centros de distribución y comercializadoras de suministros y tecnologías médicas, así como otros centros y entidades nacionales destinados a actividades técnicas y de apoyo. [42]

# CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

Los otros dos niveles están representados por las direcciones provinciales y municipales de salud que agrupan a las instituciones de salud en su respectivo nivel y desde el punto de vista administrativo las estructuras de Gobierno se ponen en función de resolver las demandas y necesidades de la población de los diferentes niveles organizativos y asistenciales. [43]

Las áreas de salud y las unidades que prestan servicios en la atención primaria junto con la estructura de dirección de los Consejos Populares, se encargan de resolver los problemas más específicos de la población de su radio de acción. Además en la labor de los Equipos Básicos de Salud cooperan con su accionar los Delegados de Circunscripción pertenecientes a los Consejos Populares. [44]



Figura 1.2 Estructura Organizativa del MINSAP

El Sistema Nacional de Salud se organiza en 3 niveles de atención:

**Atención Primaria:** Se brinda a nivel de los policlínicos, policlínicos con camas y/o hospitales rurales a través del Programa de Medicina Familiar y abarca a todos los Equipos Básicos de Salud (EBS).

# CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

Constituye el primer contacto del paciente sano o enfermo con el sistema de salud, que se brinda en locales adaptados para consultas o en el domicilio de los pacientes, en cualquier unidad del sistema de salud, aunque generalmente se realiza en el Consultorio Médico.

La función principal de la Atención Primaria es la promoción-prevención de salud en las diferentes comunidades, se realizan procedimientos de diagnósticos y terapéuticos que no requieren técnicas complejas, que aplicadas con calidad resuelven la mayor parte de los padecimientos que afectan a las poblaciones. Se diagnostican enfermedades graves que son derivadas a niveles de atención superiores, realizan seguimiento a personas con padecimientos crónicos y otorga bienestar a pacientes con patologías incurables. En general tiene carácter ambulatorio y comprende tanto a personas aparentemente sanas como a enfermas y/o discapacitadas. [45]

La atención primaria de salud es un nivel cualitativamente superior de atención médica, cuya esencia radica en la participación activa de la comunidad; donde las poblaciones de objetos pasivos, en espera de que se le ofrezcan soluciones, pasan a ser sujetos protagónicos activos ante sus propios problemas de salud. Decir participación comunitaria, es decir liderazgo, comunicación, cambio de hábitos y de estilos de vida, autorresponsabilidad y acción creadora. [46]

**Atención Secundaria:** Se brinda a nivel de las instituciones hospitalarias, por lo general son de carácter provincial, atienden a toda la población de una provincia determinada. Se proporciona en un segundo escalón, al cual el paciente tiene acceso a través de una remisión del personal médico de la atención primaria o sin ella, acudiendo directamente la persona necesitada de atención médica.

Puede tener carácter ambulatorio (policlínicos especializados, servicios externos hospitalarios) o de hospitalización. En el mismo se brindan procedimientos diagnósticos y terapéuticos de elevada complejidad, que dan respuesta a los problemas moderados y graves de salud. [47]

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

# CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

Podemos destacar entre los Principios Rectores del MINSAP: “el carácter estatal y social de la medicina, accesibilidad y gratuidad de los servicios, orientación profiláctica, aplicación adecuada de los adelantos de la ciencia y la técnica, participación de la comunidad e intersectorialidad, colaboración internacional y la centralización normativa y descentralización ejecutiva”. [48]

Tiene como Funciones Rectoras: “ejercer el control y la vigilancia epidemiológica de las enfermedades y sus factores de riesgo, la vigilancia sanitaria de todos los productos que pueden tener influencia sobre la salud humana, regulación y control de las investigaciones biomédicas, normar las condiciones higiénicas y el saneamiento del medio ambiente, regular el ejercicio de la medicina y de las actividades que le son afines y ejercer la evaluación, el registro, la regulación y el control de los medicamentos de producción nacional y de importación, equipos médicos y material gastable y otros de uso médico”. [49]

En el actual proceso de perfeccionamiento, el MINSAP se ha trazado como estrategias de desarrollo el perfeccionamiento de la atención primaria, la revitalización hospitalaria, el desarrollo del programa nacional de medicamentos y medicina natural y tradicional, el desarrollo de la tecnología de punta e investigación, así como contar con sistemas para urgencia, óptica, estomatología, asistencia social, control económico, atención al hombre y los cuadros.

## **1.5 Problema a Resolver y Situación Problémica.**

El desconocimiento de las posibilidades que tienen las Tecnologías de Información y Comunicaciones provoca que su introducción en las actividades de la salud sea recibida con desconfianza, escepticismo, resistencia y poca esperanza. Las limitaciones económicas y de información acerca del tema, la falta de experiencia y el desconocimiento de las posibilidades de la Informática por parte del personal de la salud, son dificultades a las que se enfrenta nuestra Revolución, en el empeño de informatizar paulatinamente la actividad de la Salud Pública.

Desde hace años diversas instituciones y organizaciones demuestran por estudios de investigación realizados y publicados, que existe una necesidad apreciable de utilizar metodologías adecuadas para diseñar soluciones informáticas en el sector de la salud, con mayor productividad y mejor calidad,

# CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

aprovechando los avances y métodos actuales en la Ingeniería de Programación. Por ello, se necesitan profesionales para el procesamiento cuidadoso y estudiado de la información médica más completa y estandarizada, así como soluciones que garanticen y mejoren la gestión de la información, administración, entre otras en este sector tan sensible para el bienestar del ser humano y de la sociedad.

Con el desarrollo de la informática y el aumento de su impacto social, cada vez son más las instituciones u organizaciones que optan por incorporar aplicaciones que gestionen su información, logrando así una mayor dinámica en sus procesos de negocio.

La informática desde hace varias décadas ha encontrado en la medicina una de sus aplicaciones más comunes e importantes que permite al sector de la salud, no sólo contar con métodos novedosos, sencillos y eficaces de gestión administrativa en consultas, hospitales y centros de investigación biomédica, sino también disponer de recursos informáticos de gran valor, en las exploraciones con tratamiento de imágenes, señales bioeléctricas y otras que reducen la posibilidad de error en el diagnóstico de las enfermedades, facilitan el trabajo de las estadísticas médicas y aumenta la confiabilidad de la misma ya que ahorra tiempo; el tiempo libre permite dedicar más tiempo al análisis de los datos, en vez de limitarse a la preparación y compilación de los mismos y son de gran apoyo en la toma de decisiones. También ofrecen una gran ayuda en el campo de la investigación epidemiológica, farmacéutica, biológica, química, etcétera, aspectos todos ellos relacionados con la lucha para conseguir un mejor nivel de salud en las personas y la excelencia en los servicios.

Durante los últimos 20 años un grupo de instituciones cubanas han desarrollado sistemas encaminados a lograr determinados niveles de informatización de la salud. Estas soluciones carecían de integración y de una definición generalizable, además de la no existencia en aquellos momentos de recursos tecnológicos necesarios para su ejecución en el Sistema Nacional de Salud. A partir de 1997, se concibe, una primera estrategia de informatización, como respuesta del sector de la salud a los lineamientos estratégicos para la informatización de la sociedad cubana, con la finalidad de coordinar esfuerzos para el desarrollo de este proceso en el SNS. [50]

En el marco de la informatización de la sociedad cubana, el sector de la salud se propone contar con una solución informática integral que dote al sistema de mayor grado de acceso a información unificada y

# CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

confiable en tiempo real, que aporte la rapidez y fiabilidad necesaria para las modernas técnicas de administración y para la toma de decisiones en los diferentes niveles; además de permitir que la información llegue de forma rápida, actualizada y de manera eficiente a todos los niveles requeridos, posibilitando la integridad y el control de la información y sirviendo de referencia a otras áreas con similares problemas, para la solución de sus necesidades. También se persigue el mejoramiento de la calidad de los servicios médicos a la población, tanto en Cuba como en otros países del área, facilitando la ejecución de muchos de los procesos que se realizan de forma manual y logrando almacenar la información de una forma más eficaz, ya que la misma se almacena en papel, lo que provoca el deterioro o la pérdida de información y que el trabajo sea mucho más complejo y engorroso.

Con la implementación a partir de 2003, de los programas de la revolución, que incluye como prioridad la informatización de los servicios, se inicia un amplio proceso, orientado en primer lugar a la superación y desarrollo profesional, que a su vez se irá extendiendo con la automatización de los servicios médicos, la investigación, la información científico-técnica y el apoyo en la toma de decisiones, comenzando en los policlínicos y expandiéndose al resto del sistema. [51]

Para ello se convocan a un grupo de instituciones propias del sector, del Ministerio de Informática y Comunicaciones (MIC) y de otros organismos de la administración central del estado, para definir de conjunto la estrategia a desarrollar.

Entre estas instituciones se destaca la Empresa Softel, dedicada al desarrollo de aplicaciones informáticas para la salud, a la cual se le asigna la misión de informatizar los procesos del Sistema Nacional de Salud. A raíz de esta tarea se forman varios grupos de desarrollo para informatizar los tres niveles de atención definidos por el Ministerio de Salud Pública (MINSAP) y se definen las estrategias a seguir; asumiendo el diseño y puesta en marcha de manera gradual de un Sistema de Gestión de la Información y el Conocimiento de nuevas dimensiones: el Sistema de Información para la Salud (SISalud), que incluye los sistemas informatizados para el nivel de atención primaria, secundaria y especializada y de los nomencladores que a nivel nacional deben definirse para garantizar el funcionamiento de los mismos de forma homogénea.

# CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

La Empresa Softel aportando profesionales con experiencia se vincula a los estudiantes y profesores de la Universidad de las Ciencias Informáticas (UCI), para aunar esfuerzos y de esta forma darle una correcta solución a las prioridades de informatización definidas; de manera que se logre la incorporación progresiva y sistemática de las TICs, en función de la adquisición y gestión del conocimiento y los servicios de salud, ésta propone aportar rapidez, fiabilidad y un mayor grado de acceso a la información pertinente y necesaria para las modernas técnicas de dirección. De esta integración surge, en la primera etapa de trabajo el Proyecto APS, con la misión de informatizar la Atención Primaria de Salud.

En paralelo con la prioridad de la informatización de la Atención Primaria surgen también varios sistemas que sirven de base a estos sistemas de la APS, como es el Caso del Registro de Ciudadano, Registro de Personal de la Salud, Registro de Equipos Médicos, Registro de la Clasificación Internacional de Enfermedades y Problemas Relacionados con la Salud, etc. Asumidos en su mayoría por el mismo Proyecto APS.

En el proceso de desarrollo de las aplicaciones informáticas asumido por el Proyecto APS, se define la dificultad para la gestión de la información relacionada con los problemas de salud, para la clasificación en grupos homogéneos de los pacientes, que facilite el proceso de dispensarización y diagnóstico, procesos que presuponen diferenciar los problemas de un paciente en riesgos, enfermedades y discapacidades. Esta estructura, no se presenta en los sistemas elaborados mundialmente para la gestión de los problemas de salud en este nivel, por ser un elemento particular del Sistema Nacional de Salud Cubano.

Por otra parte los sistemas realizados anteriormente a nivel nacional e internacional, presentan complejidad en la definición de los nombres de los problemas de salud, que en ocasiones son registrados con muchas especificaciones y detalles que no son importantes en el entorno en que se manejan y que en diversas situaciones no se corresponden con la idiosincrasia del país, debido a que deben regirse por estándares internacionales que ofrece la OMS, como es el caso de la Clasificación Estadística Internacional de Enfermedades y Problemas Relacionados con la Salud en la décima revisión (CIE-10), que brinda una clasificación muy detallada y de difícil manejo, trayendo consigo que el trabajo del médico sea más complejo. Al mismo tiempo, al realizar los diagnósticos de salud sin regirse por estos estándares impiden el establecimiento de un lenguaje común entre las organizaciones mundiales. En resumen las clasificaciones existentes no permiten utilizar de forma eficiente, fácil y sencilla los problemas de salud.

# CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

La situación descrita con anterioridad nos permite definir el Problema a Resolver de la siguiente forma: el sistema actual para la gestión de la información relacionada con los problemas de salud no garantiza de forma eficiente los procesos de dispensarización y de diagnóstico en la atención primaria, afectando la estandarización y homogeneidad de la información en este nivel de atención.

Como resultado de la problemática planteada, se espera obtener un producto que garantice la gestión de los problemas de salud de manera eficiente para la dispensarización de un paciente, así como para otros procesos propios de la atención primaria relacionados con el diagnóstico; permitiendo el manejo adecuado de la codificación y el renombramiento de los mismos acorde al país específico. Facilitando además la integración con otros módulos de la atención primaria, la reestructuración de los problemas de salud en riesgos, enfermedades y discapacidades y la forma de relacionar códigos de problemas de salud de la atención primaria con códigos de la CIE-10, para el entendimiento común y mundial.

El paciente será el principal beneficiado con este producto, ya que al garantizar la calidad, oportunidad y consistencia de la información, se incrementará la efectividad y eficiencia de los procesos relacionados con la salud, trayendo como consecuencia un incremento continuo y sostenido de la calidad en la atención médica.

## **1.6 Objeto de Automatización e Información Manipulada.**

A partir de las necesidades existentes y luego de un análisis profundo del alcance de la investigación, se propone automatizar el proceso de gestión de la información relacionada con los problemas de salud de la atención primaria, en el Sistema Nacional de Salud. Dentro del mismo intervienen a su vez dos procesos fundamentales: la clasificación y subdivisión de los problemas de salud en riesgos, enfermedades y discapacidades para la APS y la relación de la clasificación de la APS con esta nueva estructura con el estándar internacional CIE-10.

La clasificación de los problemas de salud en riesgos, enfermedades y discapacidades fue aprobada por la Dirección Nacional de Atención Primaria mediante investigaciones y reuniones de varios especialistas. Hoy en día estas son las clasificaciones que se utilizan en Cuba para llevar a cabo la dispensarización y

# CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

diagnóstico de un paciente en la APS, así como para toda la gestión de los problemas de salud que se lleva cabo en la misma. Esta nueva clasificación ofrecerá a otros sistemas una nueva estructura que facilitará el trabajo del médico en varios de los procesos que se llevan a cabo en este nivel, así como permitirá realizar una eficiente dispensarización y diagnóstico de los pacientes, explotando al máximo los recursos informáticos para darle un mejor seguimiento y así mejorar la calidad de vida de la población.

La relación del estándar internacional CIE-10 con la clasificación de la APS que presenta una nueva estructura de riesgos, enfermedades y discapacidades, es de vital importancia para mantener un lenguaje común a nivel internacional, así como para dar informes y reportes a organizaciones mundiales. De esta forma se establece una adecuada comunicación con las diferentes organizaciones internacionales que utilizan esta clasificación y al mismo tiempo nos permite manejar los problemas de salud de una manera más inherente a nuestro país, posibilitando un mejor desempeño profesional del médico de la familia.

Para llevar a cabo este proceso de automatizar la gestión de la información relacionada con los problemas de salud de la atención primaria, un conjunto de expertos funcionales del MINSAP realizaron un estudio, como consecuencia del mismo se derivaron los documentos de los riesgos, las enfermedades y discapacidades que se manejarán para la atención primaria. El resultado de este trabajo fue confirmado por especialistas de la Dirección Nacional de la Atención Primaria y de Estadísticas. Estos documentos sentaron las bases para definir el objetivo de la automatización, así como propiciaron conclusiones importantes que marcaron pautas en el desarrollo del sistema.

## **1.7 Tendencias y Tecnologías.**

En la informática el uso de los términos tendencia y tecnología es muy común. En este campo se conoce como tecnologías al conjunto de instrumentos, recursos técnicos o procedimientos empleados y las tendencias son los elementos informáticos que llevan la dirección o rumbo de dicha rama hacia el avance futuro. Para su mejor entendimiento se exponen en varios subepígrafes.

# CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

## 1.7.1 Estilos Arquitectónicos.

La programación, es sin dudas para muchos, considerada un arte. Un tiempo atrás se tornaba muy difícil para la mayoría de las personas, pero con el tiempo se han ido desarrollando metodologías y fórmulas o trucos para conseguir propósitos de la mejor forma posible. A todas estas técnicas se les ha llamado arquitectura de software.

Una arquitectura de software, consiste en un conjunto de patrones y abstracciones coherentes, que proporcionan el marco de referencia necesario para guiar la construcción del software de un sistema de información. Establece los fundamentos para que analistas, diseñadores, programadores y todos los que intervienen en el ciclo de vida de desarrollo del software, trabajen en una línea común que permita alcanzar los objetivos y necesidades del cliente.

La arquitectura para el desarrollo de la solución propuesta se definió por el MINSAP y entre sus principales características se destacan:

**Arquitectura Basada en Componentes (CBA):** Tiene como objetivo construir aplicaciones complejas mediante ensamblado de módulos (componentes), que han sido previamente diseñados por otras personas a fin de ser reusados en múltiples aplicaciones. Cada componente debe describir de forma completa las interfaces que ofrece, así como las interfaces que requiere para su operación y debe operar correctamente con independencia de los mecanismos internos que utilice para soportar la funcionalidad de la interfaz. Es actualmente una de las más prometedoras técnicas para incrementar la calidad del software, abreviar los tiempos de acceso al mercado y gestionar el continuo incremento de su complejidad.

**Arquitectura Orientada a Servicios (SOA):** es un concepto de arquitectura de software que define la utilización de servicios para dar soporte a los requerimientos de software del usuario. SOA proporciona una metodología y un marco de trabajo para documentar las capacidades de negocio y puede dar soporte a las actividades de integración y consolidación.

# CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

**Arquitectura en Tres Capas:** es un modelo de desarrollo de software en el que se identifican tres capas: capa de presentación o cliente, capa de negocio o capa intermedia y capa de datos. Provee de una gran flexibilidad y escalabilidad en la configuración y posibilita balancear la carga de la aplicación en varios servidores.

## 1.7.2 Metodologías.

A nivel mundial existen diversas metodologías para el desarrollo de un software. A pesar de que existen divergencias y diferencias notables entre una y otra, hay una que constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos, la misma se denomina Proceso Unificado de Desarrollo (RUP), la cual no es un proceso con pasos firmemente establecidos, sino un conjunto de metodologías adaptables al contexto y a las necesidades de cada organización.

RUP se caracteriza por ser iterativo e incremental, estar centrado en la arquitectura y guiado por los casos de uso, en el se destacan dos elementos esenciales los artefactos y los roles. Los artefactos son los productos tangibles del proceso y los roles son el papel que desempeña una persona en un determinado momento, teniendo en cuenta que una persona puede desempeñar distintos roles a lo largo del proceso.

RUP divide el proceso de desarrollo en ciclos, obteniendo un producto al final de cada ciclo. Costa de cuatro fases que finalizan con un hito, donde se debe tomar una decisión importante, las mismas son: inicio, elaboración, construcción y transición.

## 1.7.3 Lenguajes.

**PHP (Personal Home Page):** Es el acrónimo de Hypertext Preprocessor. Es un lenguaje de programación del lado del servidor gratuito e independiente de plataforma, rápido, con una gran librería de funciones y mucha documentación. Se considera un lenguaje interpretado y embebido en el HTML.

Algunas de las más importantes capacidades de PHP son: compatibilidad con las bases de datos más comunes, como por ejemplo MySQL, MSSQL, MS SQL, Oracle, Informix, y ODBC. Incluye funciones para el

# CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

envío de correo electrónico, upload de archivos, crear dinámicamente en el servidor imágenes en formato GIF, incluso animadas y una lista interminable de utilidades adicionales.

El PHP pertenece a los lenguajes de software libre, corre en 7 plataformas, funciona en 11 tipos de servidores, ofrece soporte sobre unas 20 Bases de Datos y contiene unas 40 extensiones estables sin contar las que se están experimentando, además posee simplicidad, rapidez, sintaxis está inspirada en C y librería estándar amplia. Presenta una de las comunidades más grandes en Internet, por lo que es fácil encontrar ayuda, documentación, artículos, noticias, y demás recursos.

**XSL (Extensible Stylesheet Language):** Ha surgido como una solución para transformar documentos XML a otro formato como: HTML, WML, texto simple, PDF e inclusive en otro documento XML con diferentes parámetros.

El lenguaje XSL está dividido en tres partes: XSLT (Extensible Stylesheet Lenguaje Transformation), el cual define la transformación de documentos en XML, XPath (XML Path Language), donde se especifica como acceder cierto punto de la estructura del documento XML y XSLF (Extensible Stylesheet Lenguaje Format), que precisa el formato que deben tomar los objetos dentro del documento en XML. Generalmente, se necesita un procesador de hojas de estilo, o stylesheet processor para procesarlas, aplicándolas a un fichero XML.

**XHTML (Extensible HyperText Markup Language):** Es el lenguaje de marcado pensado para sustituir a HTML como estándar para las páginas web. XHTML es la versión XML de HTML, por lo que tiene, básicamente, las mismas funcionalidades, pero cumple las especificaciones, más estrictas, de XML. Su objetivo es avanzar en el proyecto del World Wide Web Consortium de lograr una web semántica, donde la información, y la forma de presentarla estén claramente separadas. En este sentido, XHTML serviría únicamente para transmitir la información que contiene un documento, dejando para hojas de estilo y JavaScript su aspecto y diseño en distintos medios (ordenadores, PDAs, teléfonos móviles, impresoras).

**JavaScript:** Es un lenguaje de programación interpretado, con capacidades elementales orientadas a objeto. El código JavaScript es embebido directamente en el código HTML, haciendo fácil la creación de páginas web con contenido dinámico. Está diseñado para controlar la apariencia y manipular los eventos

# CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

dentro de la ventana del navegador web y es soportado por la gran mayoría de los navegadores. Este lenguaje ligado con el HTML nos permite hacer aplicaciones web más potentes.

**UML (Unified Modeling Lenguaje):** Es un lenguaje gráfico para especificar, construir, visualizar y documentar las partes o artefactos (información que se utiliza o produce mediante un proceso de software). Pueden ser artefactos: un modelo, una descripción que comprenda el desarrollo de software que se base en el enfoque Orientado a Objetos. UML usa procesos de otras metodologías, aprovechando la experiencia de sus creadores, eliminó los componentes que resultaban de poca utilidad práctica y añadió nuevos elementos. Es un lenguaje más expresivo, claro y uniforme que los anteriores definidos para el diseño Orientado a Objetos, que no garantiza el éxito de los proyectos, pero si mejora sustancialmente el desarrollo de los mismos, al permitir una nueva y fuerte integración entre las herramientas, los procesos y los dominios.

De forma general sus principales características son: lenguaje unificado para la modelación de sistemas, tecnología orientada a objetos, corrección de errores viables en todas las etapas y aplicable para tratar asuntos de escala inherentes a sistemas complejos de misión crítica, tiempo real y cliente-servidor.

## 1.7.4 Servidores.

**Servidor Web Apache:** Es el servidor web libre más utilizado en el mundo. Su costo gratuito, gran fiabilidad y extensibilidad lo convierten en una herramienta potente y muy configurable.

Dentro de sus puntos o características fuertes se encuentran: posee interfaz con todos los sistemas de autenticación, facilita la integración como "plug-ins" de los lenguajes de programación de páginas web dinámicas más comunes, presenta integración con el estándar del protocolo de seguridad Secure Sockets Layer (SSL), provee interfaz a todas las bases de datos y posee Virtual Host.

Apache fue hecho para proveer un alto grado de calidad y fortaleza para las implementaciones que utilizan el protocolo HTTP. Está ligado a la plataforma (Linux, Windows, UNIX) sobre la cual los individuos o instituciones pueden construir sistemas confiables con fines experimentales o para resolver un problema específico de la organización.

# CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

**Servidor de Bases de Datos MySQL:** Está basado en Open Source (Código abierto) diseñado para los sistemas Unix formando parte de la tecnología LAMP (Linux, Apache, MySQL y PHP), aunque existen versiones para Windows. Actualmente está en su versión 5.0.6-beta incluyendo procedimientos almacenados (stored procedures), disparadores (triggers), vistas (views) y otras características.

Presenta las siguientes ventajas: diseñado con el objetivo de aumentar la velocidad, consume muy pocos recursos de CPU y memoria, por lo que posee un buen rendimiento, tamaño del registro sin límite, buena integración con PHP, utilidades de administración (PhpMyAdmin) y buen control de acceso usuarios-tablas-permisos.

## 1.7.5 PLASER.

**PLASER (Plataforma de Servicios):** Está conformada fundamentalmente por varias clases en PHP, es una librería que puede o no ser usada, pero de no ser usada, la seguridad corre por parte del programador. En esta versión, PLASER sólo soporta comunicación a través del protocolo SOAP, pero 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, entre otros.

Está concebida completamente sobre Arquitectura Basada en Componentes y Orientada a Servicios, usando el paradigma de XML - Web Services específicamente SOAP. En su concepción se han utilizado estándares actuales y normas abiertas. PLASER constituye una plataforma sobre la que se pueden desplegar aplicaciones XML - Web Services, además facilita la programación y homogeneidad de los componentes. 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.

# CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

## 1.7.6 XML - Web Services.

XML (Extensible Markup Lenguaje), se ha convertido en un formato estándar en Internet y está diseñado para representar datos estructurados, no es un lenguaje de marcado como su nombre lo indica; es un metalenguaje para definir otros lenguajes de marcados adecuados a un uso en específico, éste es la base de los servicios web. XML, al que algunos consideran el Esperanto de los sistemas de información, se emplea principalmente para separar el contenido de la presentación de forma total, o sea, permite representar datos de forma homogénea en entornos heterogéneos, lo que facilita la interoperabilidad entre distintos sistemas.

Los servicios Web XML (XML - Web Services) actúan de forma independiente y además permiten que las aplicaciones compartan información e invoquen funciones de otras aplicaciones independientemente del sistema operativo o la plataforma en que se ejecutan y los dispositivos utilizados para obtener acceso a ellos, o sea, son rutinas en Internet actuando como catalizadores de transacciones vía web.

Los Servicios Web usan SOAP (Simple Object Access Protocol) como protocolo de transporte estándar por su simplicidad, se puede identificar un mensaje SOAP como un documento XML conformado por una envoltura “envelope” obligatoria, un encabezamiento “header” opcional y un cuerpo “body”. La forma de acceder a ellos es a través del WSDL (Web Services Description Languages). Estos servicios deben publicar una interfaz que funja como un contrato de servicio, donde se describan cada una de las funciones que provee, además de las funciones que estos ofrecen, como realizar el intercambio de mensajes, especificar el contenido de una petición y el aspecto de la respuesta en una notación inequívoca. Además de describir el contenido de los mensajes, WSDL define dónde está disponible el servicio y qué protocolo de comunicaciones utilizar para hablar con el servicio. Esto significa que el archivo WSDL define todo lo necesario para escribir un programa que interactúe con un Servicio Web.

## 1.7.7 Herramientas.

Las herramientas son un punto importante en la elaboración de una aplicación, son los programas que se reutilizan para automatizar las actividades definidas en el proceso de desarrollo de software; nos permiten

# CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

crear y darle soporte al mismo, muchas veces haciendo el trabajo más factible y sencillo. Las siguientes son las que se utilizan en todo el flujo de desarrollo.

**MySQL-Front versión 4.0.15:** Es un entorno gráfico (similar a Windows) para el manejo y gestión de bases de datos MySQL. A diferencia de otros sistemas, no está programado en PHP y HTML, por lo que la respuesta es mucho más rápida y eficaz.

Sus características principales son: soporte para bases de datos MySQL a partir de la versión 3.23 y de todo tipo de tablas y columnas MySQL, dispone la opción de conectar mediante MySQL DLL, funciona con cualquier versión de Windows, usa un gran número de funciones API de Windows, para asegurar compatibilidad con futuras versiones de Windows, prohíbe la conexión a la Base de Datos por ISP desde el exterior, dispone de interfaces de conexión basado en PHP, soporte para 25 idiomas y corrector de gramática SQL coloreando la sintaxis.

Con MySQL-Front se pueden realizar acciones básicas como añadir, borrar o modificar tablas, campos, registros y además ver variables del servidor, ejecutar y terminar procesos, ejecutar SQL-scripts, exportar tablas a SQL-scripts o a otras bases de datos, replicar bases de datos, guardar datos en formato HTML o CSV (ideal para Excel), escribir consultas en SQL, importar datos de ODBC, realizar diagnóstico de tablas (optimización, reparación, etc.) y ver propiedades avanzadas de tablas (tipo, comentario, key\_length, etc.).

**Stylus Studio:** Es un completo entorno de desarrollo integrado que incluye un potente editor de XML, un "debugger" XSLT y otras muchas herramientas pensadas especialmente para facilitar y mejorar la productividad en el desarrollo de sitios web y aplicaciones. Permite la edición de XML en modo visual y sincronizado y un completo set de herramientas para desarrollo en XSLT, entre las que se incluyen un debugger, un mapeador y una utilidad de diseño de hojas de estilo de HTML a XSLT. Soporta además edición visual de XQuery, e incluye un editor DTD, utilidades para XPath, y mucho más.

**Macromedia Dreamweaver 8:** Es un software fácil de usar que permite crear páginas Web profesionales. Las funciones de edición visual 8 permiten agregar rápidamente diseño y funcionalidad a las páginas, sin la necesidad de programar manualmente el código HTML. Se pueden crear tablas, editar marcos, trabajar con capas, insertar comportamientos JavaScript, etc., de una forma muy sencilla y visual. Presenta

# CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

mejoras respecto a Dreamweaver MX y las CSS. Esta última versión ha mejorado mucho respecto a la compatibilidad y manejo de estilos de cascada. Se ha añadido una barra de herramientas, esta barra hace mucho más accesible el código al permitirnos la navegación por etiquetas y su contracción. Permite convertir un documento HTML a XHTML y crear y validar documentos XSL.

**NuSphere Phped versión 3.3:** Es un editor para programadores con soporte para múltiples formatos, similar a otras aplicaciones como PHP Coder. PHPEd facilita el trabajo de programación con numerosas características de gran utilidad entre las que se destacan: completo sistema de ayuda, plantillas de documento y de fragmentos de código frecuentes y código de colores para comandos en PHP, Perl, JavaScript, SQL y HTML. Además, esta herramienta incluye un cliente de FTP y un servidor web integrado, totalmente configurables según las necesidades de trabajo.

**Rational Rose Enterprise Edition 2003:** Se utiliza para sustentar la documentación, como modelador visual de la notación UML (Unified Modeling Language) para la confección de los diagramas. Son varias herramientas encargadas de generar el código una vez se haya realizado el modelado de objetos y los distintos diagramas correspondientes al desarrollo. Es una herramienta que se podría encuadrar dentro del grupo de herramientas más técnicas debido a que se encarga de llevar a cabo tanto la automatización de los sistemas para la posterior generación de código (esto es, realización de los distintos diagramas y generación del código posterior), como para labores de ingeniería inversa (es decir, realización de los diagramas una vez conocido el código).

## Conclusiones.

En este capítulo se definieron los conceptos básicos, el proceso de la dispensarización y diagnóstico de los pacientes en Cuba. Se realizó un estudio del estado del arte de las aplicaciones existentes, debido a las diferentes clasificaciones en el mundo sobre los problemas de salud. Además, se analizaron aspectos generales del Sistema Nacional de Salud, del problema a resolver y de la situación problemática; así como del objeto de automatización, la información manipulada, las tendencias y las tecnologías.

# Capítulo 2. Características del Sistema

## Introducción.

El resultado final de un proyecto de software, es un producto que toma forma a lo largo de su ciclo de desarrollo. La eficacia del mismo, está estrechamente ligada a la calidad del proceso de arranque y de avance, entre otros aspectos.

Hoy en día, los sistemas informáticos contribuyen activamente al éxito de los objetivos de una empresa, organización, institución y hasta de una nación en algunas ocasiones, la tendencia actual de los mismos está dirigida a aplicaciones web. Un elemento vital a tener en cuenta para hacer buenas aplicaciones es, sin duda, el conocimiento a profundidad del negocio del problema que se plantea en cada situación, así como la buena caracterización de los requisitos y funcionalidades del sistema, para así poder sugerir soluciones más certeras y eficientes. Por su importancia el capítulo 2 tiene entre unos de sus propósitos adentrarse en el negocio, la especificación de requerimientos funcionales y no funcionales y en el modelo del sistema del problema a resolver.

## 2.1 Modelo de Dominio.

El primer paso en el proceso de desarrollo de software es precisamente alcanzar cierto nivel de conocimiento sobre el problema en cuestión. Téngase en cuenta que los profesionales de la informática, tienen la difícil tarea de interactuar con personal que, aunque perfectamente capacitado en su campo de acción, raramente domina terminología de la especialidad de informática. Son los ingenieros informáticos los que deben ser capaces de entrar en su entorno e identificar el problema para luego ofrecer una solución eficiente.

Para lograr una concepción general del problema, los analistas de sistemas se valen fundamentalmente de un conjunto de técnicas y habilidades que se van perfeccionando en la práctica y con la experiencia en procesos de desarrollo de software, no obstante no basta con alcanzar una visión individual, el analista además debe ser capaz de transmitir, al resto del equipo de desarrollo sus impresiones, en un lenguaje entendible tanto para los desarrolladores como para el cliente.

## CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

---

Para la comprensión del contexto del sistema los analistas utilizan la técnica del Modelado del Negocio, llamando negocio a cualquier ambiente o entorno en el cual está enmarcado el problema y procesos de negocio a las funciones que allí se desarrollan.

En algunos casos definir el Negocio se convierte en un problema y se torna un poco complejo cuando el negocio no está bien definido o cuando sus fronteras no están bien delimitadas. Por lo que los especialistas utilizan en estos casos el Modelado del Dominio, que es un subconjunto del Modelado del Negocio y constituye un modelo conceptual donde se destacan como componentes fundamentales los conceptos y sus relaciones entre ellos, integrándose los mismos en el Modelo de Dominio, el cual podría considerarse como un diccionario visual de las abstracciones relevantes, con vocabulario e información estrictamente del dominio.

### 2.1.1 Conceptos Fundamentales.

Los conceptos son un elemento importante en la realización del Modelo de Dominio, ya que nos permite de manera visual, representar sus principales características, que se manejan en el dominio del sistema en desarrollo. Luego de un análisis profundo del problema que se plantea se han derivado los siguientes:

**Riesgo:** “representación de una medida que refleja la probabilidad de que ocurra un daño a la salud (enfermedad, discapacidad, muerte, etc)”. [52]

**Enfermedad:** “proceso y el estado consecuente de afección de un ser vivo, caracterizado por una alteración de su estado ontológico de salud o de algunos de sus órganos, que interrumpe o acelera el rendimiento de las funciones vitales”. [53]

**Discapacidad:** “limitación física o mentalmente de una función básica o invalidación de la misma por completo. No es sólo la consecuencia de los déficits existentes en la persona, sino la resultante de un conjunto de condiciones, actividades y relaciones interpersonales, muchas de las cuales están motivadas por factores ambientales”. [54]

## CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

**RCIE-Enfermedades:** representa un contenedor de información de las enfermedades de CIE-10 y su relación con la Clasificación de Problemas de Salud de la Atención Primaria.

**RCIE-Discapacidades:** representa un contenedor de información de las discapacidades de CIE-10 y su relación con la Clasificación de Problemas de Salud de la Atención Primaria.

**Especialista de la Dirección Nacional de Atención Primaria:** representa el actor que interactúa con más frecuencia y directamente con el sistema, ya que se encarga de insertar, eliminar, modificar e imprimir la información de los riesgos, enfermedades y discapacidades, así como relacionarla con el CIE-10.

### 2.1.2 Diagrama del Modelo de Dominio.

Un diagrama del modelo de dominio es una representación de las clases conceptuales del mundo real, no de componentes software. No se trata de un diagrama que describen clases de software u objetos software con responsabilidades, se trata de una representación de conceptos puramente del dominio, donde se establece una terminología estrictamente del dominio. [55]

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. A continuación se muestra el mismo:



Figura 2.1 Diagrama del Modelo de Dominio

# CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

---

## 2.2 Propuesta del Sistema.

La parte más difícil de construir un sistema software es decidir qué construir y cómo construirlo. Ninguna otra parte del trabajo afecta más negativamente al sistema final si se realiza de manera incorrecta este paso, además de que ninguna otra parte se torna tan difícil de rectificar después. Es por eso que el flujo de requerimientos en su totalidad constituye un punto fundamental a tener siempre en cuenta en la elaboración de un producto de software.

El coste de un cambio en los requisitos, una vez entregado el producto, es entre 60 y 100 veces superior al coste que hubiera representado el mismo durante las fases iniciales de desarrollo.

El flujo de requerimientos en su ciclo de vida de desarrollo tiene como procesos básicos y elementales a la especificación de los requisitos funcionales y no funcionales y a la definición o propuesta de sistema, convirtiéndose los mismos en dos pasos indispensables para alcanzar el éxito, así como la satisfacción del cliente. Este flujo presenta varios objetivos, como son establecer y mantener un acuerdo con los clientes y los otros interesados acerca de qué debe hacer el sistema, proveer a los desarrolladores del sistema de un mejor entendimiento de los requerimientos del sistema, definir los límites (o delimitar) del sistema, proveer una base para la planeación de los contenidos técnicos de las iteraciones, proveer una base para la estimación de costo y tiempo necesarios para desarrollar el sistema y definir una interfaz de usuario para el sistema, enfocada en las necesidades y objetivos del usuario.

### 2.2.1 Especificación de Requerimientos de Software.

El proceso para la especificación de requerimientos se encarga de establecer los servicios que un sistema debe suministrar, así como las restricciones bajo las cuales debe operar el mismo, donde la meta final es construir un documento con la Especificación de Requerimientos de Software (ERS), esencial para el logro de un producto de software correcto, cualquiera que sea su área de aplicación.

La obtención de requerimientos es un paso muy importante para el posterior desarrollo de las siguientes etapas, pues un error en estas fases iniciales puede dar al traste con un sistema que no cumpla las

## CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

---

expectativas de los usuarios y difícilmente aporte valor agregado al negocio para el que debe ser concebido.

Un requerimiento según la IEEE Standard Glossary of Software Engineering Terminology se define como: “condición o capacidad que necesita un usuario para resolver un problema o lograr un objetivo, condición o capacidad que tiene que ser alcanzada o poseída por un sistema o componente de un sistema para satisfacer un contrato, estándar, u otro documento impuesto formalmente y una representación documentada de una condición o capacidad”. [56]

Todas las ideas que los clientes, usuarios y miembros del equipo de proyecto tengan acerca de lo que debe hacer el sistema, deben ser analizadas como candidatas a requisitos. Los requisitos se pueden clasificar en: funcionales y no funcionales.

### 2.2.1.1 Requerimientos Funcionales.

Los requerimientos funcionales son capacidades o condiciones que el sistema debe cumplir, los mismos no alteran la funcionalidad del producto, esto quiere decir que los requerimientos funcionales se mantienen invariables sin importarle con que propiedades o cualidades se relacionen. [57]

A partir del estudio y las investigaciones realizadas en el negocio del problema, se obtuvieron los siguientes requerimientos funcionales que ha de cumplir el sistema, con vista a la satisfacción final del cliente:

| N°         | REQUERIMIENTOS FUNCIONALES   |
|------------|--|
| <b>RF1</b> | Listar la información de los riesgos por taxonomía de la Clasificación de Problemas de Salud de la Atención Primaria.        |
| <b>RF2</b> | Listar la información de las enfermedades por taxonomía de la Clasificación de Problemas de Salud de la Atención Primaria.   |
| <b>RF3</b> | Listar la información de las discapacidades por taxonomía de la Clasificación de Problemas de Salud de la Atención Primaria. |

## CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

---

|              |   |
|--------------|---|
| <b>RF4</b>   | Gestionar la información de los riesgos de la Clasificación de Problemas de Salud de la Atención Primaria.        |
| <b>RF4.1</b> | Insertar la información de los riesgos de la Clasificación de Problemas de Salud de la Atención Primaria.         |
| <b>RF4.2</b> | Eliminar la información de los riesgos de la Clasificación de Problemas de Salud de la Atención Primaria.         |
| <b>RF4.3</b> | Modificar la información de los riesgos de la Clasificación de Problemas de Salud de la Atención Primaria.        |
| <b>RF5</b>   | Buscar la información de los riesgos de la Clasificación de Problemas de Salud de la Atención Primaria.           |
| <b>RF6</b>   | Gestionar la información de las enfermedades de la Clasificación de Problemas de Salud de la Atención Primaria.   |
| <b>RF6.1</b> | Insertar la información de las enfermedades de la Clasificación de Problemas de Salud de la Atención Primaria.    |
| <b>RF6.2</b> | Eliminar la información de las enfermedades de la Clasificación de Problemas de Salud de la Atención Primaria.    |
| <b>RF6.3</b> | Modificar la información de las enfermedades de la Clasificación de Problemas de Salud de la Atención Primaria.   |
| <b>RF7</b>   | Buscar la información de las enfermedades de la Clasificación de Problemas de Salud de la Atención Primaria.      |
| <b>RF8</b>   | Gestionar la información de las discapacidades de la Clasificación de Problemas de Salud de la Atención Primaria. |
| <b>RF8.1</b> | Insertar la información de las discapacidades de la Clasificación de Problemas de Salud de la Atención Primaria.  |
| <b>RF8.2</b> | Eliminar la información de las discapacidades de la Clasificación de Problemas de Salud de la Atención Primaria.  |

## CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

---

|               |   |
|---------------|---|
| <b>RF8.3</b>  | Modificar la información de las discapacidades de la Clasificación de Problemas de Salud de la Atención Primaria. |
| <b>RF9</b>    | Buscar la información de las discapacidades de la Clasificación de Problemas de Salud de la Atención Primaria.    |
| <b>RF10</b>   | Gestionar la información de los grupos de la Clasificación de Problemas de Salud de la Atención Primaria.         |
| <b>RF10.1</b> | Insertar la información de los grupos de la Clasificación de Problemas de Salud de la Atención Primaria.          |
| <b>RF10.2</b> | Eliminar la información de los grupos de la Clasificación de Problemas de Salud de la Atención Primaria.          |
| <b>RF10.3</b> | Modificar la información de los grupos de la Clasificación de Problemas de Salud de la Atención Primaria.         |
| <b>RF10.4</b> | Buscar la información de los grupos de la Clasificación de Problemas de Salud de la Atención Primaria.            |
| <b>RF11</b>   | Gestionar la información de los subgrupos de la Clasificación de Problemas de Salud de la Atención Primaria.      |
| <b>RF11.1</b> | Insertar la información de los subgrupos de la Clasificación de Problemas de Salud de la Atención Primaria.       |
| <b>RF11.2</b> | Eliminar la información de los subgrupos de la Clasificación de Problemas de Salud de la Atención Primaria.       |
| <b>RF11.3</b> | Modificar la información de los subgrupos de la Clasificación de Problemas de Salud de la Atención Primaria.      |
| <b>RF11.4</b> | Buscar la información de los subgrupos de la Clasificación de Problemas de Salud de la Atención Primaria.         |
| <b>RF12</b>   | Imprimir la información de los riesgos de la Clasificación de Problemas de Salud de la Atención Primaria.         |

## CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

---

|             |  |
|-------------|--|
| <b>RF13</b> | Imprimir la información de las enfermedades de la Clasificación de Problemas de Salud de la Atención Primaria.   |
| <b>RF14</b> | Imprimir la información de las discapacidades de la Clasificación de Problemas de Salud de la Atención Primaria. |
| <b>RF15</b> | Imprimir la información de los grupos de la Clasificación de Problemas de Salud de la Atención Primaria.         |
| <b>RF16</b> | Imprimir la información de los subgrupos de la Clasificación de Problemas de Salud de la Atención Primaria.      |

Tabla 2.1 Requerimientos Funcionales

### 2.2.1.2 Requerimientos No Funcionales.

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable. En muchos casos son fundamentales en el éxito del producto. Normalmente están vinculados a requerimientos funcionales, es decir una vez se conozca lo que el sistema debe hacer podemos determinar cómo ha de comportarse, qué cualidades debe tener o cuán rápido o grande debe ser. [58]

Los requerimientos no funcionales forman una parte significativa de la especificación. Son importantes para que clientes y usuarios puedan valorar las características no funcionales del producto, pues si se conoce que el mismo cumple con la toda la funcionalidad requerida, las propiedades no funcionales, como cuán usable, seguro, conveniente y agradable, pueden marcar la diferencia entre un producto bien aceptado y uno con poca aceptación.

Existen múltiples categorías para clasificar a los requerimientos no funcionales, siendo las siguientes las más representativas dentro del conjunto de los mismos.

## CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

| REQUERIMIENTOS NO FUNCIONALES |   |
|-------------------------------|---|
| <b>N°</b>                     | <b>RNF1</b>   |
| <b>Categoría</b>              | <b>Usabilidad</b>   |
| <b>Descripción</b>            | Cada usuario que se autentique sólo tendrá acceso a la información que le corresponde, ya sea editor o visualizador a nivel nacional, provincial, municipal o de unidad de salud. |
| <b>N°</b>                     | <b>RNF2</b>   |
| <b>Categoría</b>              | <b>Soporte</b>  |
| <b>Descripción</b>            | El personal que trabaja con el módulo debe contar con el nivel técnico requerido mediante adiestramiento de servicio.   |
| <b>N°</b>                     | <b>RNF3</b>   |
| <b>Categoría</b>              | <b>Portabilidad</b>   |
| <b>Descripción</b>            | Permitir que el sistema se ejecute sobre el Sistema Operativo Windows 98 o superior o Linux.  |
| <b>N°</b>                     | <b>RNF4</b>   |
| <b>Categoría</b>              | <b>Seguridad</b>  |
| <b>Subcategoría</b>           | <b>Confidencialidad</b>   |
| <b>Descripción</b>            | Disponer de un mecanismo de seguridad basado en el modelo de Autenticación, Autorización y Auditoría (AAA).   |
| <b>N°</b>                     | <b>RNF4.1</b>   |
| <b>Categoría</b>              | <b>Seguridad</b>  |
| <b>Subcategoría</b>           | <b>Confidencialidad</b>   |
| <b>Descripción</b>            | La autenticación será la primera acción del usuario en el sistema y consistirá en suministrar un nombre de usuario único y una contraseña que debe ser de                         |

## CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

|                     |   |
|---------------------|---|
|                     | conocimiento exclusivo de la persona que se autentica.  |
| <b>N°</b>           | <b>RNF4.1.1</b>   |
| <b>Categoría</b>    | <b>Seguridad</b>  |
| <b>Subcategoría</b> | <b>Confidencialidad</b>   |
| <b>Descripción</b>  | Si el usuario autenticado no se encuentra registrado se debe reportar un error de acceso.   |
| <b>N°</b>           | <b>RNF4.1.2</b>   |
| <b>Categoría</b>    | <b>Seguridad</b>  |
| <b>Subcategoría</b> | <b>Confidencialidad</b>   |
| <b>Descripción</b>  | Si el usuario autenticado se encuentra registrado se autoriza su acceso y se crea un certificado digital.   |
| <b>N°</b>           | <b>RNF4.1.2.1</b>   |
| <b>Categoría</b>    | <b>Seguridad</b>  |
| <b>Subcategoría</b> | <b>Confidencialidad</b>   |
| <b>Descripción</b>  | <p>El certificado digital constará de:</p> <ul style="list-style-type: none"> <li>- Un identificador único (token) de 32 caracteres generado de manera aleatoria en el momento de la autorización.</li> <li>- El identificador del usuario.</li> <li>- El nivel de acceso del usuario (Nacional, Provincial, Municipal o Unidad de Salud).</li> <li>- El identificador del nivel de acceso.</li> <li>- Un listado de los módulos que el usuario tiene acceso y el tipo de acceso en cada uno de ellos (Editor o Visualizador).</li> </ul> |
| <b>N°</b>           | <b>RNF4.1.2.2</b>   |
| <b>Categoría</b>    | <b>Seguridad</b>  |
| <b>Subcategoría</b> | <b>Confidencialidad</b>   |

## CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

|                     |   |
|---------------------|---|
| <b>Descripción</b>  | Deberá registrarse el token, el identificador del usuario presente en el certificado digital y el día, mes, año, hora, minuto, segundo de la creación del certificado.                      |
| <b>N°</b>           | <b>RNF4.2</b>   |
| <b>Categoría</b>    | <b>Seguridad</b>  |
| <b>Subcategoría</b> | <b>Confidencialidad</b>   |
| <b>Descripción</b>  | Para cada petición que se haga al sistema, debe enviarse el token recibido en el certificado digital el cual será validado con la lista de tokens autorizados que el sistema ha registrado. |
| <b>N°</b>           | <b>RNF4.2.1</b>   |
| <b>Categoría</b>    | <b>Seguridad</b>  |
| <b>Subcategoría</b> | <b>Confidencialidad</b>   |
| <b>Descripción</b>  | Si el token se encuentra registrado y la petición se corresponde con los derechos del usuario se procederá a ejecutar la petición solicitada.   |
| <b>N°</b>           | <b>RNF4.2.2</b>   |
| <b>Categoría</b>    | <b>Seguridad</b>  |
| <b>Subcategoría</b> | <b>Confidencialidad</b>   |
| <b>Descripción</b>  | Si el token no se encuentra registrado o la petición no se corresponde con los derechos del usuario se debe reportar un error de acceso.  |
| <b>N°</b>           | <b>RNF4.3</b>   |
| <b>Categoría</b>    | <b>Seguridad</b>  |
| <b>Subcategoría</b> | <b>Confidencialidad</b>   |
| <b>Descripción</b>  | Cada Petición de usuario, autorizada o no, será registrada, así como el día, mes, año, hora, minuto, segundo en que se registra y si fue o no autorizada.                                   |
| <b>N°</b>           | <b>RNF5</b>   |

## CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

---

|                     |  |
|---------------------|--|
| <b>Categoría</b>    | <b>Seguridad</b>   |
| <b>Subcategoría</b> | <b>Integridad</b>  |
| <b>Descripción</b>  | Prevenir posibles fallos y recuperarse ante ellos.   |
| <b>N°</b>           | <b>RNF5.1</b>  |
| <b>Categoría</b>    | <b>Seguridad</b>   |
| <b>Subcategoría</b> | <b>Integridad</b>  |
| <b>Descripción</b>  | La información manejada por el sistema será objeto de cuidadosa protección contra la corrupción y estados inconsistentes. Deberán existir mecanismos de chequeo de integridad.   |
| <b>N°</b>           | <b>RNF5.2</b>  |
| <b>Categoría</b>    | <b>Seguridad</b>   |
| <b>Subcategoría</b> | <b>Integridad</b>  |
| <b>Descripción</b>  | Deberá existir una estrategia de replicación que permita, de manera transparente para el usuario final, balancear la carga de acceso entre múltiples servidores aumentando los tiempos de respuesta y facilitar la recuperación inmediata del sistema si falla uno de ellos. |
| <b>N°</b>           | <b>RNF5.3</b>  |
| <b>Categoría</b>    | <b>Seguridad</b>   |
| <b>Subcategoría</b> | <b>Integridad</b>  |
| <b>Descripción</b>  | Se permitirá la creación de copias de respaldo que puedan restaurar el sistema en caso de fallo crítico o pérdida total de la información.   |
| <b>N°</b>           | <b>RNF6</b>  |
| <b>Categoría</b>    | <b>Seguridad</b>   |
| <b>Subcategoría</b> | <b>Disponibilidad</b>  |

## CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

|                    |   |
|--------------------|---|
| <b>Descripción</b> | La seguridad no implicará lentitud o retraso en la repuesta dada por el sistema, por lo que se debe minimizar y reducir el tiempo de respuesta, así como optimizar el código.   |
| <b>N°</b>          | <b>RNF7</b>   |
| <b>Categoría</b>   | <b>Apariencia o Interfaz Externa</b>  |
| <b>Descripción</b> | Todas las funciones se realizarán desde el portal principal y una vez realizada la opción, se vuelve a él.  |
| <b>N°</b>          | <b>RNF8</b>   |
| <b>Categoría</b>   | <b>Apariencia o Interfaz Externa</b>  |
| <b>Descripción</b> | Diseñar una interface donde el personal que trabaja con la aplicación, seleccione y organice la información que desee mostrar con sólo hacer clic.  |
| <b>N°</b>          | <b>RNF9</b>   |
| <b>Categoría</b>   | <b>Apariencia o Interfaz Interna</b>  |
| <b>Descripción</b> | Todos los componentes del sistema deben desarrollarse siguiendo el principio de máxima cohesión y mínimo acoplamiento.  |
| <b>N°</b>          | <b>RNF10</b>  |
| <b>Categoría</b>   | <b>Apariencia o Interfaz Interna</b>  |
| <b>Descripción</b> | Los componentes reutilizables entre los módulos que componen el Sistema de Información para la Salud (SISalud) deberán ser desarrollados como servicios web XML que interactúan a través de SOAP con otros componentes. |
| <b>N°</b>          | <b>RNF11</b>  |
| <b>Categoría</b>   | <b>Ayuda y Documentación en Línea</b>   |
| <b>Descripción</b> | Disponer de instrucciones en una opción de ayuda.   |
| <b>N°</b>          | <b>RNF12</b>  |
| <b>Categoría</b>   | <b>Ayuda y Documentación en Línea</b>   |

## CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

|                    |   |
|--------------------|---|
| <b>Descripción</b> | Disponer de una documentación del sistema, documentada con alguna metodología de desarrollo.  |
| <b>N°</b>          | <b>RNF13</b>  |
| <b>Categoría</b>   | <b>Software</b>   |
| <b>Descripción</b> | Los clientes tendrán acceso al Sistema de Información para la Salud (SISalud) a través de cualquier navegador web, recomendado Mozilla 1.5, Internet Explorer 5.0 o superior.                                 |
| <b>N°</b>          | <b>RNF14</b>  |
| <b>Categoría</b>   | <b>Software</b>   |
| <b>Descripción</b> | Las aplicaciones legacy (aplicaciones basadas en tecnologías y hardware más viejos) deberán integrar los componentes que estén desarrollados según los requerimientos tecnológicos que está evaluando SOFTEL. |
| <b>N°</b>          | <b>RNF15</b>  |
| <b>Categoría</b>   | <b>Hardware</b>   |
| <b>Descripción</b> | Se permitirá aumentar la cantidad de servidores o adicionar componentes de hardware en función de disminuir el tráfico o balancear la carga, sin que sea necesario realizar modificaciones al software.       |
| <b>N°</b>          | <b>RNF16</b>  |
| <b>Categoría</b>   | <b>Hardware</b>   |
| <b>Descripción</b> | Impresora local o de red para imprimir la información o los reportes solicitados.   |
| <b>N°</b>          | <b>RNF17</b>  |
| <b>Categoría</b>   | <b>Restricciones en el diseño y la implementación</b>   |
| <b>Descripción</b> | La lógica de presentación constituirá una capa independiente de la lógica de negocio, centrandó su función en la interfaz de usuario y validaciones simples de los datos de entrada.                          |

## CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

|                    |  |
|--------------------|--|
| <b>N°</b>          | <b>RNF18</b>   |
| <b>Categoría</b>   | <b>Restricciones en el diseño y la implementación</b>  |
| <b>Descripción</b> | Validar el proceso de la captación de datos para evitar entradas inadecuadas.                                    |
| <b>N°</b>          | <b>RNF19</b>   |
| <b>Categoría</b>   | <b>Restricciones en el diseño y la implementación</b>  |
| <b>Descripción</b> | Se utilizarán herramientas de desarrollo que garanticen la calidad de todo el ciclo de desarrollo del producto.  |
| <b>N°</b>          | <b>RNF20</b>   |
| <b>Categoría</b>   | <b>Rendimiento</b>   |
| <b>Descripción</b> | El sistema debe soportar la conexión simultánea de más de 60 000 usuarios.                                       |
| <b>N°</b>          | <b>RNF21</b>   |
| <b>Categoría</b>   | <b>Rendimiento</b>   |
| <b>Descripción</b> | El sistema debe soportar un tiempo de respuesta menor o igual a 30 segundos.                                     |
| <b>N°</b>          | <b>RNF22</b>   |
| <b>Categoría</b>   | <b>Rendimiento</b>   |
| <b>Descripción</b> | El sistema debe soportar respuestas SOAP que no deben exceder los 50 Kbyte en el tiempo de respuesta al usuario. |

Tabla 2.2 Requerimientos No Funcionales

### 2.2.2 Modelo de Casos de Uso del Sistema.

Un software suele ser extremadamente intrincado, por ello es necesario dividir el sistema en partes o fragmentos, si queremos entender y administrar su complejidad. Estas partes podemos representarlas como modelos, que describan y abstraigan sus aspectos esenciales.

## CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

---

Por tanto, un paso útil en la construcción de un sistema de software es el de crear el Modelo de Casos de Uso del Sistema, ya que organiza y comunica los detalles importantes del sistema a construir y representa los procesos más importantes que se informatizarán en el futuro.

El Modelo de Casos de Uso del Sistema nos brinda la primera visión del software que se quiere construir, ya que define lo que debemos automatizar, sirviendo así de base a otros flujos del ciclo de vida de desarrollo de software.

### 2.2.2.1 Definición de Actores del Sistema.

El actor es una entidad externa del sistema que de alguna manera participa en la historia del caso de uso, por tal razón siempre presenta una relación con al menos uno de estos. Simbolizan un rol de una o varias personas, un equipo o sistema automatizado que interactúa con el sistema, por lo que no forma parte de este. Puede intercambiar información o ser un recipiente pasivo de datos. [59] Los actores del sistema que intervienen en este entorno son los siguientes:

| ACTORES DEL SISTEMA   | JUSTIFICACIÓN   |
|---|---|
| Especialista de la Dirección Nacional de Atención Primaria. | Es el encargado de toda la gestión de la información, por tal razón es el que más interactúa directamente con el sistema; insertando, eliminando y modificando los datos. Además puede realizar búsquedas, listar por taxonomía e imprimir información de los riesgos, enfermedades y discapacidades, así como establecer la relación de las enfermedades y discapacidades con los problemas de salud del CIE-10. |
| Usuario.  | Es el que necesita consultar información de los riesgos, enfermedades y discapacidades, pero que solamente puede visualizar esta información, así como realizar búsquedas e imprimir la misma.  |
| Registro de la Clasificación Internacional                  | Sistema externo con el cual se interactúa y de donde se   |

## CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

---

|   |  |
|---|--|
| de Enfermedades y Problemas Relacionados con la Salud (RCIE). | extrae información de los problemas de salud estandarizados a nivel internacional, utilizando así sus WebServices. |
|---|--|

Tabla 2.3 Justificación de Actores del Sistema

### 2.2.2.2 Diagrama de Casos de Uso del Sistema.

Un diagrama de casos de uso explica gráficamente un conjunto de casos de uso de un sistema, los actores y las relaciones entre estos y los casos de usos. Estos últimos se muestran en óvalos representando procesos y los actores son figuras estilizadas. Por otro lado estos elementos no deben representarse de forma aislada en este diagrama, si por el contrario esto ocurriera constituirá un error en este modelo o en los requerimientos planteados. Las líneas simbolizan las comunicaciones entre los casos de uso y los actores, no poseen nombre y sólo se necesita especificar el punto de inicio y el fin de la relación; las saetas indican el flujo de la información o el estímulo.

El diagrama tiene por objeto ofrecer una clase de diagrama contextual que nos permite conocer rápidamente los actores externos de un sistema y las formas básicas en que lo utilizan. Los profesionales de la informática en su mayoría dedican un esfuerzo considerable a la realización del mismo, ya que es rico en características visuales que nos permite tener una concepción general del sistema. Luego de varios análisis se ha obtenido como resultado el siguiente diagrama:

# CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

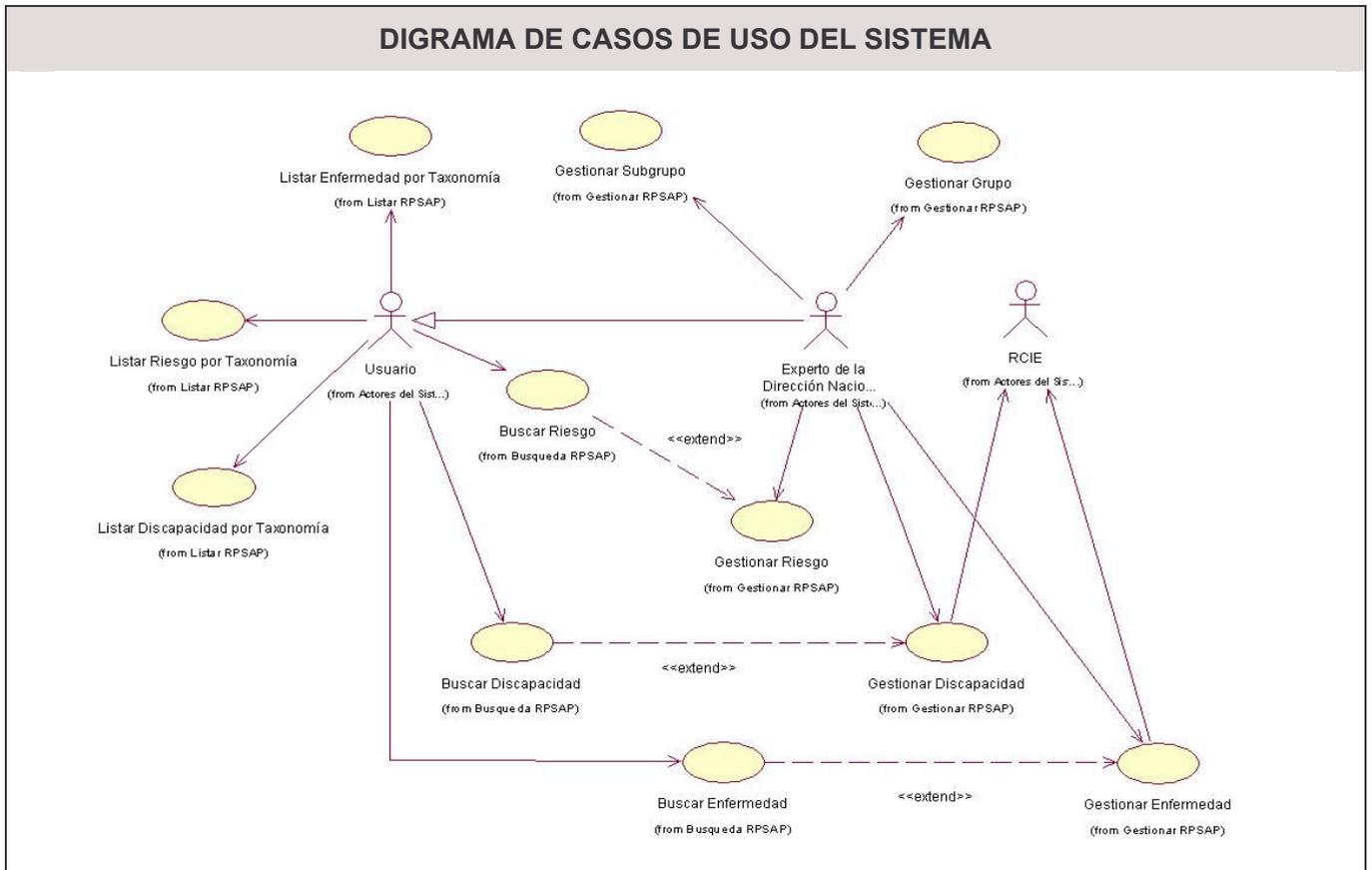


Figura 2.2 Diagrama de Casos de Uso del Sistema

### 2.2.2.3 Descripción Textual de los Casos de Uso.

Para entender la funcionalidad asociada a cada caso de uso no es suficiente con la representación gráfica del Diagrama de Casos de Uso del Sistema. Para enriquecer la documentación y entendimiento de estos se realiza una descripción detallada, denominada “Descripción Textual”, donde se explica el flujo de eventos correspondiente a cada caso de uso y se incluyen: precondiciones, post-condiciones y requisitos funcionales asociados a los mismos. A continuación se plantean un resumen de las descripciones textuales de algunos de los casos de uso del sistema.

|                                |                                    |
|--------------------------------|------------------------------------|
| <b>Nombre del Caso de Uso:</b> | <b>Listar Riesgo por Taxonomía</b> |
| <b>Actores:</b>                | Usuario.                           |

## CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

|                           |  |
|---------------------------|--|
| <b>Propósito:</b>         | Brindarle al usuario la posibilidad de que liste los riesgos definidos para la atención primaria de forma taxonómica. La estructura taxonómica nos muestra los riesgos agrupados en sus respectivos grupos.                        |
| <b>Breve descripción:</b> | El Caso de Uso comienza cuando el usuario selecciona la opción para listar los riesgos, el sistema hace una búsqueda en la base de datos y muestra los riesgos agrupados en sus respectivos grupos, culminando así el caso de uso. |
| <b>Precondiciones:</b>    | El usuario debe ser autenticado en el sistema.   |
| <b>Poscondiciones:</b>    | El sistema retorna los datos de los riesgos encontrados en la Base de Datos.   |
| <b>Requisitos:</b>        | <b>RF1</b>   |
| <b>Prioridad</b>          | Crítico  |

Tabla 2.4 Descripción Textual del Caso de Uso Listar Riesgo por Taxonomía

| <b>Nombre del Caso de Uso:</b> | <b>Buscar Riesgo</b>   |
|--------------------------------|--|
| <b>Actores:</b>                | Usuario.   |
| <b>Propósito:</b>              | Brindarle al usuario la posibilidad de que busque los riesgos definidos para la atención primaria y que pueda acotar la búsqueda con parámetros de entrada, además de facilitar la impresión de datos en pdf o excel.  |
| <b>Breve descripción:</b>      | El Caso de Uso comienza cuando el usuario selecciona la opción para buscar los riesgos, luego escoge dentro de la interfaz mostrada la opción "Mostrar Todos" o inserta parámetros de búsquedas para acotar la misma y ejecuta la opción "Buscar". El sistema hace una búsqueda en la base de datos y muestra el nombre del riesgo, la observación y el grupo a que pertenece, posibilitando la impresión de datos en pdf o excel de manera opcional a través de la opción |

## CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

|                        |  |
|------------------------|--|
|                        | “Exporta a”, culminando así el caso de uso.                                  |
| <b>Precondiciones:</b> | El usuario debe ser autenticado en el sistema.                               |
| <b>Poscondiciones:</b> | El sistema retorna los datos de los riesgos encontrados en la Base de Datos. |
| <b>Requisitos:</b>     | <b>RF5</b>   |
| <b>Prioridad</b>       | Crítico  |

Tabla 2.5 Descripción Textual del Caso de Uso Buscar Riesgo

|                                |  |
|--------------------------------|--|
| <b>Nombre del Caso de Uso:</b> | <b>Listar Discapacidad por Taxonomía</b>   |
| <b>Actores:</b>                | Usuario.   |
| <b>Propósito:</b>              | Brindarle al usuario la posibilidad de que liste las discapacidades definidas para la atención primaria de forma taxonómica. La estructura taxonómica nos muestra las discapacidades agrupados en sus respectivos subgrupos y grupos.                        |
| <b>Breve descripción:</b>      | El Caso de Uso comienza cuando el usuario selecciona la opción para listar las discapacidades, el sistema hace una búsqueda en la base de datos y muestra las discapacidades agrupadas en sus respectivos subgrupos y grupos, culminando así el caso de uso. |
| <b>Precondiciones:</b>         | El usuario debe haber sido autenticado en el sistema.  |
| <b>Poscondiciones:</b>         | El sistema retorna los datos de las discapacidades encontradas en la Base de Datos.  |
| <b>Requisitos:</b>             | <b>RF3</b>   |
| <b>Prioridad</b>               | Crítico  |

Tabla 2.6 Descripción Textual del Caso de Uso Listar Discapacidad por Taxonomía

|                                |                            |
|--------------------------------|----------------------------|
| <b>Nombre del Caso de Uso:</b> | <b>Buscar Discapacidad</b> |
| <b>Actores:</b>                | Usuario.                   |

## CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

|                           |   |
|---------------------------|---|
| <b>Propósito:</b>         | Brindarle al usuario la posibilidad de que busque las discapacidades definidas para la atención primaria y que pueda acotar la búsqueda con parámetros de entrada, además de facilitar la impresión de datos en pdf o excel.  |
| <b>Breve descripción:</b> | El Caso de Uso comienza cuando el usuario selecciona la opción para buscar las discapacidades, luego escoge dentro de la interfaz mostrada la opción "Mostrar Todos" o inserta parámetros de búsquedas para acotar la misma y ejecuta la opción "Buscar". El sistema hace una búsqueda en la base de datos y muestra el nombre de la discapacidad, la observación y el subgrupo a que pertenece, además del código y descripción CIE que presenta como relación de la CIE-10, posibilitando la impresión de datos en pdf o excel de manera opcional a través de la opción "Exporta a", culminando así el caso de uso. |
| <b>Precondiciones:</b>    | El usuario debe ser autenticado en el sistema.  |
| <b>Poscondiciones:</b>    | El sistema retorna los datos de las discapacidades encontradas en la Base de Datos.   |
| <b>Requisitos:</b>        | <b>RF9</b>  |
| <b>Prioridad</b>          | Crítico   |

Tabla 2.7 Descripción Textual del Caso de Uso Buscar Discapacidad

| Nombre del Caso de Uso:   | Gestionar Riesgo  |
|---------------------------|---|
| <b>Actores:</b>           | Especialista de la Dirección Nacional de Atención Primaria.   |
| <b>Propósito:</b>         | Mantener actualizada la información referente a los riesgos definidos en la atención primaria, así como posibilitar la realización de las operaciones de insertar, modificar y eliminar los mismos. |
| <b>Breve descripción:</b> | El Caso de Uso comienza cuando el actor selecciona la opción para insertar o modificar un riesgo.   |

## CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

|                        |  |
|------------------------|--|
|                        | <p>Si escoge la opción para modificar los riesgos, ejecuta primero el caso de uso “Buscar Riesgo”, luego selecciona la opción “Agregar” o elige el riesgo que quiere actualizar y ejecuta la opción de “Eliminar” o “Modificar”, actualizando así la base de datos y mostrando en la interfaz de la búsqueda los datos actualizados, culminando así el caso de uso.</p> <p>Si escoge la opción para agregar los riesgos, inserta los datos en la interfaz mostrada y ejecuta la opción “Aceptar”, actualizando de esta forma la base de datos y mostrando en la interfaz de la búsqueda los datos actualizados, culminando así el caso de uso.</p> |
| <b>Precondiciones:</b> | El usuario debe haber sido autenticado en el Sistema.  |
| <b>Poscondiciones:</b> | El sistema actualiza la información de los riesgos en la Base de Datos.  |
| <b>Requisitos:</b>     | <b>RF4</b>   |
| <b>Prioridad</b>       | Crítico  |

Tabla 2.8 Descripción Textual del Caso de Uso Gestiona Riesgo

|                                |  |
|--------------------------------|--|
| <b>Nombre del Caso de Uso:</b> | <b>Gestionar Discapacidad</b>  |
| <b>Actores:</b>                | Especialista de la Dirección Nacional de Atención Primaria.  |
| <b>Propósito:</b>              | Mantener actualizada la información referente a las discapacidades definidas en la atención primaria, así como posibilitar la realización de las operaciones de insertar, modificar y eliminar las mismas. |
| <b>Breve descripción:</b>      | <p>El Caso de Uso comienza cuando el actor selecciona la opción para insertar o modificar una discapacidad.</p> <p>Si escoge la opción para modificar las discapacidades, ejecuta</p>                      |

## CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

|                        |  |
|------------------------|--|
|                        | <p>primero el caso de uso “Buscar Discapacidad”, luego selecciona la opción “Agregar” o elige la discapacidad que quiere actualizar y ejecuta la opción de “Eliminar” o “Modificar”, actualizando así la base de datos y mostrando en la interfaz de la búsqueda los datos actualizados, culminando así el caso de uso.</p> <p>Si escoge la opción para agregar las discapacidades, inserta los datos en la interfaz y ejecuta la opción “Aceptar”, actualizando de esta forma la base de datos y mostrando en la interfaz de la búsqueda los datos actualizados, culminando así el caso de uso.</p> |
| <b>Precondiciones:</b> | El usuario debe haber sido autenticado en el Sistema.  |
| <b>Poscondiciones:</b> | El sistema actualiza la información de las discapacidades en la Base de Datos.   |
| <b>Requisitos:</b>     | <b>RF8</b>   |
| <b>Prioridad</b>       | Crítico  |

Tabla 2.9 Descripción Textual del Caso de Uso Gestionar Discapacidad

### Conclusiones.

En este capítulo, se han explicado los procesos de dos de los flujos del ciclo de desarrollo de un software: la Captura de Requerimientos y el Modelamiento del Negocio. El primero de los cuales contiene dos actividades principales: la especificación de requerimientos funcionales y no funcionales. Además se realizó el modelamiento del sistema, donde se incluyen la definición de actores, el diagrama de casos de uso del sistema y la descripción textual de los mismos. Luego en el flujo de Modelamiento del Negocio, que presenta como tareas principales establecer un entendimiento con el cliente y la organización; y proporcionar una concepción profunda del negocio planteado, se realizó el Modelo de Dominio, incluyendo dos aspectos fundamentales: el diagrama del modelo del Dominio y las descripciones de los conceptos en torno al dominio.

# Capítulo 3. Diseño del Sistema

## Introducción.

Cada paso dentro del ciclo de vida de desarrollo de un proyecto se torna vital en el éxito de un producto final, tal es el caso del Flujo de Análisis y Diseño. Dentro de este, el diseño es una parte importante, que tiene como objetivo general modelar el sistema, encontrar su forma para que soporte todos sus requerimientos, tanto los funcionales como no funcionales, siendo el centro de atención de la fase de inicio y elaboración.

Por su importancia, este capítulo se dedicará al Diseño del Sistema, en el se explicará la estructura, los patrones y una representación de los diagramas de clases del diseño. También, se hará una breve descripción de las tablas de la bases de datos y de las clases del diseño, se mostrará el diagrama de clases persistentes y el modelo de datos.

## 3.1 Modelo de Diseño.

El Modelo de Diseño es otro de los tantos modelos que se tienen en cuenta en el ciclo de vida del desarrollo de un software, consumiendo una elevada cantidad de esfuerzo para su realización. Este modelo es de vital importancia para los especialistas y desarrolladores, ya que pone de relieve una solución lógica: cómo el sistema cumple con los requerimientos; además define e identifica las consecuencias del ambiente de implementación. Es un modelo físico, no genérico y específico para una implementación, más formal, dinámico, que contiene un sinnúmero de estereotipos físicos sobre las clases en dependencia del lenguaje utilizado y ha sido creado principalmente como programación visual en procesos iterativos e incrementales.

En concreto, el diseño tiene el propósito de formar la base para la implementación, desarrollar la arquitectura, adaptar el diseño para que sea consistente con el entorno de implementación, dándole soporte a todos los requisitos, incluyendo los no funcionales y las restricciones que se le suponen y descomponer los trabajos a desarrollar en partes más manejables que puedan ser llevadas a cabo por diferentes equipos de desarrollo, teniendo en cuenta la posible concurrencia. [60]

### 3.1.1 Patrones de Diseño Utilizados.

Los especialistas informáticos a través de su experiencia van formando un amplio repertorio de principios generales y de expresiones que los guía al crear software, a los cuales se le denomina patrones. Los mismos han tomado un gran auge en desarrollo de sistemas informáticos. Dentro de esta rama los patrones de diseño se destacan por su relevante utilidad en el modelamiento de diagramas de clases.

Los patrones de diseño describen un problema que ocurre repetidas veces en algún contexto determinado de desarrollo de software, y entregan una buena solución ya probada. Esto ayuda a diseñar correctamente en menos tiempo y a construir problemas reutilizables y extensibles, y facilita la documentación. [61]

#### 3.1.1.1 Modelo Vista Controlador.

La lógica de una 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, entonces la consecuencia será que, cuando necesitemos cambiar la interfaz, tendremos que modificar trabajosamente los componentes de negocio. Mayor trabajo y más riesgo de error. Es por eso que muchos especialistas usan para el diseño de aplicaciones con sofisticadas interfaces el patrón de diseño Modelo-Vista-Controlador.

Se trata de realizar un diseño que desacople 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.

Aunque la aproximación de ligar la lógica de negocio con la interfaz puede ser válida para implementar sistemas presenta varios inconvenientes: no permite reutilizar código en cada uno de los manejadores y se complica cuando se realizan operaciones que implican llamar a varios servicios de la lógica de negocio, sobre el navegador recae la responsabilidad de controlar la navegación, por lo que se crea dependencia entre los formularios, lo que hace que sea muy difícil modificar el flujo de navegación, añadir nuevos formularios o cambiar el comportamiento de alguno y obliga a los formularios a acceder de forma

## CAPÍTULO 3. DISEÑO DEL SISTEMA

---

separada a la lógica de negocio, por lo que se tendrá una copia distinta de los datos y dificultará la sincronización de las vistas cuando se modifiquen.

Para solucionar estos inconvenientes se puede utilizar un patrón de diseño llamado Modelo Vista Controlador (MVC). Este patrón de diseño propone dividir la estructura de la aplicación cliente en tres clases distintas:

**Modelo:** Gestiona el comportamiento y los datos de la aplicación, responde a las peticiones que realizan las vistas sobre su estado y permite su actualización normalmente desde el controlador. [62]

**Controlador:** Interpreta las acciones del usuario, accediendo a las operaciones de negocio de la aplicación y modificando a partir de sus resultados el estado del modelo y la navegación entre vistas. [63]

**Vista:** Muestra el estado al usuario de la aplicación, redirigiendo las acciones que realiza sobre la interfaz al controlador. Puede ser desarrollada por un equipo de diseñadores independiente de los programadores. [64]

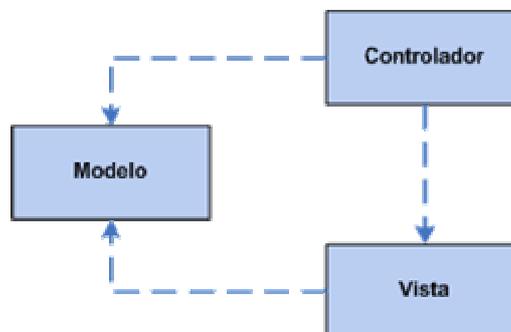


Figura 3.1 Patrón Modelo Vista-Controlador

Este patrón Modelo – Vista – Controlador se tuvo en cuenta para realizar el módulo RPSAP, aplicándose en el mismo para permitir la reutilización de código y separar partes susceptibles para reducir el impacto en el cambio. En el sistema están bien definidos los tres componentes del patrón, la vista constituida por los ficheros XSL y JavaScript, clientes PHP de la capa de presentación y la FachadaRPSAP, el controlador identificados por los métodos del negocio y el modelo conformado por la base de datos de RPSAP. En la aplicación podemos ver varios ejemplos de los beneficios de su aplicabilidad, un ejemplo

## CAPÍTULO 3. DISEÑO DEL SISTEMA

---

clave es como las interfaces se han separado y desligado del código PHP y de la lógica de negocio, permitiendo que las mismas sean fácil de cambiar, así como de darle soporte, lo que permite reducir el impacto al cambio en el código de los demás componentes. En general en el módulo se pueden ver diferentes aplicaciones de este patrón que han permitido que el sistema construya sus componentes de forma reutilizables, disminuyendo el riesgo de error y optimizando el código de forma coherente y apropiada.

### 3.1.1.2 Bajo Acoplamiento.

El patrón Bajo Acoplamiento pertenece al conjunto de patrones GRASP, acrónimo que significa General Responsibility Assignment Software Patterns (patrones generales de software para asignar responsabilidades). Este patrón hace referencia a las relaciones que tienen las clases o componentes entre sí dentro de un sistema, mide qué tan fuerte está una clase o componente conectadas con otras, es decir, cuántas clases o componentes conoce y necesita. Una clase o un componente con bajo o débil acoplamiento no depende de "muchas otras" clases o componentes. Una clase o componente con alto o fuerte acoplamiento recurre a muchas otras clases o componentes.

Teóricamente, cuando una serie de clases o componentes tienen una relación con varias clases o componentes, cuando estos son cambiados, las clases o componentes relacionados han de verse afectados necesariamente y son más difíciles de entender y reutilizar. Por ello la situación ideal es que cada clase o componente tenga las mínimas dependencias posibles con el resto de las clases, para poder realizar modificaciones sin necesidad de cambiar las restantes, manteniendo de esta forma una mínima repercusión en el cambio de los componentes del sistema.

En el módulo RPSAP también se ha usado este patrón, permitiendo que los componentes del mismo sean más independientes. Un ejemplo lo podemos ver en la conformación y realización de los métodos del negocio, los mismos están hechos con la menor dependencia en todos los casos, lo que ha permitido su reutilización en cada proceso y la reducción en el riesgo al error, ya que un error en uno de ellos no afecta la ejecución a los demás.

### 3.1.1.3 Alta Cohesión.

El patrón de Alta Cohesión es otro de los que conforman el grupo de patrones GRASP. Este patrón indica que cada clase del diseño debe realizar una labor única dentro del sistema, no desempeñada por el resto de los elementos y auto-identificable, además nos plantea que la información que almacena una clase debe de ser coherente y está en la mayor medida de lo posible relacionada con la clase. Ejemplos de una baja cohesión son clases que hacen demasiadas cosas o que están saturadas de métodos.

Cuando decimos que una clase tiene una alta cohesión queremos decir que el objeto tiene bien delimitadas sus responsabilidades. Es decir, la cohesión de una clase significa cuan relacionadas y enfocadas están las acciones de la misma, si implementamos el sistema con baja cohesión entonces complicamos sus clases, su organización y la posibilidad de mantenerlo en el futuro. Un buen diseño asignará a cada clase sólo las responsabilidades que realmente requiera cada una, desde el punto de vista conceptual no debemos sobrecargar de responsabilidades a la misma.

En el módulo RPSAP también se ha usado este patrón, permitiendo que los componentes del mismo tengan una única responsabilidad en el sistema. Un ejemplo lo podemos ver en la conformación y realización de los métodos del negocio, los mismos tiene bien delimitadas sus responsabilidades, no recargando en ningún caso su funcionalidad, para que su funcionamiento sea eficiente y el tiempo de respuesta y ejecución no se exceda del estimado en el sistema, reportando eficiencia y generando por ende bajo acoplamiento.

### 3.1.1.4 Fachada.

Otro de los que conforma el grupo de patrones GRASP es el patrón Fachada, también conocido como: no hables con extraños. Este patrón trata de simplificar la interfaz 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 sólo se ofrezca un (o unos pocos) punto de entrada al sistema

## CAPÍTULO 3. DISEÑO DEL SISTEMA

---

tapado por la fachada. El proceso consiste en asignar la responsabilidad a una clase directa para que colabore con una clase indirecta, de modo que no se necesite saber nada de la clase indirecta.

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

En la solución de este sistema se utiliza este patrón Fachada para abstraernos de la librería de clases PLASER. Este patrón está implementado en una clase que lleva el mismo nombre de fachada, permitiendo que la comunicación se establezca siempre con la clase fachada y no con las clases de PLASER directamente, muchas veces no conocidas en su totalidad por los desarrolladores. Además nos permite que nuestro sistema sea portable a cualquier cambio en la librería de clases de PLASER, por tanto reduce el impacto al cambio y permite la escalabilidad del sistema.

### 3.1.2 Definición de la Estructura del Diseño.

Los sistemas o arquitecturas en capas constituyen uno de los estilos que aparecen con mayor frecuencia en las aplicaciones de la actualidad. “Se define el estilo en capas como una organización jerárquica, tal que cada capa proporciona servicios a la capa inmediatamente superior y se sirve de las prestaciones que le brinda la inmediatamente inferior”. [65] Donde las restricciones del estilo pueden incluir una limitación, más o menos rigurosa, que exige a cada capa operar sólo con capas adyacentes, y a los elementos de una capa entenderse sólo con otros elementos de la misma. Su objetivo primordial es la separación de la lógica de negocios de la lógica de diseño.

El módulo RPSAP se dividió en tres capas para su mejor desempeño e implementación. La estructura por capa está representada por la Capa de Presentación, Capa de Negocio y Capa de Datos, además utiliza la librería de clases PlaSer, que sirve de soporte para la implementación.

**Capa de Presentación:** Esta capa reúne todos los aspectos del software que tienen que ver con las interfaces y la interacción con los diferentes tipos de usuarios. Por lo que presenta el sistema al usuario, le comunica la información y captura la información del usuario realizando un mínimo de proceso de filtrado

## CAPÍTULO 3. DISEÑO DEL SISTEMA

previo para comprobar que no hay errores de formato. Típicamente incluyen el manejo y aspecto de las ventanas, el formato de los reportes, menús, gráficos y elementos multimedia en general. Esta capa se comunica únicamente con la capa de negocio.

**Capa de Negocio:** Esta capa reúne todos los aspectos del software que automatizan o apoyan los procesos de negocio que llevan a cabo los usuarios. Es donde residen los programas que se ejecutan, recibiendo las peticiones del usuario y enviando las respuestas tras el proceso. Se denomina capa de negocio (e incluso de lógica del negocio), pues es aquí donde se establecen todas las reglas que deben cumplirse y las restricciones, así como las tareas que forman parte de los procesos. Esta capa se comunica con la capa de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de datos, para solicitar, almacenar o recuperar datos de él.

**Capa de Datos:** Esta capa reúne todos los aspectos del software que tienen que ver con el manejo de los datos persistentes. Es donde residen los datos. Está formado por el servidor de bases de datos que realiza todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

Las tres capas del sistema interactúan con la librería de clases PlaSer; la capa de presentación accede solamente a esta a través de la clase Fachada y la capa de negocio solamente accede a la capa de datos a través de la clase dbz\_class de la misma.

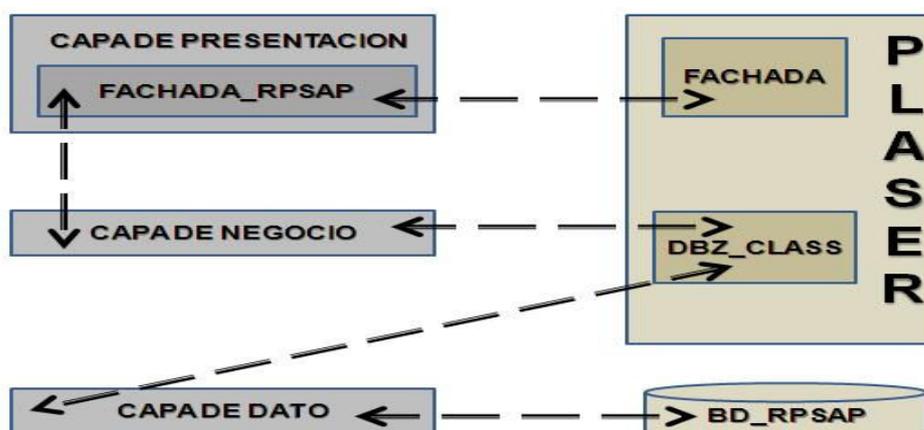


Figura 3.2 Estructura del Diseño

## CAPÍTULO 3. DISEÑO DEL SISTEMA

---

Las ventajas del estilo en capas son obvias. Primero que nada, el estilo soporta un diseño basado en capas de abstracción crecientes, lo cual a su vez permite a los implementadores la partición de un problema complejo en una secuencia de pasos incrementales, posibilitando así el desarrollo en paralelo en varias capas. También nos permite reducir al mínimo el impacto del cambio o error en algunas de las capas, ya que sólo se verá afectada la que contiene el mismo. En segundo lugar, el estilo admite muy naturalmente optimizaciones y refinamientos. En tercer lugar, proporciona amplia reutilización. Al igual que los tipos de datos abstractos, se pueden utilizar diferentes implementaciones o versiones de una misma capa en la medida que soporten las mismas interfaces de cara a las capas adyacentes. Esto conduce a la posibilidad de definir interfaces de capa estándar, a partir de las cuales se pueden construir extensiones o prestaciones específicas.

### 3.1.3 Diagramas de Clases del Diseño.

El diagrama de clase es el diagrama principal del diseño para un sistema, elaborado para satisfacer los detalles concretos de la implementación. En él se muestra la estructura estática del sistema, es donde se pueden ver las relaciones entre todas las clases, además de los atributos y operaciones de cada una de las mismas.

Los diagramas de clases del diseño describen gráficamente las especificaciones de las clases de software y de las interfaces en una aplicación. Normalmente contienen la información de: las clases, asociaciones y atributos, interfaces, con sus operaciones y constantes, métodos, información sobre los tipos de los atributos, navegabilidad y dependencia. [66]

A diferencia del modelo conceptual, un diagrama de este tipo contiene las definiciones de las entidades del software en vez de conceptos del mundo real. Seguidamente se muestra algunas de las representaciones de los mismos:

# CAPÍTULO 3. DISEÑO DEL SISTEMA

## DIAGRAMA DE CLASES DEL DISEÑO LISTAR RIESGO

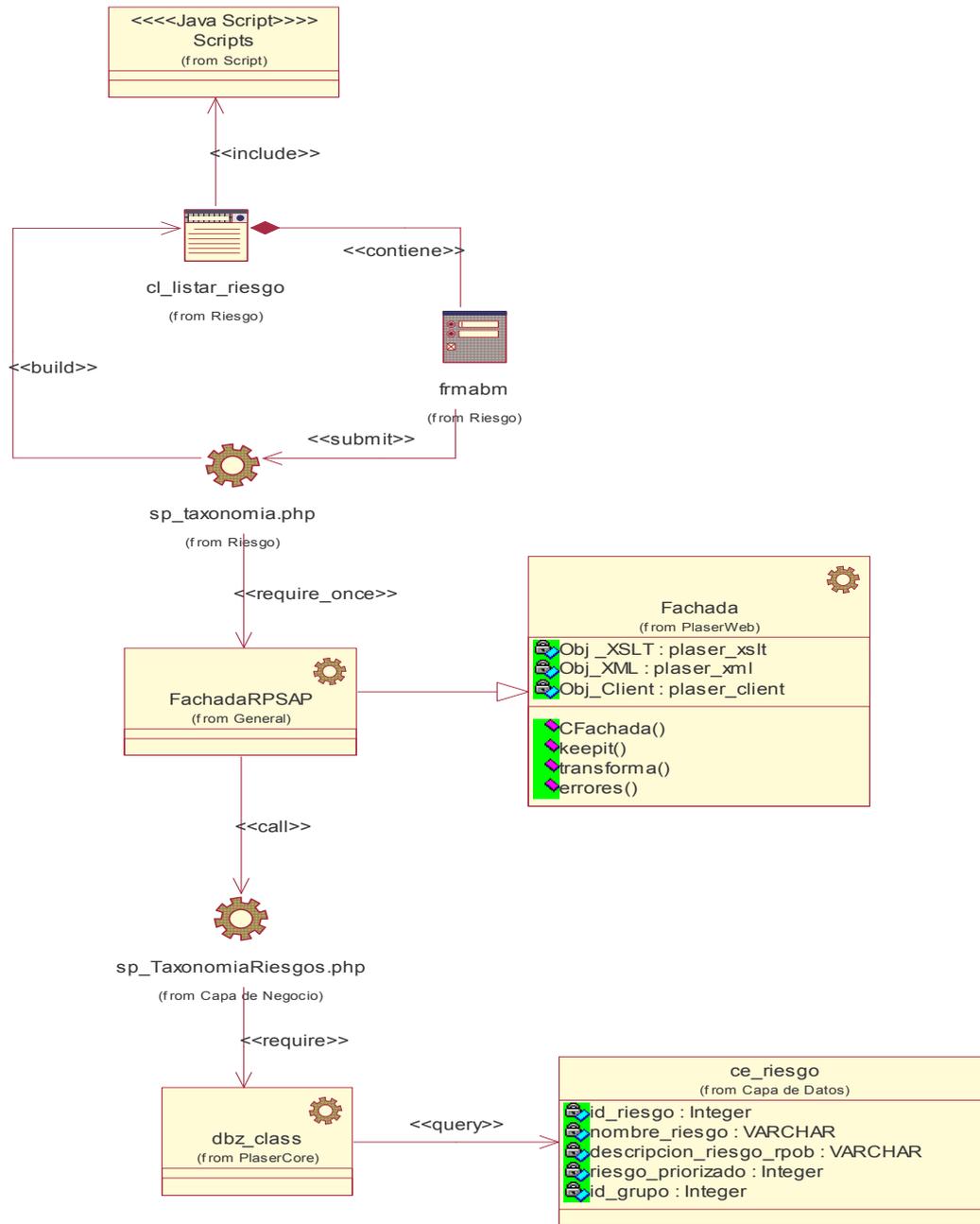


Figura 3.3 Diagrama de Clases del Diseño Listar Riesgo

# CAPÍTULO 3. DISEÑO DEL SISTEMA

## DIAGRAMA DE CLASES DEL DISEÑO LISTAR DISCAPACIDAD

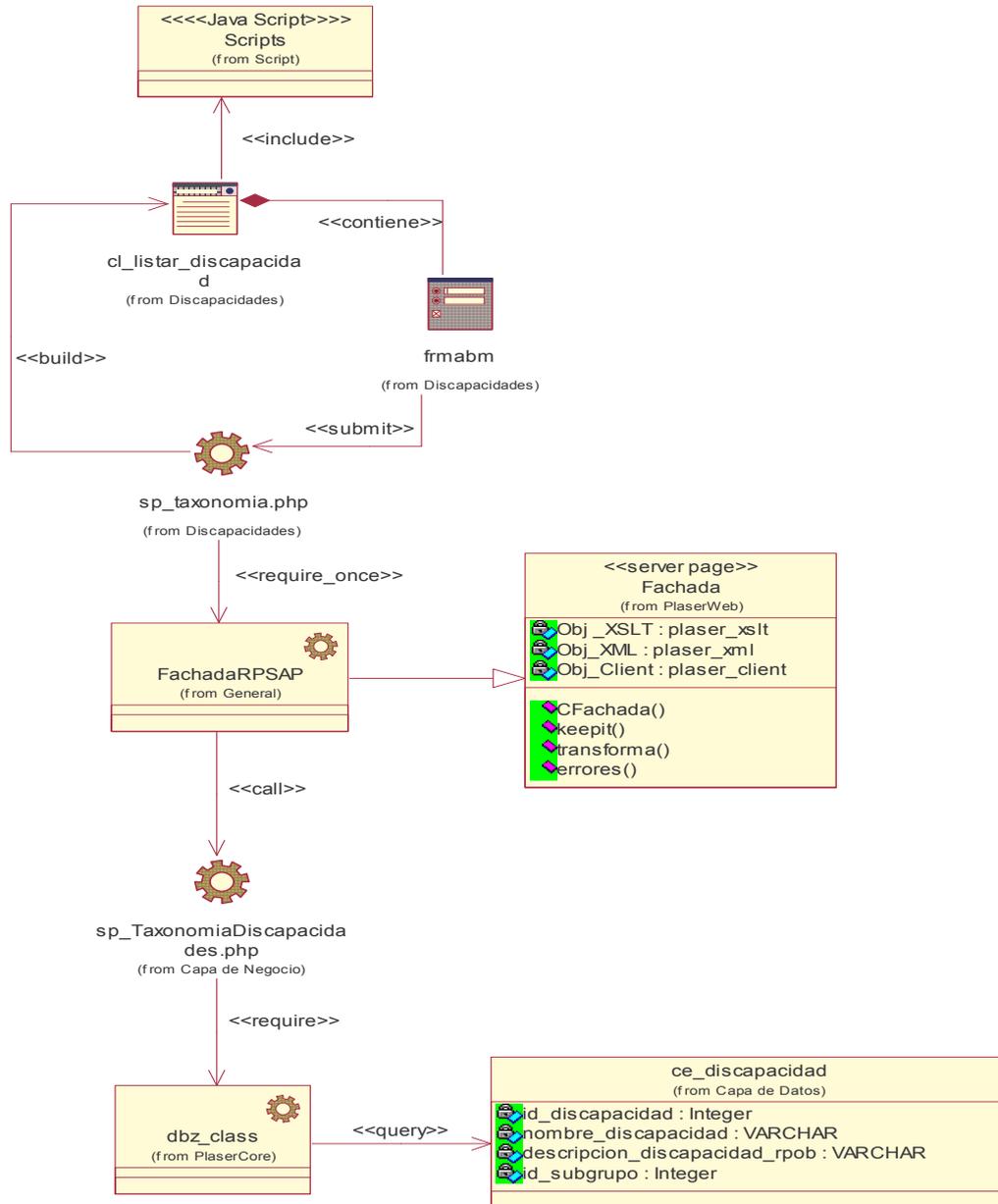


Figura 3.4 Diagrama de Clases del Diseño Listar Discapacidad

# CAPÍTULO 3. DISEÑO DEL SISTEMA

## DIAGRAMA DE CLASES DEL DISEÑO BUSCAR RIESGO

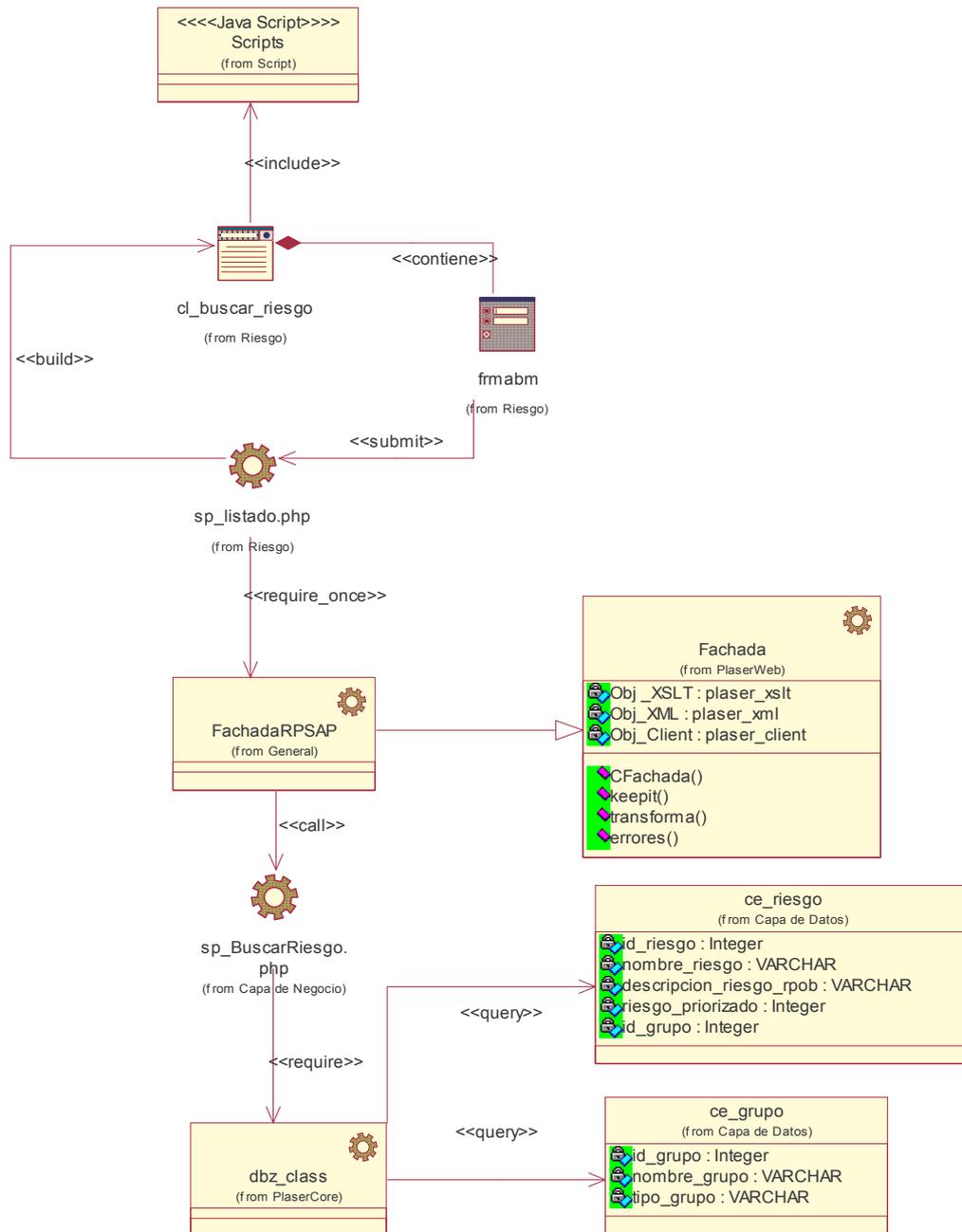


Figura 3.5 Diagrama de Clases del Diseño Buscar Riesgo

# CAPÍTULO 3. DISEÑO DEL SISTEMA

## DIAGRAMA DE CLASES DEL DISEÑO BUSCAR DISCAPACIDAD

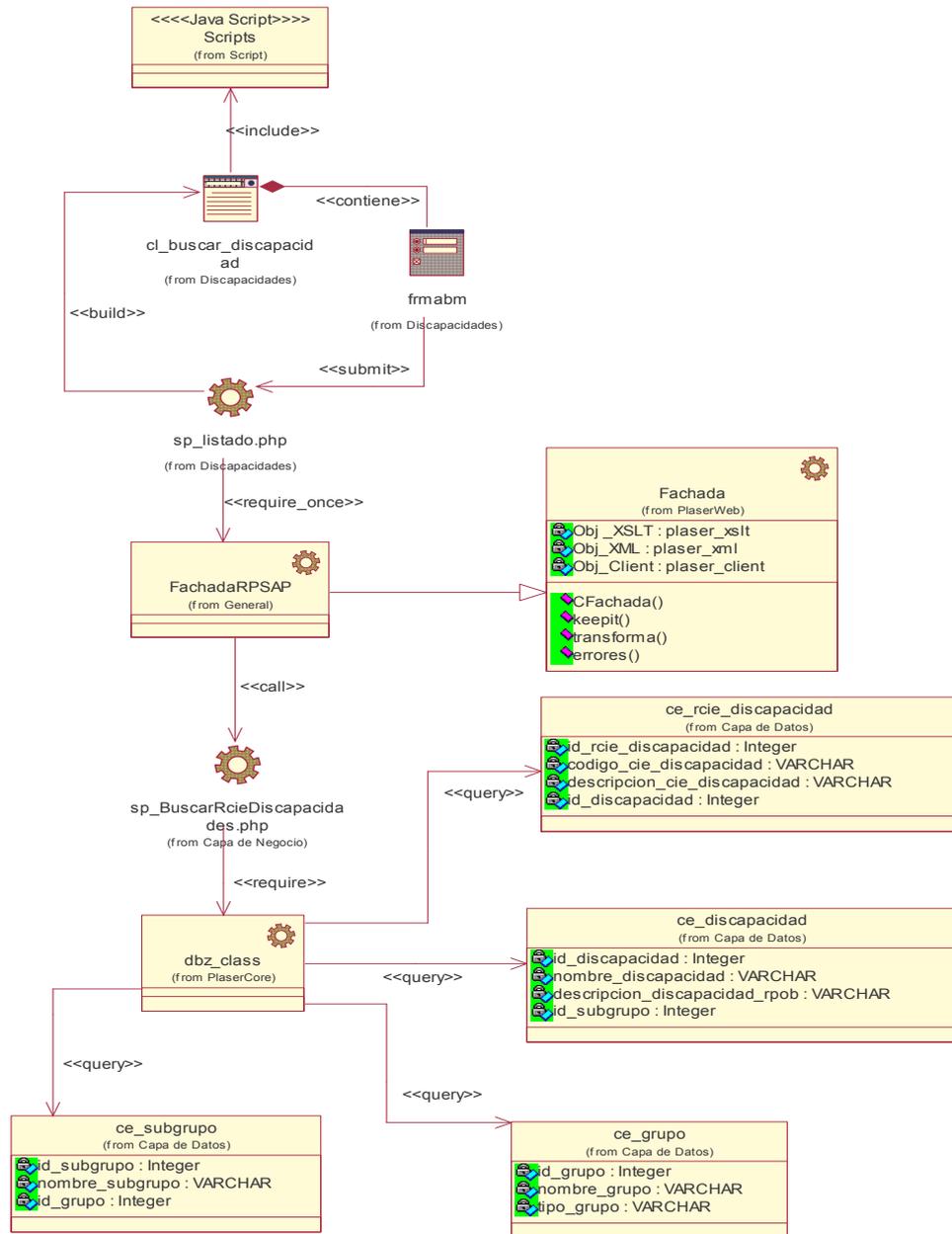


Figura 3.6 Diagrama de Clases del Diseño Buscar Discapacidad

# CAPÍTULO 3. DISEÑO DEL SISTEMA

## DIAGRAMA DE CLASES DEL DISEÑO GESTIONAR RIESGO

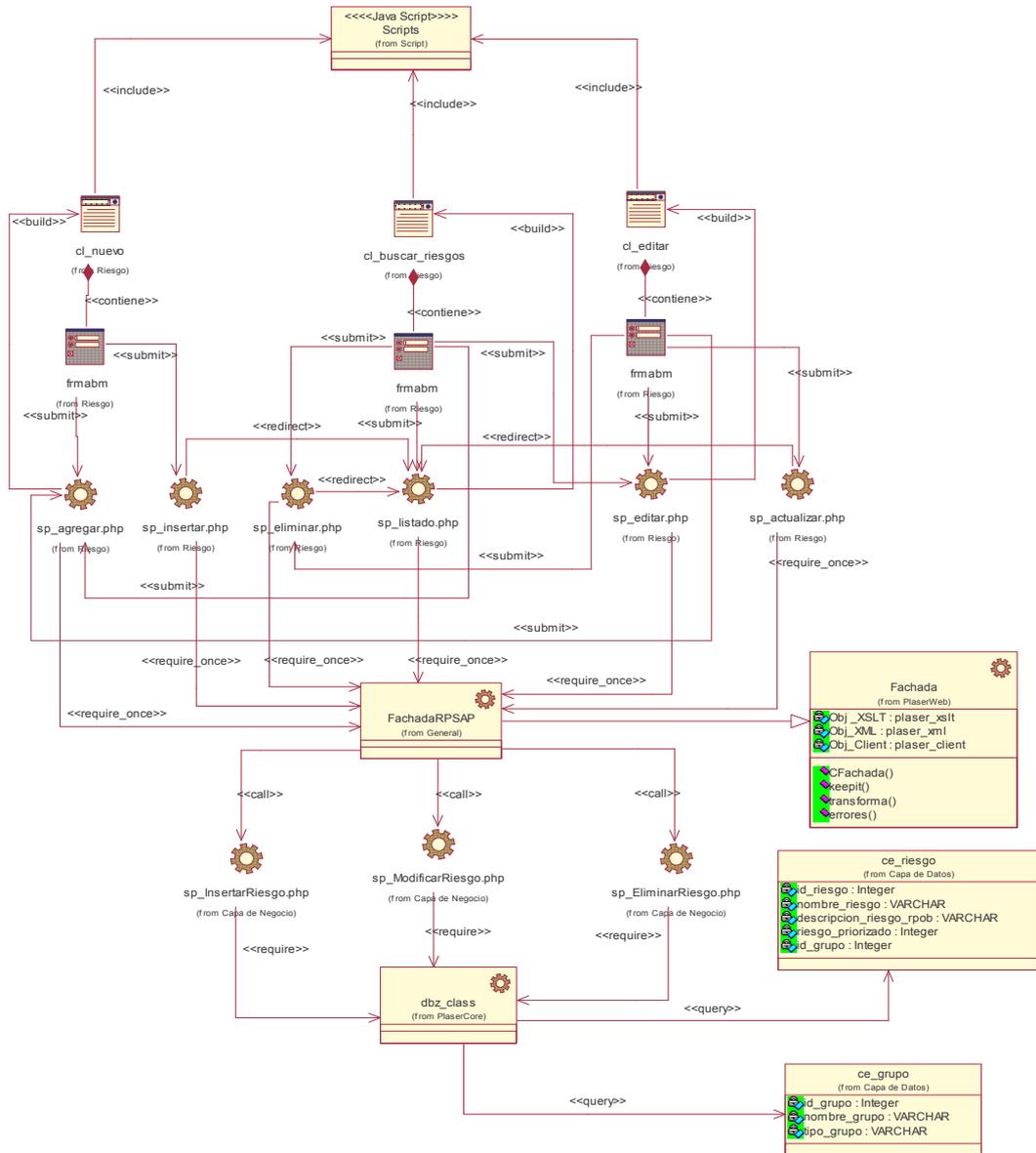


Figura 3.7 Diagrama de Clases del Diseño Gestionar Riesgo

# CAPÍTULO 3. DISEÑO DEL SISTEMA

## DIAGRAMA DE CLASES DEL DISEÑO GESTIONAR DISCAPACIDAD

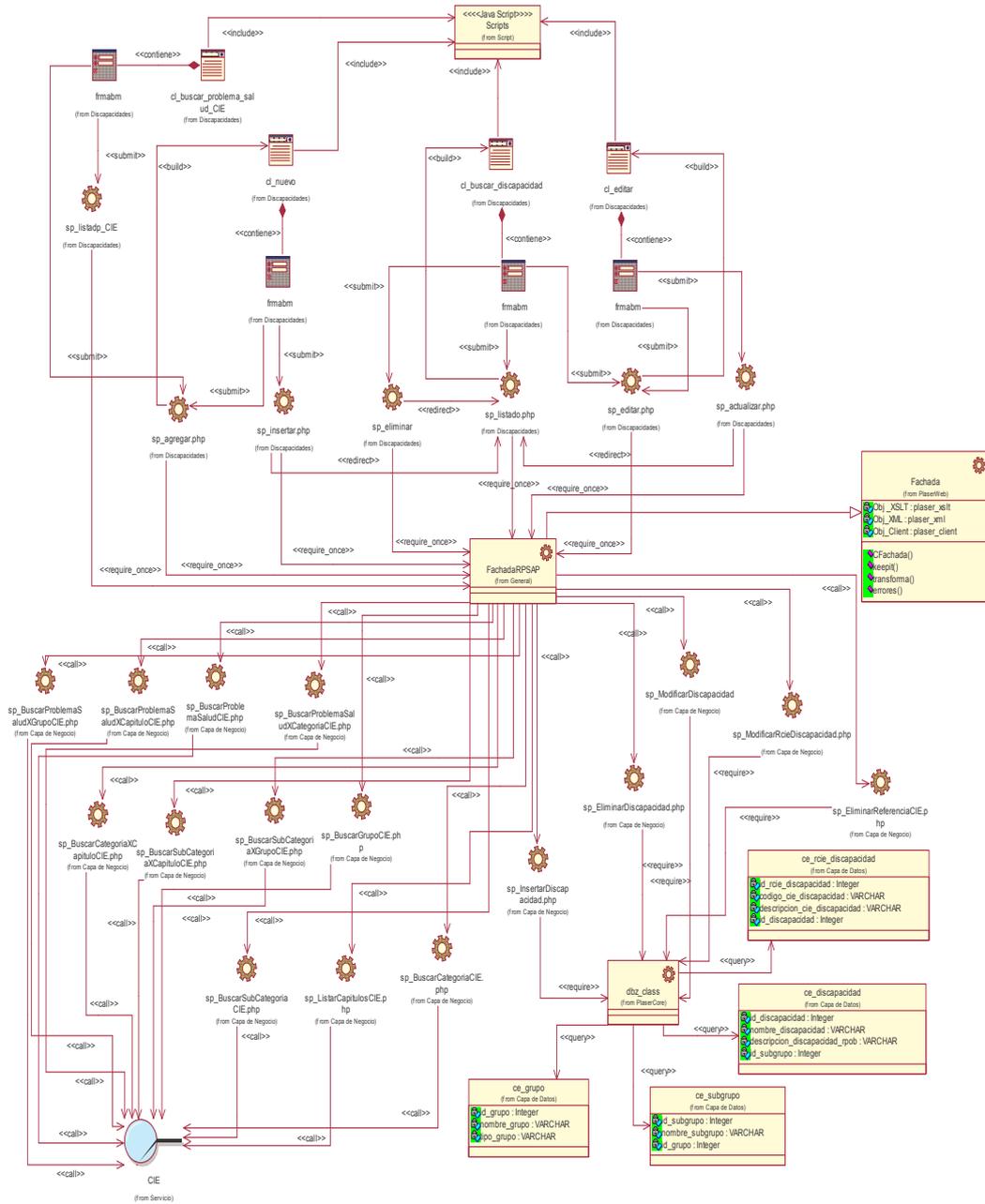


Figura 3.8 Diagrama de Clases del Diseño Gestionar Discapacidad

### 3.1.4 Descripción de las Clases y Atributos.

El lenguaje UML se utiliza para diseñar sistemas, éste tiene varios mecanismos de extensión que fueron definidos para extender su semántica en aquellas situaciones en las que no es posible capturar todas las características de una determinada arquitectura o dominio en particular.

Una de las características más relevantes de la notación UML, es su capacidad para absorber nueva semántica sin romper su lógica interna. Jim Conallen ha desarrollado desde 1998 una extensión de la notación UML denominada WAE "Web Application Extensión" que permite rentabilizar toda la gramática interna de UML para modelar aplicaciones con elementos específicos de la arquitectura de un entorno web. [67]

Después de hacer un estudio sobre los estereotipos web definidos por Jim Conallen se decidió utilizarlos para diseñar el módulo Registro de Problemas de Salud de la Atención Primaria debido a que es un mecanismo de extensibilidad más utilizado dentro de UML y representa una distinción de uso. Los mismos pueden ser aplicados a cualquier elemento de modelado, incluyendo clases, paquetes y relaciones de herencia.

En sentido general los estereotipos web más significativos así como sus relaciones se muestran a continuación:

A continuación se muestran las descripciones de las clases arquitectónicamente más importantes mostradas en los diagramas de clases del diseño, con el objetivo de dar mejor entendimiento a los mismos.

|   |
|---|
| <b>Nombre:</b> cl_nuevo   |
| <b>Tipo de clase:</b> Client Page   |
| <b>Descripción:</b> La clase cl_nuevo es una página web que se ejecuta del lado del cliente sobre un navegador web. Permite capturar los datos que serán insertados en la base de datos, presenta un conjunto de validaciones en la clase JavaScript, esto permite que no se realicen peticiones innecesarias |

## CAPÍTULO 3. DISEÑO DEL SISTEMA

y por lo tanto se incrementa su usabilidad. La misma se utiliza en los casos de uso: Gestionar Riesgo, Gestionar Enfermedad y Gestionar Discapacidad.

Tabla 3.1 Descripción de la Clase cl\_nuevo

|   |
|---|
| <b>Nombre: cl_listado</b>   |
| <b>Tipo de clase:</b> Client Page   |
| <b>Descripción:</b> La clase cl_listado es una página web que se ejecuta del lado del cliente sobre un navegador web. Permite la realización de listado y búsquedas de información permitiendo generar documentos: Portable Document Format (.pdf) y Microsoft Office Excel (.xls). A través de ella se puede visualizar la información y el resultado de las búsquedas, presenta un paginado de 10 elementos, esto garantiza la movilidad por tales resultados, inclusive ir directamente a la última página, además presenta un conjunto de validaciones en la clase JavaScript que permite que no se realicen peticiones innecesarias y que por lo tanto se incremente su usabilidad. Esta clase se utiliza en los casos de uso Gestionar Riesgo, Gestionar Enfermedad y Gestionar Discapacidad. |

Tabla 3.2 Descripción de la Clase cl\_listado

|  |
|--|
| <b>Nombre: cl_editar</b>   |
| <b>Tipo de clase:</b> Client Page  |
| <b>Descripción:</b> La clase cl_editar es una página web que se ejecuta del lado del cliente sobre un navegador web. Permite capturar los datos que serán modificados en la base de datos, presenta un conjunto de validaciones en la clase JavaScript, esto permite que no se realicen peticiones innecesarias y por lo tanto se incrementa su usabilidad. También presenta las opciones de eliminar y agregar que nos permiten ejecutar desde esta interface estas opciones. La misma se utiliza en los casos de uso: Gestionar Riesgo, Gestionar Enfermedad y Gestionar Discapacidad. |

Tabla 3.3 Descripción de la Clase cl\_editar

|   |
|---|
| <b>Nombre: sp_listado.php</b>   |
| <b>Tipo de clase:</b> Server Page   |
| <b>Descripción:</b> La clase sp_listado.php se ejecuta del lado del servidor en la Capa de Presentación. Su |

## CAPÍTULO 3. DISEÑO DEL SISTEMA

función es construir una página cliente que está asociada a ella. La misma busca los datos que se desean listar y luego transforma el XML que llega de la petición realizada a formato XHTML y se lo muestra al usuario. Esta clase se utiliza en los casos de uso Gestionar Riesgo, Gestionar Enfermedad y Gestionar Discapacidad.

Tabla 3.4 Descripción de la Clase sp\_listado.php

|   |
|---|
| <b>Nombre: sp_eliminar.php</b>  |
| <b>Tipo de clase:</b> Server Page   |
| <b>Descripción:</b> La clase sp_eliminar.php se ejecuta del lado del servidor en la Capa de Presentación. Su función es eliminar información en la base de datos después de realizada esta petición, luego se redirecciona para la clase servidora sp_listado.php. Esta clase se utiliza en los casos de uso Gestionar Riesgo, Gestionar Enfermedad y Gestionar Discapacidad. |

Tabla 3.5 Descripción de la Clase sp\_listado.php

|   |
|---|
| <b>Nombre: sp_actualizar.php</b>  |
| <b>Tipo de clase:</b> Server Page   |
| <b>Descripción:</b> La clase sp_actualizar.php se ejecuta del lado del servidor en la Capa de Presentación. Su función es actualizar información en la base de datos después de realizada esta petición, luego se redirecciona para la clase servidora sp_listado.php. Esta clase se utiliza en los casos de uso Gestionar Riesgo, Gestionar Enfermedad y Gestionar Discapacidad. |

Tabla 3.6 Descripción de la Clase sp\_actualizar.php

|   |
|---|
| <b>Nombre: sp_agregar.php</b>   |
| <b>Tipo de clase:</b> Server Page   |
| <b>Descripción:</b> La clase sp_agregar.php se ejecuta del lado del servidor en la Capa de Presentación. Su función es construir la página cliente cl_nuevo. Esta clase se utiliza en los casos de uso Gestionar Riesgo, Gestionar Enfermedad y Gestionar Discapacidad. |

Tabla 3.7 Descripción de la Clase sp\_agregar.php

## CAPÍTULO 3. DISEÑO DEL SISTEMA

|   |
|---|
| <b>Nombre: sp_editar.php</b>  |
| <b>Tipo de clase:</b> Server Page   |
| <b>Descripción:</b> La clase sp_editar.php se ejecuta del lado del servidor en la Capa de Presentación. Su función es construir la página cliente cl_editar. Esta clase servidora se encarga de buscar los datos que están registrados hasta ese momento en la base de datos, luego transforma el XML que llega de la petición realizada a formato XHTML y se lo muestra al usuario. La misma se utiliza en los casos de uso Gestionar Riesgo, Gestionar Enfermedad y Gestionar Discapacidad. |

Tabla 3.8 Descripción de la Clase sp\_editar.php

|   |
|---|
| <b>Nombre: sp_insertar.php</b>  |
| <b>Tipo de clase:</b> Server Page   |
| <b>Descripción:</b> La clase sp_insertar.php se ejecuta del lado del servidor en la Capa de Presentación. Su función es insertar información en la base de datos después de realizada esta petición, luego se redirecciona para la clase servidora sp_listado.php. Esta clase se utiliza en los casos de uso Gestionar Riesgo, Gestionar Enfermedad y Gestionar Discapacidad. |

Tabla 3.9 Descripción de la Clase sp\_insertar.php

|   |
|---|
| <b>Nombre: sp_taxonomia.php</b>   |
| <b>Tipo de clase:</b> Server Page   |
| <b>Descripción:</b> La clase sp_taxonomia.php se ejecuta del lado del servidor en la Capa de Presentación. Su función es construir las páginas clientes cl_listar_riesgo y cl_listar_discapacidades. Esta clase permite buscar los datos que están registrados hasta ese momento en la base de datos, luego transforma el XML que llega de la petición realizada a formato XHTML y se lo muestra al usuario. La misma se utiliza en los casos de uso Listar Riesgos, Listar Enfermedades y Listar Discapacidades. |

Tabla 3.10 Descripción de la Clase sp\_taxonomia.php

|                                   |
|-----------------------------------|
| <b>Nombre: CFachada</b>           |
| <b>Tipo de clase:</b> Server Page |

## CAPÍTULO 3. DISEÑO DEL SISTEMA

**Descripción:** La clase Fachada es una clase que pertenece a la librería de clases de PlaSer. Su función principal es modelar la comunicación entre la capa de presentación y PlaSer, es decir representa la fachada de esta librería de clase, lo que nos permite utilizar las funcionalidades que tiene la misma si acceder directamente a sus otras clases, lo que facilita la portabilidad y escabilidad del sistema. Esta clase se encuentra en todos los casos de uso del módulo.

Tabla 3.11 Descripción de la Clase CFachada

**Nombre:** FachadaRPSAP

**Tipo de clase:** Server Page

**Descripción:** La clase FachadaRPSAP es una clase que representa la fachada de la capa de presentación. La misma hereda de la clase Fachada de PlaSer y función principal es modelar la comunicación con la capa de negocio, por lo que se encarga de realizar las peticiones a los métodos del negocio. Esta clase se encuentra en todos los casos de uso del módulo.

Tabla 3.12 Descripción de la Clase FachadaRPSAP

**Nombre:** sp\_BuscarRiesgo.php

**Tipo de clase:** Server Page

**Descripción:** La clase sp\_BuscarRiesgo.php es una clase que se ejecuta del lado del servidor en la Capa de Negocio. Su función es interactuar con la capa de datos y devolver a la capa de presentación un listado de los riesgos según los criterios de búsqueda insertados.

**Parámetros de entrada:** nombre\_riesgo, id\_riesgo, id\_grupo, ordenar\_por, orden, offset y cantidad.

Tabla 3.13 Descripción de la Clase sp\_BuscarRiesgo.php

**Nombre:** sp\_BuscarRcieDiscapacidades.php

**Tipo de clase:** Server Page

**Descripción:** La clase sp\_BuscarRcieDiscapacidades.php es una clase que se ejecuta del lado del servidor en la Capa de Negocio. Su función es interactuar con la capa de datos y devolver a la capa de presentación un listado de las discapacidades definidas en la atención primaria y su referencia con la

## CAPÍTULO 3. DISEÑO DEL SISTEMA

|  |
|--|
| CIE-10, según los criterios de búsqueda insertados.  |
| <b>Parámetros de entrada:</b> nombre_discapacidad, id_discapacidad, nombre_subgrupo, id_grupo, nombre_grupo, ordenar_por, ordena, offset y cantidad. |

Tabla 3.14 Descripción de la Clase sp\_BuscarRcieDiscapacidades.php

|   |
|---|
| <b>Nombre:</b> sp_EliminarRiesgo.php  |
| <b>Tipo de clase:</b> Server Page   |
| <b>Descripción:</b> La clase sp_EliminarRiesgo.php es una clase que se ejecuta del lado del servidor en la Capa de Negocio. Su función es interactuar con la capa de datos y eliminar de la base de datos la información correspondiente a la petición realizada desde la capa de presentación. |
| <b>Parámetros de entrada:</b> id_riesgo.  |

Tabla 3.15 Descripción de la Clase sp\_EliminarRiesgo.php

|   |
|---|
| <b>Nombre:</b> sp_EliminarDiscapacidad.php  |
| <b>Tipo de clase:</b> Server Page   |
| <b>Descripción:</b> La clase sp_EliminarDiscapacidad.php es una clase que se ejecuta del lado del servidor en la Capa de Negocio. Su función es interactuar con la capa de datos y eliminar de la base de datos la información correspondiente a la petición realizada desde la capa de presentación. |
| <b>Parámetros de entrada:</b> nombre_discapacidad.  |

Tabla 3.16 Descripción de la Clase sp\_EliminarDiscapacidad.php

|   |
|---|
| <b>Nombre:</b> sp_InsertarRiesgo.php  |
| <b>Tipo de clase:</b> Server Page   |
| <b>Descripción:</b> La clase sp_InsertarRiesgo.php es una clase que se ejecuta del lado del servidor en la Capa de Negocio. Su función es interactuar con la capa de datos e insertar en la base de datos la información correspondiente a la petición realizada desde la capa de presentación. |
| <b>Parámetros de entrada:</b> nombre_riesgo, id_grupo, id_priorizado y descripcion_riesgo.  |

Tabla 3.17 Descripción de la Clase sp\_InsertarRiesgo.php

## CAPÍTULO 3. DISEÑO DEL SISTEMA

|   |
|---|
| <b>Nombre:</b> sp_InsertarDiscapacidad.php  |
| <b>Tipo de clase:</b> Server Page   |
| <b>Descripción:</b> La clase sp_InsertarRiesgo.php es una clase que se ejecuta del lado del servidor en la Capa de Negocio. Su función es interactuar con la capa de datos e insertar en la base de datos la información correspondiente a la petición realizada desde la capa de presentación. |
| <b>Parámetros de entrada:</b> nombre_discapacidad, descripcion_discapacidad, id_subgrupo, array_codigo_CIE y array_descripcion_CIE.   |

Tabla 3.18 Descripción de la Clase sp\_InsertarDiscapacidad.php

|   |
|---|
| <b>Nombre:</b> sp_ModificarRiesgo.php   |
| <b>Tipo de clase:</b> Server Page   |
| <b>Descripción:</b> La clase sp_ModificarRiesgo.php es una clase que se ejecuta del lado del servidor en la Capa de Negocio. Su función es interactuar con la capa de datos y modificar en la base de datos la información correspondiente a la petición realizada desde la capa de presentación. |
| <b>Parámetros de entrada:</b> id_riesgo, nombre_riesgo, descripcion_riesgo, id_grupo e id_priorizado.   |

Tabla 3.19 Descripción de la Clase sp\_ModificarRiesgo.php

|   |
|---|
| <b>Nombre:</b> sp_ModificarDiscapacidad.php   |
| <b>Tipo de clase:</b> Server Page   |
| <b>Descripción:</b> La clase sp_ModificarDiscapacidad.php es una clase que se ejecuta del lado del servidor en la Capa de Negocio. Su función es interactuar con la capa de datos y modificar en la base de datos la información correspondiente a la petición realizada desde la capa de presentación. |
| <b>Parámetros de entrada:</b> id_discapacidad, nombre_discapacidad, descripcion_discapacidad, id_grupo e id_subgrupo.   |

Tabla 3.20 Descripción de la Clase sp\_ModificarDiscapacidad.php

### 3.1.5 Diagrama de Clases Persistentes.

Los almacenes de datos son el centro de atención para las grandes empresas de hoy en día, ya que constituyen uno de los soportes fundamentales para el proceso de toma de decisiones gerenciales; de ahí la importancia de que la información guardada en ellos sea confiable y con calidad. Por tal razón, las bases de datos han cobrado gran auge en la actualidad.

Para llegar a obtener una base de datos se necesita pasar por varios niveles o pasos. Uno de los procesos que se realizan para obtenerlas usando la metodología RUP es la realización del diagrama de clases persistentes, elemento fundamental en la construcción del modelo de datos que tiene como propósito simplificar el diseño de bases de datos a partir de descripciones textuales de los requerimientos.

Para realizar el Diagrama de Clases Persistentes se necesita definir cuales son las clases persistentes, siendo la persistencia la capacidad de un objeto de mantener su valor en el espacio y en el tiempo. Este diagrama describe la representación lógica de los datos persistentes, de acuerdo con el enfoque para modelado relacional de datos. Para expresar este diagrama se utiliza un Diagrama de Clases, donde se utiliza UML, para conseguir la representación de tablas, claves, etc. [68] A continuación se representa este diagrama:

## CAPÍTULO 3. DISEÑO DEL SISTEMA

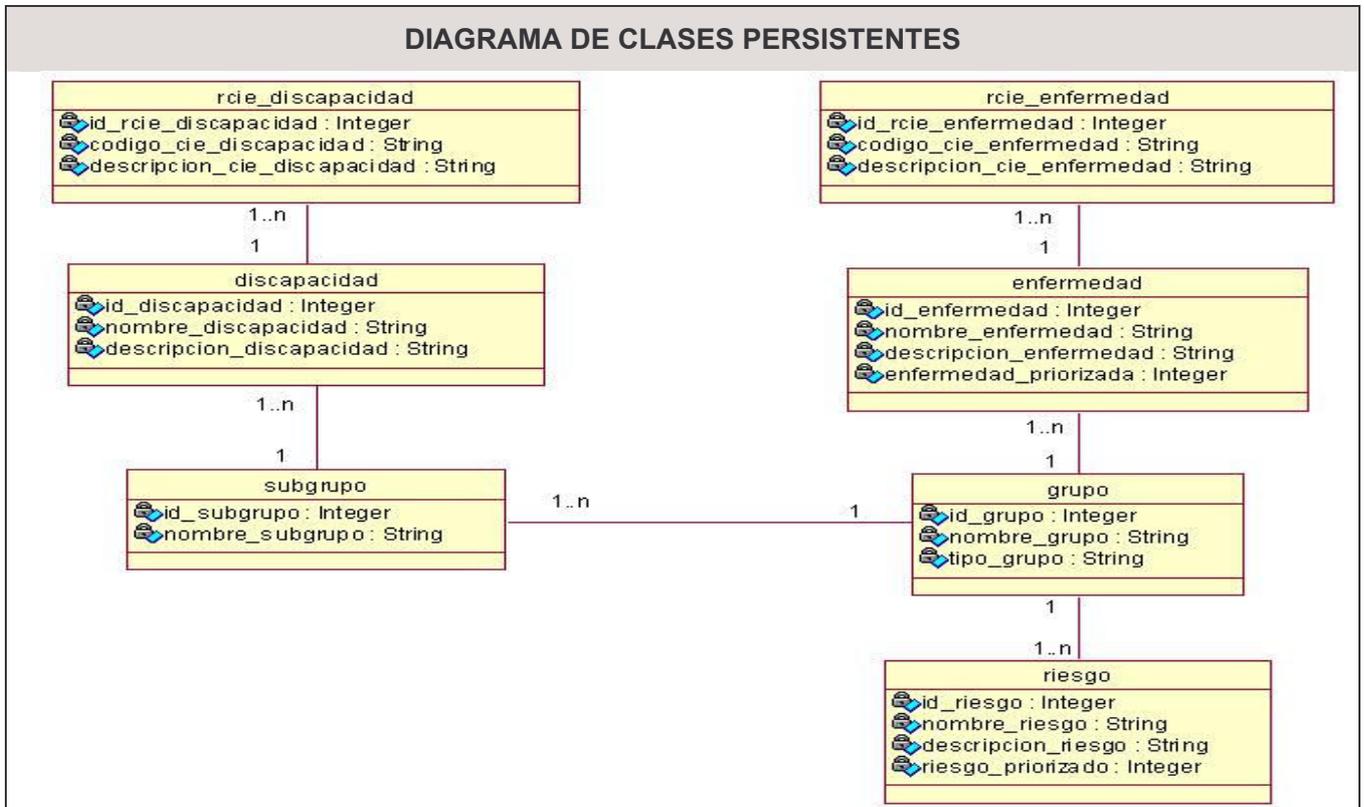


Tabla 3.21 Diagrama de Clases Persistentes

### 3.5 Modelo de Datos.

Cuando se utiliza una base de datos para gestionar información, se está plasmando una parte del mundo real en una serie de tablas, registros y campos ubicados en un ordenador; creándose un modelo parcial de la realidad. Antes de crear físicamente estas tablas en el ordenador se debe realizar un modelo de datos.

El modelo de datos es el artefacto más importante concebido por RUP dentro del proceso que incluye generar la base de datos por esta metodología. Este modelo nos proporciona una representación visual y física de los datos persistentes del sistema, que en el futuro serán la base de datos. Se obtiene a partir del diagrama de clases persistentes y su forma se expresa mediante un diagrama de UML, siendo sus elementos esenciales del modelo son las entidades, los atributos y las relaciones entre las entidades. A continuación se muestra el modelo de datos del sistema.

# CAPÍTULO 3. DISEÑO DEL SISTEMA

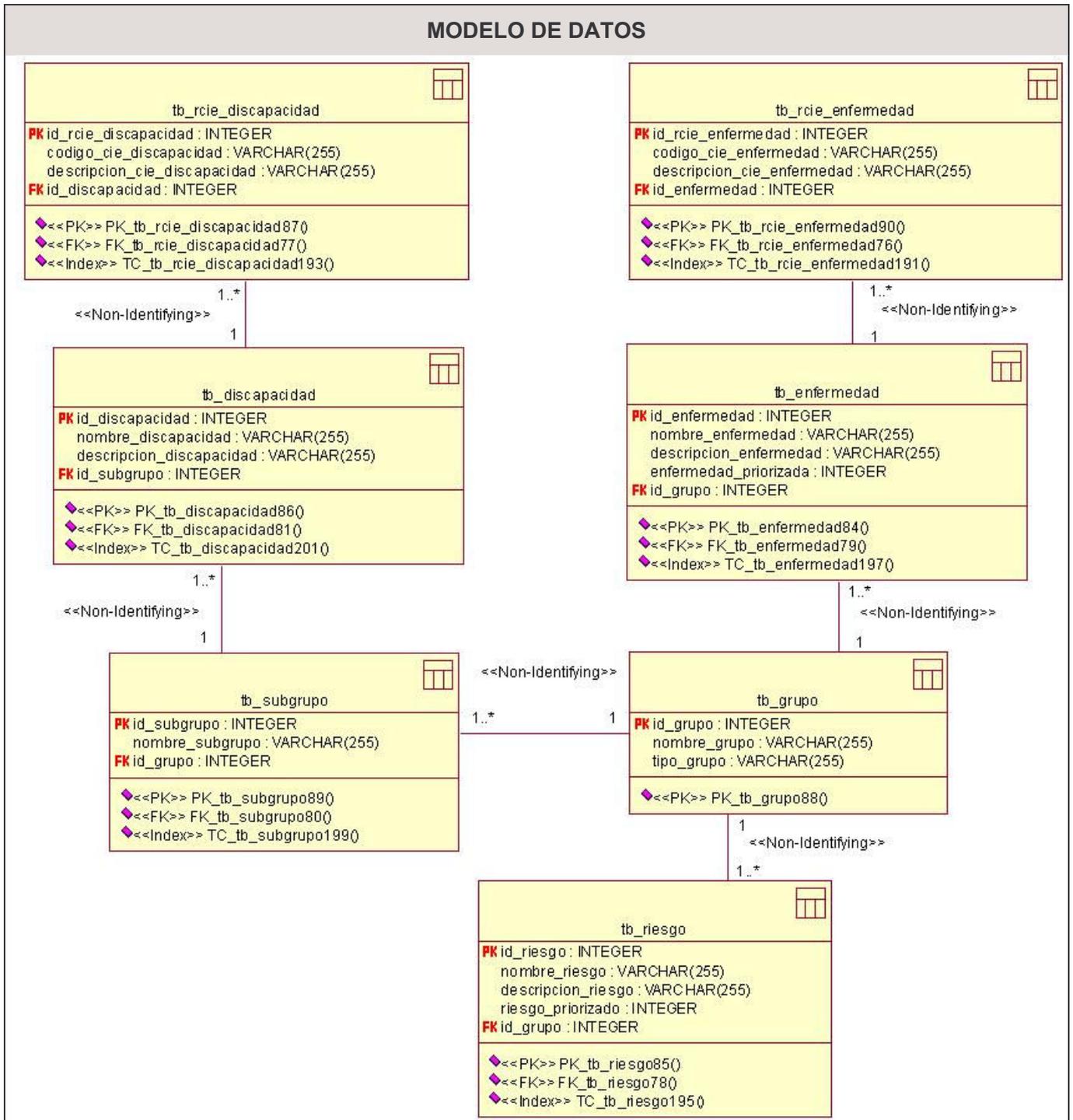


Tabla 3.22 Modelo de Datos

## CAPÍTULO 3. DISEÑO DEL SISTEMA

### 3.6 Descripción de tablas y atributos.

Las descripciones de tablas y atributos son un elemento importante que complementa la descripción gráfica que genera el diagrama de clases persistentes y el modelo de dato. Su objetivo principal es hacer comprender con claridad los diagramas de clases concebidos para generar la base de datos, sirviendo de soporte a los mismos. En este artefacto se podrá encontrar información más detallada sobre los atributos y clases que conformarán la base de datos. Estas descripciones se muestran a continuación:

| <b>Nombre: tb_riesgo</b>   |              |  |
|--|--------------|--|
| <b>Descripción:</b> Esta tabla guarda los datos de todos los riesgos definidos en la Atención Primaria. Sirve como contenedor de información al que se puede encuestar y realizar las operaciones de modificar, eliminar e insertar riesgos. |              |  |
| <b>Atributo</b>  | <b>Tipo</b>  | <b>Descripción</b>   |
| id_riesgo  | INTEGER      | En este atributo se almacena el identificador del riesgo definido en la Atención Primaria. Es un autoincremento y la llave primaria de la tabla tb_riesgo. |
| nombre_riesgo  | VARCHAR(255) | Este atributo contiene el nombre del riesgo definido en la Atención Primaria.  |
| descripcion_riesgo   | VARCHAR(255) | Este atributo contiene la descripción del riesgo definido en la Atención Primaria.   |
| riesgo_priorizado  | INTEGER      | Este atributo contiene un número que representa si es priorizado el riesgo o no. Toma valor 1 si es priorizado y 2 si no es priorizado.                    |

## CAPÍTULO 3. DISEÑO DEL SISTEMA

|            |         |  |
|------------|---------|--|
| id _ grupo | INTEGER | Este atributo contiene el identificador del grupo al que pertenece dicho riesgo, por lo que es una llave foránea de la tabla tb_grupo. |
|------------|---------|--|

Tabla 3.23 Descripción de la Tabla tb\_riesgo

| Nombre: tb_enfermedad   |              |  |
|---|--------------|--|
| <p><b>Descripción:</b> Esta tabla guarda los datos de todas las enfermedades definidas en la Atención Primaria. Sirve como contenedor de información al que se puede encuestar y realizar las operaciones de modificar, eliminar e insertar enfermedades.</p> |              |  |
| Atributo  | Tipo         | Descripción  |
| id_enfermedad   | INTEGER      | En este atributo se almacena el identificador de la enfermedad definida en la Atención Primaria. Es un autoincremento y la llave primaria de la tabla tb_enfermedad. |
| nombre_enfermedad   | VARCHAR(255) | Este atributo contiene el nombre de la enfermedad definida en la Atención Primaria.  |
| descripcion_enfermedad  | VARCHAR(255) | Este atributo contiene la descripción de una enfermedad definida en la Atención Primaria.  |
| enfermedad_priorizada   |              | Este atributo contiene un número que representa si es priorizada la enfermedad o no. Toma valor 1 si es priorizada y 2 si no es priorizada.                          |
| id _ grupo  | INTEGER      | Este atributo contiene el identificador del grupo al que pertenece dicha enfermedad, por lo que es una llave foránea de la tabla tb_grupo.                           |

Tabla 3.24 Descripción de la Tabla tb\_enfermedad

## CAPÍTULO 3. DISEÑO DEL SISTEMA

| <b>Nombre: tb_rcie_enfermedad</b>  |              |  |
|--|--------------|--|
| <b>Descripción:</b> Esta tabla guarda una parte de la información de las enfermedades de CIE-10, que puedan tener referencia con la enfermedad analizada en la atención primaria, estableciéndose la relación en dicha tabla. Sirve como contenedor de información al que se puede encuestar y realizar las operaciones de modificar, eliminar e insertar las enfermedades de CIE-10, que no son más que las referencias de las enfermedades de la atención primaria; siempre teniendo en cuenta que debe quedar al menos una referencia CIE-10 en esta tabla por cada enfermedad de la atención primaria. |              |  |
| Atributo   | Tipo         | Descripción  |
| id_rcie_enfermedad   | INTEGER      | En este atributo se almacena el identificador de la tabla tb_rcie_enfermedad, que es autoincremento y constituye su llave primaria.  |
| codigo_cie_enfermedad  | VARCHAR(10)  | Este atributo es el código de la enfermedad que se importa del CIE-10.   |
| descripcion_cie_enfermedad   | VARCHAR(255) | Este atributo es la descripción de la enfermedad que se importa del CIE-10.  |
| id_enfermedad  | INTEGER      | Este atributo contiene el identificador de la enfermedad definida en la atención primaria, por lo que es una llave foránea de la tabla tb_enfermedad. Nos permite mantener la relación las enfermedades de la atención primaria con la CIE-10. |

Tabla 3.25 Descripción de la Tabla tb\_rcie\_enfermedad

| <b>Nombre: tb_discapacidad</b>   |      |             |
|--|------|-------------|
| <b>Descripción:</b> Esta tabla guarda los datos de todas las discapacidades definidas en la Atención Primaria. Sirve como contenedor de información al que se puede encuestar y realizar las operaciones de modificar, eliminar e insertar discapacidades. |      |             |
| Atributo   | Tipo | Descripción |

## CAPÍTULO 3. DISEÑO DEL SISTEMA

|                          |              |  |
|--------------------------|--------------|--|
| id_discapacidad          | INTEGER      | En este atributo se almacena el identificador de la discapacidad definida en la Atención Primaria. Es un autoincremento y la llave primaria de la tabla tb_discapacidad. |
| nombre_discapacidad      | VARCHAR(255) | Este atributo contiene el nombre de la discapacidad definida en la Atención Primaria.  |
| descripcion_discapacidad | VARCHAR(255) | Este atributo contiene la descripción de una discapacidad definida en la Atención Primaria.  |
| id_subgrupo              | INTEGER      | Este atributo contiene el identificador del subgrupo al que pertenece dicha discapacidad, por lo que es una llave foránea de la tabla tb_subgrupo.                       |

Tabla 3.26 Descripción de la Tabla tb\_discapacidad

|  |             |   |
|--|-------------|---|
| <b>Nombre: tb_rcie_discapacidad</b>  |             |   |
| <b>Descripción:</b> Esta tabla guarda una parte de la información de las discapacidades de CIE-10, que puedan tener referencia con la discapacidad analizada en la atención primaria, estableciéndose la relación en dicha tabla. Sirve como contenedor de información al que se puede encuestar y realizar las operaciones de modificar, eliminar e insertar las discapacidades de CIE-10, que no son más que las referencias de las discapacidades de la atención primaria; siempre teniendo en cuenta que debe quedar al menos una referencia CIE-10 en esta tabla por cada discapacidad de la atención primaria. |             |   |
| <b>Atributo</b>  | <b>Tipo</b> | <b>Descripción</b>  |
| id_rcie_discapacidad   | INTEGER     | En este atributo se almacena el identificador de la tabla tb_rcie_discapacidad, que es autoincremento y constituye su llave primaria. |
| codigo_cie_discapacidad  | VARCHAR(10) | Este atributo es el código de la discapacidad que se importa del CIE-10.  |

## CAPÍTULO 3. DISEÑO DEL SISTEMA

|                              |              |  |
|------------------------------|--------------|--|
| descripcion_cie_discapacidad | VARCHAR(255) | Este atributo es la descripción de la discapacidad que se importa del CIE-10.  |
| id_discapacidad              | INTEGER      | Este atributo contiene el identificador de la discapacidad definida en la atención primaria, por lo que es una llave foránea de la tabla tb_discapacidad. Nos permite mantener la relación las enfermedades de la atención primaria con la CIE-10. |

Tabla 3.27 Descripción de la Tabla tb\_rcie\_discapacidad

| <b>Nombre: tb_grupo</b>   |              |   |
|---|--------------|---|
| <b>Descripción:</b> Esta tabla guarda los datos de todos los grupos definidos en la Atención Primaria. Sirve como contenedor de información al que se puede encuestar y realizar las operaciones de modificar, eliminar e insertar grupo. |              |   |
| Atributo  | Tipo         | Descripción   |
| id_grupo  | INTEGER      | En este atributo se almacena el identificador del grupo definido en la Atención Primaria. Es un autoincremento y la llave primaria de la tabla tb_grupo.  |
| nombre_grupo  | VARCHAR(255) | Este atributo contiene el nombre del grupo definido en la Atención Primaria.  |
| tipo_grupo  | VARCHAR(255) | Este atributo contiene el tipo de grupo al que pertenece el grupo definido en la Atención Primaria. El grupo puede ser Riesgo, Enfermedad o Discapacidad. |

Tabla 3.28 Descripción de la Tabla tb\_grupo

## CAPÍTULO 3. DISEÑO DEL SISTEMA

| Nombre: tb_subgrupo   |              |  |
|---|--------------|--|
| <b>Descripción:</b> Esta tabla guarda los datos de todos los subgrupos definidos en la Atención Primaria. Sirve como contenedor de información al que se puede encuestar y realizar las operaciones de modificar, eliminar e insertar subgrupo. |              |  |
| Atributo  | Tipo         | Descripción  |
| id_subgrupo   | INTEGER      | En este atributo se almacena el identificador del subgrupo definido en la Atención Primaria. Es un autoincremento y la llave primaria de la tabla tb_subgrupo. |
| nombre_subgrupo   | VARCHAR(255) | Este atributo contiene el nombre del subgrupo definido en la Atención Primaria.  |
| id_grupo  | INTEGER      | Este atributo contiene el identificador del grupo al que pertenece dicho subgrupo, por lo que es una llave foránea de la tabla tb_grupo.                       |

Tabla 3.29 Descripción de la Tabla tb\_subgrupo

### Conclusiones.

En este capítulo se han obtenido diferentes artefactos de diseño, se explicaron los patrones, la estructura, los diagramas de clases del diseño, el diagrama de clases persistentes y el modelo de datos. También se hace una descripción de cada una de las clases, tablas y atributos con el objetivo de complementar y entender mejor el diseño del sistema. Toda la información analizada, ha aportado una concepción completa y concisa del Modelo del Diseño.

# Capítulo 4. Implementación

## Introducción.

En diversas ocasiones el desarrollo de los proyectos de software causan confusiones y malas interpretaciones en los clientes, usuarios, especialistas y técnicos de la informática, debido a que sus funcionalidades se complejizan con la existencia de disímiles notaciones y metodologías para su implementación. En la construcción de aplicaciones informáticas se utiliza mucho el concepto de “divide y vencerás”. Se trata de dividir la elaboración de un producto en diferentes modelos y fases con el fin de dirigir de forma eficiente este proceso, así como de obtener un software que satisfaga las necesidades del cliente.

Uno de los tantos flujos, que define RUP como metodología de desarrollo, es el de implementación. En él se materializa la creación del analista, se refina la arquitectura y se obtiene un sistema ejecutable. Sus propósitos son: definir la organización del código, implementar las clases y objetos en forma de componentes, probar los componentes desarrollados e integrar las componentes en un sistema ejecutable. Su principal resultado es el modelo de implementación.

Este capítulo, se dedicará a la implementación, se hará una breve explicación sobre la integración de este módulo con otros sistemas, una descripción de los métodos y agentes más importantes, los estándares de diseño, codificación y tratamiento de excepciones. También se definirán y se representarán los diagramas de despliegue y de componentes, éste último de forma general.

## 4.1 Integración con Otros Sistemas.

El Módulo Registro de Problemas de Salud de la Atención Primaria para su buen funcionamiento y desempeño está integrado con dos módulos pertenecientes al SISalud, el Sistema de Autenticación, Autorización y Auditoría (SAAA) y el Registro de la Clasificación Internacional de Enfermedades y Problemas Relacionados con la Salud (RCIE). El módulo SAAA le brinda la seguridad al módulo, permitiendo que cada usuario acceda sólo a la información que le corresponde, facilitando la consistencia y seguridad de la información. Del módulo RCIE se consumen servicios web, para tener la información de

## CAPÍTULO 4. IMPLEMENTACIÓN

---

los problemas de salud de la Atención Primaria de Cuba vinculados a la información de la CIE-10, utilizada ésta última por la mayoría de las organizaciones e instituciones de salud en el mundo y establecida como un estándar a nivel mundial, instaurando de esta forma un lenguaje común a nivel internacional y facilitando la realización de los reportes estadísticos a instituciones mundiales.

El mayor aporte práctico de la aplicación es brindar una estructura adecuada de los problemas de salud en riesgos, enfermedades y discapacidades para la dispensarización y diagnóstico de los pacientes. Actualmente el módulo de Población consume los servicios web ofrecidos por el RPSAP de forma satisfactoria, facilitando la dispensarización y diagnóstico de los pacientes, así como el trabajo del médico. Otro módulo que utilizará los servicios web del RPSAP es el Registro de Actividades Diarias, para realizar el diagnóstico de los pacientes, fortaleciendo el trabajo del médico, así como dándole eficiencia a este proceso tan importante.

Además de estos dos módulos del SISalud ya definidos, podrán utilizar los servicios web del RPSAP todos los sistemas relacionados con el entorno de la Atención Primaria, que necesiten manejar los problemas de salud, para realizar o facilitar algún proceso general o específico.

### **4.2 Modelo de Implementación.**

El modelo de implementación es una colección de componentes y los subsistemas que los contienen. Estos componentes incluyen ficheros ejecutables, ficheros de código fuente, y cualquier otro tipo de ficheros necesarios para la implantación y despliegue del sistema. Describe también como se organizan los componentes de acuerdo con los mecanismos de estructuración y modularización disponibles en el entorno de implementación y en el lenguaje o lenguajes de programación utilizados, y cómo dependen de los componentes unos de otros. [69]

El modelo de implementación define una jerarquía tal y como se ilustra en la siguiente figura:

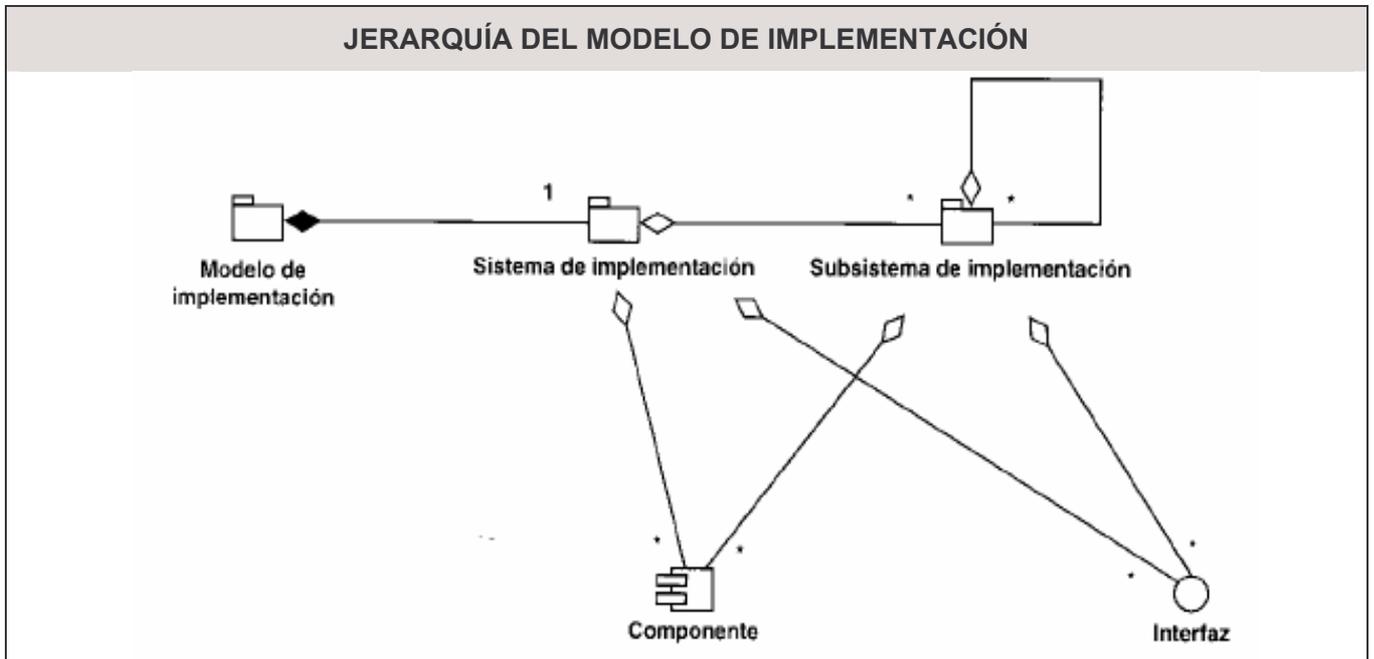


Figura 4.1 Jerarquía del Modelo de Implementación

Este modelo tiene dos artefactos esenciales planteados por la metodología RUP, el diagrama de componentes y el diagrama de despliegue, elementos que constituyen dos de las vistas de la arquitectura y que son significativos en el flujo de implementación.

## 4.2.1 Diagrama de Componentes.

Los diagramas de componentes modelan la estructura del software en un conjunto de componentes, agrupados en ocasiones por paquetes. Muestran la organización y las dependencias lógicas entre un conjunto de paquetes y componentes de software, siendo estos últimos componentes fuentes, binarios o ejecutables. Los componentes de software presentan tipo, que indica si son útiles en tiempo de compilación, enlace o ejecución. [70]

Además estos diagramas de componentes cubren la vista de la implementación estática y se relacionan con los diagramas de clases, ya que un componente suele tener una o más clases, interfaces o colaboraciones, dejando así una traza con el modelo de diseño.

# CAPÍTULO 4. IMPLEMENTACIÓN

Se consideran en este tipo de diagramas sólo tipos de componentes, instancias específicas que se encuentran en el diagrama de ejecución. A continuación se muestra una vista general del modelo de implementación expresado en un diagrama de componentes.

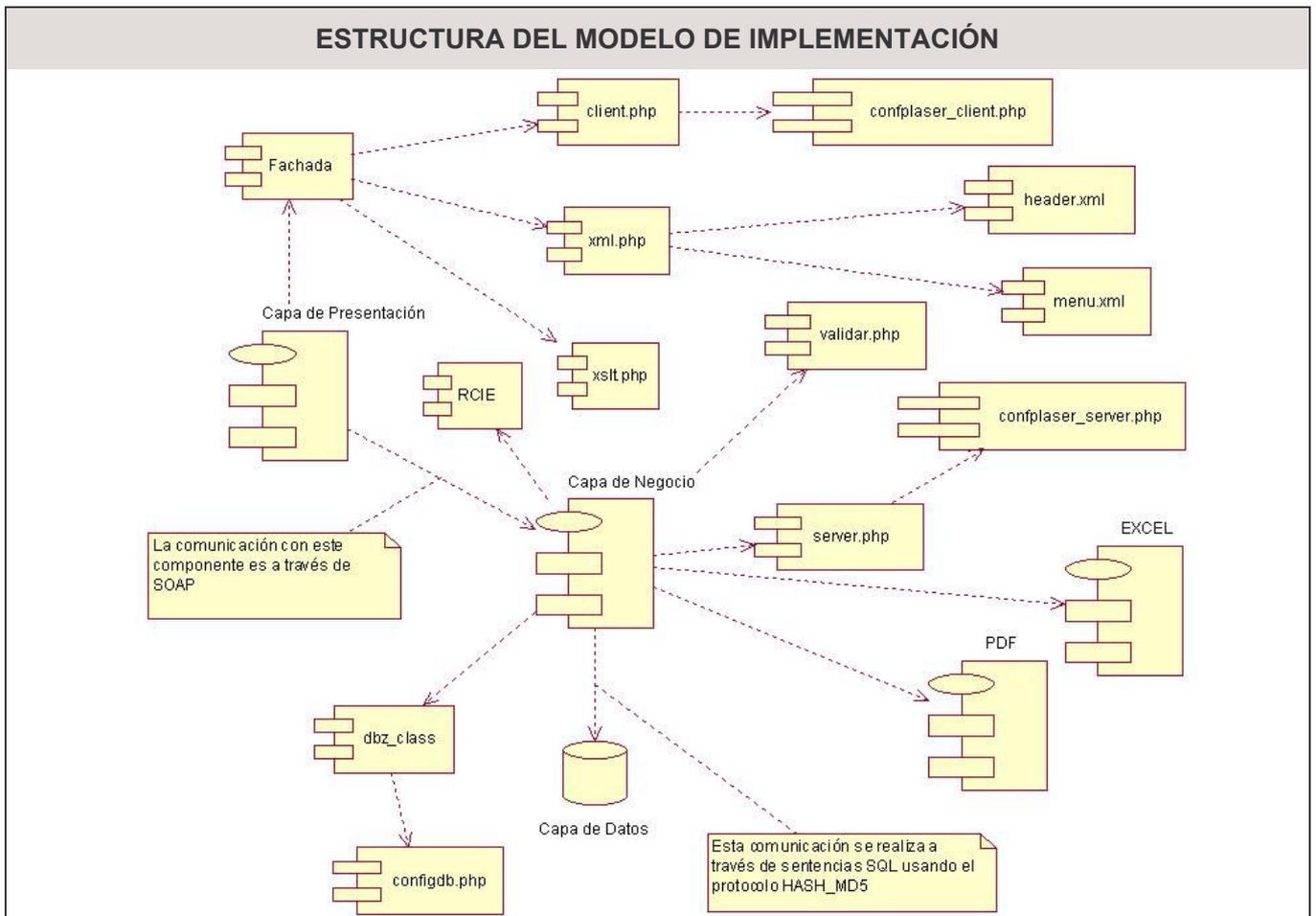


Figura 4.2 Modelo Estructura del Modelo de Implementación

## 4.2.2 Diagrama de Despliegue.

Un diagrama de despliegue representa las relaciones físicas entre los componentes hardware y software en el sistema final o la configuración de los nodos de procesamiento en tiempo de ejecución y los componentes de software que residen en ellos. Muestran la vista de despliegue estática de una arquitectura y se relacionan con los componentes, ya que por lo común, los nodos contienen una o más

## CAPÍTULO 4. IMPLEMENTACIÓN

instancias de componentes de software que representan manifestaciones del código en tiempo de ejecución. [71]

Este diagrama es un grafo de nodos unidos por conexiones de comunicación. Un nodo puede contener instancias de componentes software, objetos y procesos. En general los nodos representan objetos físicos con recursos computacionales como procesadores y periféricos; pueden mostrarse como una clase (una familia de procesadores) o una instancia, por lo que su nombre sigue la misma sintaxis establecida para clases y objetos. Las conexiones son asociaciones de comunicación entre los nodos, y se etiquetan con un estereotipo que identifica el protocolo de comunicación o la red utilizada. Seguidamente se representa este diagrama:

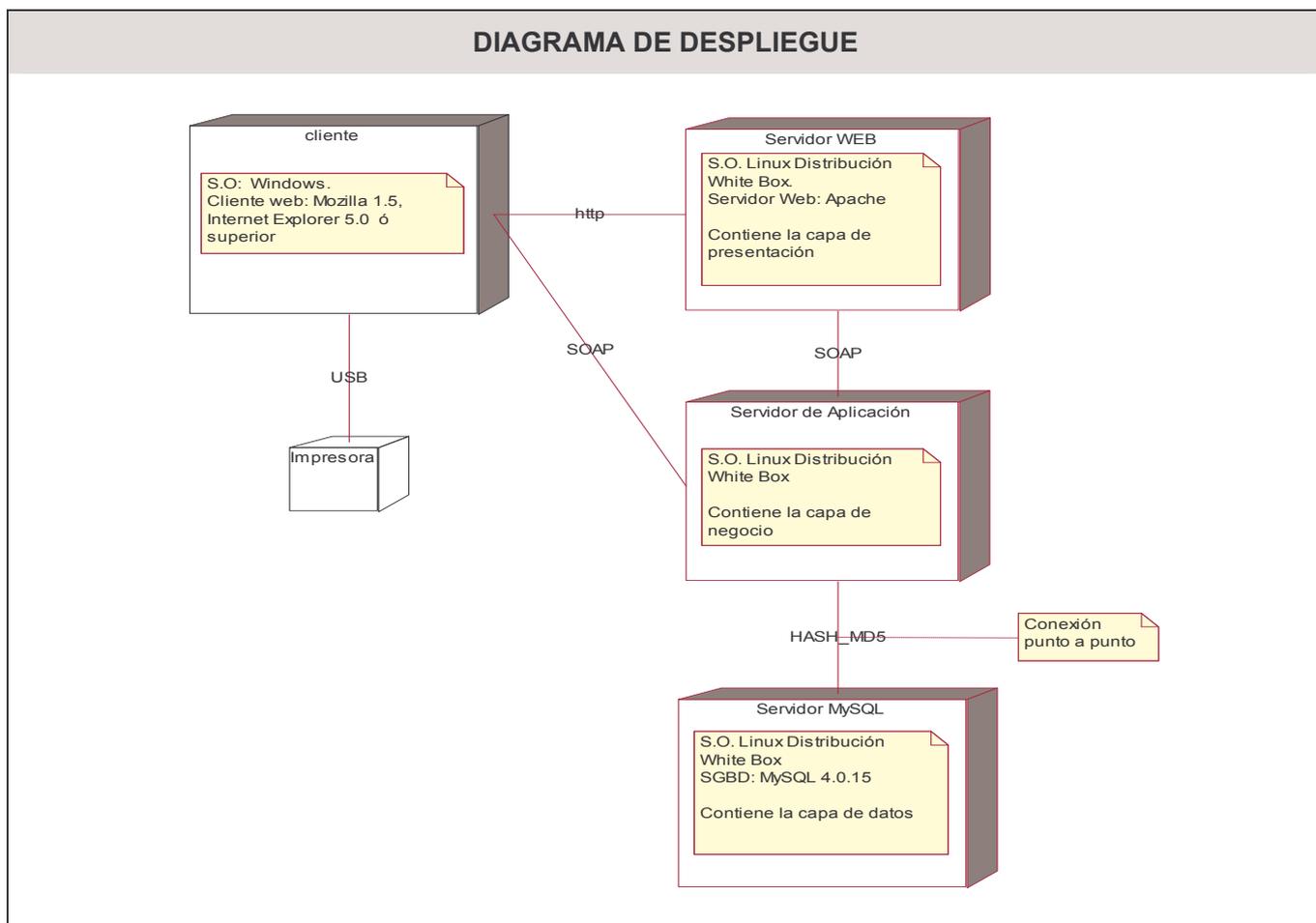


Figura 4.3 Modelo de Despliegue

### 4.3 Descripción de Métodos y Agentes.

La capa de negocio del sistema está compuesta por varios ficheros, entre ellos se encuentran ficheros de configuración y métodos del negocio, estos últimos son los que llevan el peso de las responsabilidades de esta capa.

Los métodos del negocio son ficheros PHP, que contienen las operaciones que deben ser estimuladas por los usuarios de la capa de presentación y ejecutadas por estos métodos, para mantener de esta forma la integridad de la información en la base de datos. Sus nombres son descriptivos de la operación en específico que realizan, por ejemplo: "BuscarRiesgo.php", se refiere al método que se encarga de buscar los riesgos en la base de datos. A continuación se describirán algunos de los métodos más importantes del sistema.

| <b>Nombre del método: TaxonomiaDiscapacidades.php.</b>   |                             |
|--|-----------------------------|
| <b>Descripción:</b> brinda al usuario (editor o visualizador) la posibilidad de que liste las discapacidades definidas para la atención primaria de forma taxonómica. La estructura taxonómica nos muestra las discapacidades agrupadas en sus respectivos subgrupos y grupos, lo que proporciona una vista global de todas las discapacidades existentes. |                             |
| <b>Parámetros de entrada</b>   | <b>Parámetros de salida</b> |
| No presenta  | id_grupo                    |
|  | nombre_grupo                |
|  | id_subgrupo                 |
|  | nombre_subgrupo             |
|  | id_discapacidad             |
|  | nombre_discapacidad         |

Tabla 4.1 Descripción del método TaxonomiaDiscapacidad.php

## CAPÍTULO 4. IMPLEMENTACIÓN

| <b>Nombre del método: InsertarDiscapacidad.php.</b>   |                             |
|---|-----------------------------|
| <b>Descripción:</b> brinda al usuario (editor a nivel nacional) la posibilidad de insertar las discapacidades redefinidas para la atención primaria. Registrando de esta forma las discapacidades para que sean consultadas por los diferentes usuarios y sistemas que necesitan de esta información. |                             |
| <b>Parámetros de entrada</b>  | <b>Parámetros de salida</b> |
| nombre_discapacidad   | No presenta                 |
| descripcion_discapacidad  |                             |
| id_subgrupo   |                             |
| array_codigo_CIE  |                             |
| array_descripcion_CIE   |                             |

Tabla 4.2 Descripción del método InsertarDiscapacidad.php

| <b>Nombre del método: BuscarRcieDiscapacidades.php.</b>  |                               |
|--|-------------------------------|
| <b>Descripción:</b> brinda al usuario (editor o visualizador) la posibilidad de que busque las discapacidades definidas para la atención primaria y su relación con la CIE-10, proporcionando de manera opcional criterios de búsquedas para acotar las mismas. El vínculo de las discapacidades de la atención primaria con la CIE-10 garantiza el establecimiento de un lenguaje común entre organizaciones e instituciones internacionales. |                               |
| <b>Parámetros de entrada</b>   | <b>Parámetros de salida</b>   |
| nombre_discapacidad  | id_discapacidad               |
| id_discapacidad  | nombre_discapacidad           |
| nombre_subgrupo  | descripcion_discapacidad      |
| nombre_grupo   | nombre_subgrupo               |
| id_grupo   | id_grupo                      |
| ordenar_por  | array_codigo_cie_discapacidad |

## CAPÍTULO 4. IMPLEMENTACIÓN

|          |                                    |
|----------|------------------------------------|
| ordena   | array_descripcion_cie_discapacidad |
| offset   | total                              |
| cantidad | offset                             |
|          | cantidad                           |

Tabla 4.3 Descripción del método BuscarRcieDiscapacidad.php

| <b>Nombre del método: BuscarGrupoCIE.php.</b>  |                      |
|--|----------------------|
| <p><b>Descripción:</b> a los métodos de negocio que se encargan de interactuar con los servicios de otros módulos y consumirlos al mismo tiempo, se le llaman “agentes”, este es un ejemplo de ellos. Su función es brindar al usuario (editor a nivel nacional) la posibilidad de buscar los grupos de la CIE-10 en el Registro de la Clasificación Internacional de Enfermedades y Problemas Relacionados con la Salud, para que le sirva al usuario como criterio de búsqueda para mostrar las diferentes subcategorías y enfermedades establecidas en la CIE-10.</p> |                      |
| Parámetros de entrada  | Parámetros de salida |
| id_capitulo  | id_capitulo          |
| id_grupo   | id_grupo             |
| codigo_grupo   | codigo_grupo         |
| descripcion  | descripcion          |
| incluye  | incluye              |
| excluye  | excluye              |
| observaciones  | observaciones        |
| offset   | total                |
| cantidad   | offset               |
|  | cantidad             |

Tabla 4.4 Descripción del método BuscarGrupoCIE.php

## CAPÍTULO 4. IMPLEMENTACIÓN

| <b>Nombre del método: BuscarProblemaSaludCIE.php.</b>   |                             |
|---|-----------------------------|
| <b>Descripción:</b> a los métodos de negocio que se encargan de interactuar con los servicios de otros módulos y consumirlos al mismo tiempo, se le llaman “agentes”, este es un ejemplo de ellos. Su función es brindar al usuario (editor a nivel nacional) la posibilidad de buscar las subcategorías y enfermedades de la CIE-10 en el Registro de la Clasificación Internacional de Enfermedades y Problemas Relacionados con la Salud, para ser seleccionadas por el usuario e insertar parte de su información en el Registro de Problemas de Salud de Atención Primaria para garantizar el estableciendo de un lenguaje común entre organizaciones e instituciones internacionales. |                             |
| <b>Parámetros de entrada</b>  | <b>Parámetros de salida</b> |
| codigo_sub_categoria  | codigo_sub_categoria        |
| codigo_enfermedad   | codigo_enfermedad           |
| descripcion   | descripcion                 |
| offset  | offset                      |
| cantidad  | cantidad                    |
|   | total                       |

Tabla 4.5 Descripción del método BuscarProblemaSaludCIE.php

### 4.4 Estándares Utilizados.

En el campo de la informática se define un estándar como: “una especificación que regula la realización de ciertos procesos o la fabricación de componentes para garantizar la interoperabilidad. También significa un modelo o guía que se sigue para realizar un proceso o para no desviarnos de un objetivo”. [72]

También puede decirse que estándar en informática, es un conjunto de especificaciones técnicas utilizadas para unificar el desarrollo de hardware o de software. Los estándares de computadoras se desarrollan tradicionalmente de dos maneras. Una de ellas, la menos formal, tiene lugar cuando una compañía desarrolla un producto o una filosofía en solitario y consigue convertir el concepto en un estándar por la popularidad de la idea o por las imitaciones fabricadas por los competidores. Otra forma

más ortodoxa de crear un estándar es la redacción de las especificaciones por un grupo de expertos o un comité. Esta redacción se hace después de llevar a cabo un estudio exhaustivo de los métodos existentes, las propuestas y las tendencias o desarrollos tecnológicos.

### 4.4.1 Estándar de Diseño.

En el diseño de una interfaz para un sitio web es necesario tomar en cuenta la interacción de un usuario con un objetivo determinado y un espacio virtual, el sitio web, donde se encuentra el objeto buscado por el usuario. Deben evitarse problemas comunes como el exceso de entretenimiento e interactividad innecesarias, estos influyen negativamente en el acceso del usuario a la información ofrecida y/o buscada dentro del sitio. Algunos efectos multimedia pueden ser muy atractivos, pero dentro de un sitio público traen consigo problemas de carga lenta del sitio web, poca compatibilidad en diferentes navegadores y versiones de los mismos.

El diseño de la interfaz debe responder a las consignas de comunicación para las que fue pensada, es decir, que dé relevancia a las áreas principales del sitio y en segundo lugar a los contenidos secundarios. La correcta planificación de un sitio web reduce considerablemente el tiempo de diseño de todas las instancias de navegación y ayuda a lograr más fácilmente la uniformidad en el diseño general.

El diseño de una interfaz eficaz va a estar dado también por la posibilidad de que diferentes usuarios con posibilidades diversas puedan acceder a los contenidos que un sitio ofrece (diseño web centrado en el usuario), sin que esto conlleve a tener más de un diseño web para cada tipo de visitante.

Muchas de las instituciones que se dedican a la creación de sitios web, utilizan estándares de diseño como parte de su trabajo diario, ya que permiten que las páginas se desplieguen de manera rápida y se visualicen tal como sus autores las han construido.

Un estándar de diseño se define como: "pautas que se establecen para conseguir uniformidad en el desarrollo de una página web". [73]

## CAPÍTULO 4. IMPLEMENTACIÓN

---

Para la realización del sistema se siguieron las pautas definidas por un conjunto de especialistas de la Empresa Softel de conjunto con los integrantes del Proyecto APS. De manera breve los aspectos más generales que maneja este estándar de diseño son los mencionados a continuación.

Los colores utilizados son relacionados con el entorno para el cual fue creado el sitio web, la salud; por que se han establecido diferentes tonalidades de verde, así como los colores blanco, gris y naranja claro para destacar algunas sesiones que deben acentuarse en algunas páginas. Siempre en buscando que la interfaz sea cómoda a la vista y relajante para el usuario.

Las interfaces de cada página han sido creadas de tal forma que sean ligeras, elevando así su velocidad al ser cargadas, además de que se ha realizado un diseño genérico y portable en varias plataformas y navegadores, recomendando para éste último “Internet Explorer” y “Mozilla”.

También se han definido un conjunto de especificaciones y estilos para los diferentes objetos HTML, como los botones, textarea, select, tablas, filas, columnas, así como para diferentes textos en dependencia de la ubicación que tienen en la interfaz. Además se han creado diferentes tipos de páginas con la finalidad de mantener un entorno similar en todo el módulo, como son páginas de agregar, editar, búsqueda, listado, listado por taxonomía, etc. Todo este conjunto de principios se han establecido con el fin de mantener la uniformidad en el sistema, favorecer y facilitar el entendimiento e identificación rápida de tipos de páginas a través de su estructura y posibilitar una interfaz agradable a la vista del usuario.

Todas las páginas constan de 4 áreas principales (A, B, C y D). El área A está conformada por la parte superior que exhibe los privilegios de cada usuario, como nombre, tipo de usuario (editor o visualizador), nivel (nacional, provincial, municipal y unidad de salud), módulos a los cuales posee derecho, nombre del módulo, logotipo, así como los vínculos a la Página de Inicio, Acerca de y Ayuda. El área B está compuesta por el lateral izquierdo que muestra el menú de las acciones que el usuario una vez autenticado puede realizar en el entorno del módulo. El área C está constituida por el barrer de la página el cual presenta información sobre el Ministerio a que pertenece la aplicación. Por último el área D está estructurada por el lateral derecho y por la parte central, es el área más importante de la página, pues es donde el usuario ejecuta sus operaciones, muestra los resultados de las búsquedas y verifica los datos que se encuentran registrados.

## CAPÍTULO 4. IMPLEMENTACIÓN

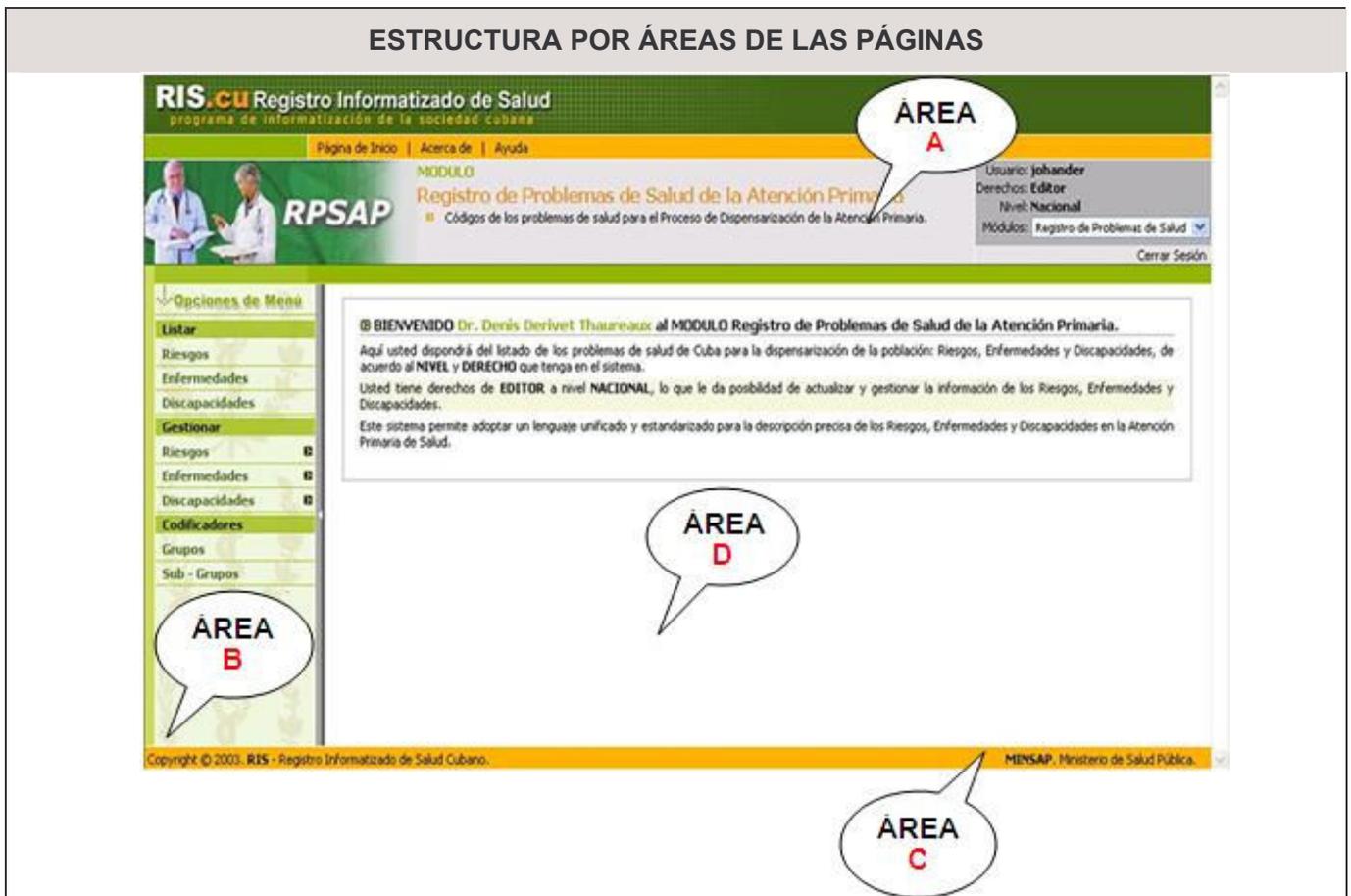


Figura 4.4 Estructura por Áreas de las Páginas

### 4.4.2 Estándar de Codificación.

Un estándar de codificación completo comprende todos los aspectos de la generación de código. Los programadores deben implementar un estándar de forma prudente, éste debe tender siempre a lo práctico. Un código fuente completo debe reflejar un estilo armonioso, como si un único programador hubiera escrito todo el código de una sola vez. Al comenzar un proyecto de software, es necesario establecer un estándar de codificación para asegurarse de que todos los programadores del proyecto trabajen de forma coordinada. Cuando el proyecto de software incorpore código fuente previo, o bien cuando realice el mantenimiento de un sistema de software creado anteriormente, el estándar de codificación debería establecer cómo operar con la base de código existente.

## CAPÍTULO 4. IMPLEMENTACIÓN

---

La legibilidad del código fuente repercute directamente en lo bien que un programador comprende un sistema de software. La mantenibilidad del código es la facilidad con que el sistema de software puede modificarse para añadirle nuevas características, modificar las ya existentes, depurar errores, o mejorar el rendimiento. Aunque la legibilidad y la mantenibilidad son el resultado de muchos factores, una faceta del desarrollo de software en la que todos los programadores influyen especialmente es en la técnica de codificación. El mejor método para asegurarse de que un equipo de programadores mantenga un código de calidad es establecer un estándar de codificación sobre el que se efectuarán luego revisiones del código de rutinas.

Un estándar de codificación se define como “reglas específicas a una lengua que reducen perceptiblemente el riesgo de que los desarrolladores introduzcan errores, no destapan problemas existentes, evitan más bien que los errores ocurran”. [74]

Los aspectos más generales que se incluyen en el estándar de codificación y que se utilizan en el sistema se presentan a continuación:

Las etiquetas de apertura y cierre del lenguaje serán de la forma `<?php?>`, ya que siempre están disponibles en cualquier configuración.

Se usarán 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 lenguaje español, utilizando como separador para las palabras el carácter “\_”, tratando de usar nombres sugerentes a la acción de la variable. Todos los campos identificadores van a comenzar con el identificador (id) seguido del nombre del campo. Ejemplo: `id _ enfermedad`. 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: `BuscarRiesgo`.

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 “/\* \*/”.

## CAPÍTULO 4. IMPLEMENTACIÓN

---

Las llaves se usarán poniendo la llave inicial en una línea para ella sola, y en su respectiva columna la llave final también en una línea.

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. Las bases de datos deben nombrarse poniendo el identificador del proyecto "APS", seguido del carácter "\_" y del nombre del módulo, todo en letra minúscula, ejemplo: bd\_rpsap. Los nombres de las tablas deben ir en minúsculas, cada palabra separada por el guión bajo: "\_" y con el prefijo delante tb (Ejemplo: tb\_enfermedad). Los campos de la base de datos se nombrarán igual que las variables.

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".

Es importante tener en cuenta que la adopción de un estándar de codificación sólo es viable si se sigue desde el principio hasta el final del proyecto de software. No es práctico, ni prudente, imponer un estándar de codificación una vez iniciado el trabajo.

Usar técnicas de codificación sólidas y realizar buenas prácticas de programación con vistas a generar un código de alta calidad es de gran importancia para la calidad del software y para obtener un buen rendimiento. Además, si se aplica de forma continuada un estándar de codificación bien definido, se utilizan técnicas de programación apropiadas, y, posteriormente, se efectúan revisiones del código de rutinas, caben muchas posibilidades de que un proyecto de software se convierta en un sistema de software fácil de comprender y de mantener.

### 4.4.3 Estándar de Tratamiento de Excepciones.

El problema de la seguridad es uno de los clásicos quebraderos de cabeza de la programación. Los diversos lenguajes han tenido siempre que lidiar con el mismo problema: ¿Qué hacer cuando se presenta una circunstancia verdaderamente imprevista? (por ejemplo un error). El asunto es especialmente importante si se trata de lenguajes para escribir programas de "Misión crítica"; digamos por ejemplo

## CAPÍTULO 4. IMPLEMENTACIÓN

---

controlar los ordenadores de una central nuclear o de un sistema de control de tráfico aéreo o algún sistema que contiene información valiosa para la salud o que da soporte al trabajo del médico o a un especialista en un centro médico determinado.

Antes que nada, digamos que en los lenguajes de programación a estas "circunstancias imprevistas" se les hace llamar con el nombre de excepciones, por lo que cualquier sistema informático debe manejar de forma correcta sus excepciones, para no perder información o para poder dar una respuesta acertada en cada situación que se presente.

Se define a las excepciones como "condiciones excepcionales que pueden ocurrir dentro del programa durante su ejecución". [75] Por ejemplo, que ocurra una operación no válida, que la información esté duplicada en la base de datos, etc., que requieren recursos especiales para su control. Una excepción es un evento que ocurre durante la ejecución del programa que interrumpe el flujo normal de las sentencias. Son una forma clara 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, si se recibe enseguida, permite gestionar un error.

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

El módulo RPSAP también maneja las excepciones en tiempo de ejecución de manera cuidadosa, siguiendo el estándar para el tratamiento de errores y excepciones definido proyecto APS. Los aspectos más generales de éste se describen a continuación:

Las excepciones, así como los errores causados por el usuario en la interfaz se han manejado a través de los mensajes 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 válidos, o les sea posible corregirlos en caso contrario.

## CAPÍTULO 4. IMPLEMENTACIÓN

---

Para elevar las excepciones desde la capa de presentación se ha utilizado la función “plaser\_exit” implementada en la librería de clases “PlaSer”, la cual tiene como único parámetro el mensaje de error, que es procesado por esta función de PlaSer, retornando un XML con el error que se muestra en la página de error de la aplicación.

Para levantar una excepción desde la capa de negocio se ha utilizado una función llamada: “fault” implementadas ambas en la librería de clases “PlaSer”, cuyos parámetros son Detail, FaultString, FaultActor, FaultCode y HttpServer los cuales se describen a continuación:

**Detail:** Este elemento se usa para llevar mensajes de error específicos de aplicaciones, se empleará únicamente en errores cuya resolución depende del Centro de Control, en cualquier otro caso este elemento debe estar vacío. Este mensaje se debe escribir con letra mayúscula.

**FaultString:** Una explicación del error asequible al humano (leíble y explicativo). Debe tenerse en cuenta que este texto puede ser mostrado al operador final del sistema. Este mensaje se debe escribir con letra mayúscula. Ejemplo: Formato de entrada no válido para la fecha de cierre estadístico.

**FaultActor:** Un texto que indica quien provocó el error, siempre será el nombre del método que eleva la excepción. Ejemplo: InsertaDiscapacidad.

**FaultCode:** Código de texto utilizado para indicar la clase de error, será codificado de la siguiente manera. Código del proyecto-código del módulo (:) número del método (.) número del error. Ejemplo: APS-RPSAP: 1.5 que indica error 5 en el método 1 del módulo Registro de Problemas de Salud de la Atención Primaria perteneciente al Proyecto APS.

**HttpServer:** En este elemento se envía la dirección del servidor al cual se quiere mandar el mensaje de error, se usa para elevar las excepciones de un servidor a otro en caso de que el sistema esté subdividido por capas en diferentes servidores, en caso de que esté todo en un mismo servidor no debe utilizarse.

## CAPÍTULO 4. IMPLEMENTACIÓN

---

También se utilizaron codificadores para evitar posibles errores por parte del usuario al registrar información de poca variabilidad, se pueden citar los casos de los grupos y subgrupos que agrupan a los riesgos, enfermedades y discapacidades.

### **Conclusiones.**

En este capítulo se describieron los elementos esenciales relacionados con la implementación. Se fundamentó la integración con otros sistemas, se obtuvo la estructura del modelo de implementación, el modelo de despliegue, los estándares de diseño, codificación y tratamientos de excepciones, así como, una breve descripción de algunos de los métodos y agentes más significativos. Todos estos fundamentos han posibilitado construir una vista global de la implementación del sistema.

### Conclusiones

La realización del presente trabajo ha posibilitado cumplir con los objetivos propuestos, por lo que se pueden plantear las siguientes conclusiones:

1. Se identificaron las interacciones de este módulo, con otros del Sistema de Información para la Salud.
2. Se elaboró una documentación coherente de todos los artefactos, realizando Modelamiento del Negocio, Gestión de Requerimientos.
3. Se diseñaron todos los componentes del sistema exitosamente, obteniendo como resultado un producto de calidad.
4. Se desarrolló una aplicación web que gestiona la información relacionada con los problemas de salud en el nivel de Atención Primaria, garantizando la realización de forma eficiente del proceso de dispensarización y diagnóstico en este nivel, lo que facilitará el procesamiento estadístico, estándar y homogéneo de la información.

### Recomendaciones

Las recomendaciones de la investigación están dirigidas a sugerir acciones para complementar el producto obtenido. Por lo que para el buen desempeño y puesta en marcha de la aplicación se hacen las siguientes recomendaciones:

1. Incluir en la concepción del Componente de Avisos de SISalud los mensajes generados por actualizaciones que se realizan en el Registro de Problemas de Salud de la Atención Primaria.
2. Definir el plan de capacitación necesario para que el personal de salud de las diferentes unidades asistenciales se familiaricen con el sistema.
3. Publicar los servicios web del Registro de Problemas de Salud de la Atención Primaria para que sean usados de manera eficiente por todos los módulos o componentes que requieran de los problemas de salud definidos para la atención primaria.
4. Incluir un nuevo nivel, el de deficiencias, a la estructura de clasificación de la atención primaria; para enriquecer la clasificación en este nivel y facilitar el trabajo del médico.

### Referencias Bibliográficas

1. **Consejo Ejecutivo de la OMS.** Promoviendo la Salud en las Américas. Sitio Web de la Organización Panamericana de la Salud. [En línea] 11 de Noviembre de 2003. [Citado el: 5 de Febrero de 2007.] <http://www.ops.org.gt/APS/APS.htm>.
2. Ídem a la referencia 1.
3. Ídem a la referencia 1.
4. **Fidel Castro Ruz.** Portal Cuba.cu. [En línea] 7 de Abril de 2003. [Citado el: 16 de Marzo de 2007.] <http://www.cuba.cu/gobierno/discursos/2003/esp/f070403e.html>.
5. **Ariel Delgado Ramos y María Vidal Ledo.** Revista Cubana de Salud Pública. [En línea] Marzo de 2006. [Citado el: 22 de Abril de 2007.] [http://bvs.sld.cu/revistas/spu/vol32\\_3\\_06/spu15306.htm](http://bvs.sld.cu/revistas/spu/vol32_3_06/spu15306.htm).
6. Ídem a la referencia 5.
7. Ídem a la referencia 5.
8. **Colectivo de Autores.** Wikipedia.com. [En línea] 12 de Junio de 2007. [Citado el: 15 de Junio de 2007.] <http://es.wikipedia.org/wiki/Informaci%C3%B3n>.
9. **Lourdes Aja Quiroga.** Biblioteca Virtual de Salud. Biblioteca Virtual de Salud de Cuba. [En línea] 10 de Mayo de 2002. [Citado el: 10 de Febrero de 2007.] [http://bvs.sld.cu/revistas/aci/vol10\\_5\\_02/aci04502.htm](http://bvs.sld.cu/revistas/aci/vol10_5_02/aci04502.htm).
10. **Colectivo de Autores.** WordReference.com. [En línea] 2007. [Citado el: 20 de Marzo de 2007.] <http://www.wordreference.com/definicion/almacenamiento>.
11. **Manuel Machuca, Martha Bibiana Oñate, María José Faus.** Farmacare.com. [En línea] 2003. [Citado el: 27 de Febrero de 2007.] [http://www.farmacare.com/revista/n\\_3/139-140.pdf](http://www.farmacare.com/revista/n_3/139-140.pdf).
12. **Luis Fernando Llanos Zavalaga.** Revista Medica Herediana. [En línea] Diciembre de 2005. [Citado el: 20 de Febrero de 2007.] [http://www.scielo.org.pe/scielo.php?script=sci\\_arttext&pid=S1018-130X2005000400003&lng=pt&nrm=iso&tlng=es](http://www.scielo.org.pe/scielo.php?script=sci_arttext&pid=S1018-130X2005000400003&lng=pt&nrm=iso&tlng=es).
13. **Colectivo de Autores.** Wikipedia.com. [En línea] 7 de Junio de 2007. [Citado el: 16 de Junio de 2007.] [http://es.wikipedia.org/wiki/Aplicaci%C3%B3n\\_web](http://es.wikipedia.org/wiki/Aplicaci%C3%B3n_web).
14. **Jesús Lázaro Amador Alvarez.** Ilustrados.com. [En línea] 2 de Junio de 2006. [Citado el: 18 de Enero de 2007.] <http://www.ilustrados.com/publicaciones/EEuAklupVEaHIOMFbu.php>.
15. **Colectivo de Autores.** Carpeta Metodológica de Atención Primaria de Salud y Medicina Familiar. España : Tallers Gráfcics Canigó, S.L., 2000.
16. Ídem a la referencia 15.

## REFERENCIAS BIBLIOGRÁFICAS

---

17. Ídem a la referencia 15.
18. Ídem a la referencia 15.
19. Ídem a la referencia 15.
20. Ídem a la referencia 15.
21. **Colectivo de Autores**. Wikipedia.com. [En línea] 6 de Junio de 2007. [Citado el: 10 de Junio de 2007.] <http://es.wikipedia.org/wiki/Diagn%C3%B3stico>.
22. Ídem a la referencia 21.
23. **Mirna Cabrera Hernández, Juan Carlos García Morales, Alberto Acuña Sánchez, Lucía Domínguez Abreu y Denis Derivet Thaureaux**. Especificación Preliminar de Requisitos del Registro de Actividades Diarias. Ciudad de la Habana : Empresa de Producción de Software SOFTEL, 2004.
24. Ídem a la referencia 23.
25. **Juan Pablo Villa**. Portal de Prevención y Tratamiento del Cáncer en la Infancia y la Adolescencia. [En línea] 2007. [Citado el: 22 de Mayo de 2007.] <http://censia.salud.gob.mx/variados/docs%20nut/diagnostico%20de%20salud.pdf>.
26. **Rodolfo J. Stusser Beltranena y Alfredo Rodríguez Díaz**. Biblioteca Virtual de Salud de Cuba. [En línea] 22 de Abril de 2006. [Citado el: 12 de Febrero de 2007.] [http://bvs.sld.cu/revistas/mgi/vol22\\_4\\_06/mgi12406.htm](http://bvs.sld.cu/revistas/mgi/vol22_4_06/mgi12406.htm).
27. Ídem a la referencia 26.
28. Ídem a la referencia 26.
29. **Colectivo de Autores**. Biblioteca Virtual de Salud de Cuba. [En línea] 2007. [Citado el: 6 de Marzo de 2007.] [http://www.sld.cu/galerias/pdf/sitios/dne/vol2\\_descripcion.pdf](http://www.sld.cu/galerias/pdf/sitios/dne/vol2_descripcion.pdf).
30. **Francisco J. Vaz Leal y Maria Angeles Cano Carrasco**. Portal de la Facultad de Medicina de Badajoz. [En línea] 2007. [Citado el: 22 de Mayo de 2007.] <http://www.futuex.com/archivo.php?id=2255>.
31. **Carlos Egea García**. DisWeb2000. [En línea] 2000. [Citado el: 16 de Marzo de 2007.] <http://usuarios.discapnet.es/disweb2000/CIDDM/ciddm.htm>.
32. **Marco Jamouille**. Portal del Comité Internacional de Clasificación de la WONCA. [En línea] 10 de Agosto de 1998. [Citado el: 29 de Febrero de 2007.] <http://www.ulb.ac.be/esp/wicc/ccw-es.html>.
33. **Ginés González García**. Boletín PROAPS - REMEDIAR. [En línea] 14 de Octubre de 2004. [Citado el: 21 de Enero de 2007.] [http://www.femeba.org.ar/fundacion/quienessomos/Novedades/boletines\\_remediar/boletinremediar14.pdf](http://www.femeba.org.ar/fundacion/quienessomos/Novedades/boletines_remediar/boletinremediar14.pdf).
34. Ídem a la referencia 33.

## REFERENCIAS BIBLIOGRÁFICAS

---

35. **Rodolfo J. Stusser Beltranena, Marco J. Albert Cabrera, Alfredo Rodríguez Díaz, Suiberto Echevarría Toledo, René I. González Fernández y Alain Cuadot Álvarez.** Biblioteca Virtual de Salud de Cuba. [En línea] 22 de Abril de 2006. [Citado el: 18 de Febrero de 2007.]  
[http://bvs.sld.cu/revistas/mgi/vol22\\_4\\_06/mgi14406.htm](http://bvs.sld.cu/revistas/mgi/vol22_4_06/mgi14406.htm).
36. Ídem a la referencia 35.
37. Ídem a la referencia 35.
38. Ídem a la referencia 26.
39. Ídem a la referencia 26.
40. Ídem a la referencia 26.
41. **Colectivo de Autores.** Portal de Salud de Cuba. Infomed. [En línea] 2000. [Citado el: 16 de Febrero de 2007.] [http://www.sld.cu/sistema\\_de\\_salud/aspectos.html](http://www.sld.cu/sistema_de_salud/aspectos.html).
42. **Abelardo Ramírez Márquez, Pastor Castell-Florit Serrate y Guillermo Mesa.** Portal Escuela Nacional Salud Pública. [En línea] 2003. [Citado el: 26 de Marzo de 2007.]  
[http://www.sld.cu/galerias/doc/sitios/infodir/09\\_el\\_sistema\\_nacional\\_de\\_salud.doc](http://www.sld.cu/galerias/doc/sitios/infodir/09_el_sistema_nacional_de_salud.doc).
43. Ídem a la referencia 42.
44. Ídem a la referencia 42.
45. Ídem a la referencia 41.
46. Ídem a la referencia 41.
47. Ídem a la referencia 41.
48. Ídem a la referencia 41.
49. Ídem a la referencia 41.
50. Ídem a la referencia 5.
51. Ídem a la referencia 5.
52. **Colectivo de Autores de la Wonca.** Portal World Health Organization. [En línea] 2007. [Citado el: 22 de Abril de 2007.] <http://www.who.int/en/>.
53. Ídem a la referencia 52.
54. Ídem a la referencia 52.
55. **Craig Larman.** UML y Patrones. Ciudad de la Habana : Félix Varela, 2004.
56. **Ivar Jacobson, Grady Booch y James Rumbaugh.** El Proceso Unificado de Desarrollo de Software. Massachusetts, USA : Addison Wesley, 2000.
57. Ídem a la referencia 53.

## REFERENCIAS BIBLIOGRÁFICAS

---

58. Ídem a la referencia 53.
59. Ídem a la referencia 53.
60. Ídem a la referencia 53.
61. Ídem a la referencia 52.
62. **Colectivo de Autores.** Portal Desarrollo MSDN España . [En línea] 2007. [Citado el: 9 de Mayo de 2007.]  
<http://www.desarrollaconmsdn.com/msdn/Help/Implementacion/Implementaci%C3%B3n/Dispensador/MVC.htm>.
63. Ídem a la referencia 59.
64. Ídem a la referencia 59.
65. **Carlos Reynoso y Nicolás Kicillof.** Portal Desarrollo MSDN en Español. [En línea] Marzo de 2004. [Citado el: 2 de Mayo de 2007.].  
[http://www.microsoft.com/spanish/msdn/arquitectura/roadmap\\_arq/style.asp#11](http://www.microsoft.com/spanish/msdn/arquitectura/roadmap_arq/style.asp#11)
66. Ídem a la referencia 53.
67. **Josep Vilalta.** Vico.org. [En línea] 17 de Julio de 2002. [Citado el: 19 de Mayo de 2007.]  
[http://vico.org/aRecursosPrivats/ClipsPerEnganxarMail/TRAD\\_UML\\_WAE.pdf](http://vico.org/aRecursosPrivats/ClipsPerEnganxarMail/TRAD_UML_WAE.pdf).
68. Ídem a la referencia 53.
69. Ídem a la referencia 53.
70. Ídem a la referencia 53.
71. Ídem a la referencia 53.
72. **Colectivo de Autores.** Wikipedia.com. [En línea] 2 de Junio de 2007. [Citado el: 22 de Mayo de 2007.] <http://es.wikipedia.org/wiki/Estandar>.
73. **Colectivo de Autores.** Anuncios Google. [En línea] 2007. [Citado el: 26 de Mayo de 2007.]  
<http://www.slideshare.net/dwebslide/estandares-de-diseo-web/>.
74. **Colectivo de Autores.** Portal SynSpace de Prevención Automatizada de Errores. [En línea] 17 de Agosto de 2005. [Citado el: 27 de Mayo de 2007.] <http://www.synspace.com/ES/Services/tcc.html>.
75. **Paul Somerson.** Portal Zator Systems. [En línea] 2006. [Citado el: 28 de Mayo de 2007.]  
[http://www.zator.com/Cpp/E1\\_6.htm](http://www.zator.com/Cpp/E1_6.htm).

## Bibliografía

1. **Ariel Delgado Ramos y María Vidal Ledo.** Revista Cubana de Salud Pública. [En línea] Marzo de 2006. [Citado el: 22 de Abril de 2007.] [http://bvs.sld.cu/revistas/spu/vol32\\_3\\_06/spu15306.htm](http://bvs.sld.cu/revistas/spu/vol32_3_06/spu15306.htm).
2. **Abelardo Ramírez Márquez, Pastor Castell-Florit Serrate y Guillermo Mesa.** Portal Escuela Nacional Salud Pública. [En línea] 2003. [Citado el: 26 de Marzo de 2007.] [http://www.sld.cu/galerias/doc/sitios/infodir/09\\_el\\_sistema\\_nacional\\_de\\_salud.doc](http://www.sld.cu/galerias/doc/sitios/infodir/09_el_sistema_nacional_de_salud.doc).
3. **Colectivo de Autores.** Carpeta Metodológica de Atención Primaria de Salud y Medicina Familiar. España : Tallers Gràfics Canigó, S.L., 2000.
4. **Colectivo de Autores.** Portal de Salud de Cuba. Infomed. [En línea] 2000. [Citado el: 16 de Febrero de 2007.] [http://www.sld.cu/sistema\\_de\\_salud/aspectos.html](http://www.sld.cu/sistema_de_salud/aspectos.html).
5. **Colectivo de Autores de Microsoft.** Portal Desarrollo MSDN en Español. [En línea] 10 Mayo de 2004. [Citado el: 2 de Mayo de 2007.]. [http://www.microsoft.com/spanish/msdn/arquitectura/roadmap\\_arq/arquitectura\\_soft.asp](http://www.microsoft.com/spanish/msdn/arquitectura/roadmap_arq/arquitectura_soft.asp)
6. **Colectivo de Autores de la Wonca.** Portal World Health Organization. [En línea] 2007. [Citado el: 22 de Abril de 2007.] <http://www.who.int/en/>.
7. **Craig Larman.** UML y Patrones. Ciudad de la Habana : Félix Varela, 2004.
8. **Ivar Jacobson, Grady Booch y James Rumbaugh.** El Proceso Unificado de Desarrollo de Software. Massachusetts, USA : Addison Wesley, 2000.
9. **Ivar Jacobson, Grady Booch y James Rumbaugh.** El Lenguaje Unificado de Modelado. Manual de Referencia. Massachusetts, USA : Addison Wesley, 2000.
10. **Mirna Cabrera Hernández, Juan Carlos García Morales, Alberto Acuña Sánchez, Lucía Domínguez Abreu y Denis Derivet Thureaux.** Especificación Preliminar de Requisitos del Registro de Actividades Diarias. Ciudad de la Habana : Empresa de Producción de Software SOFTEL, 2004.
11. **Mirna Cabrera Hernández, Juan Carlos García Morales, Alberto Acuña Sánchez, Lucía Domínguez Abreu y Denis Derivet Thureaux.** Especificación Preliminar de Requisitos del Registro de Población. Ciudad de la Habana : Empresa de Producción de Software SOFTEL, 2004.
12. **Rodolfo J. Stusser Beltranena, Marco J. Albert Cabrera, Alfredo Rodríguez Díaz, Suiberto Echevarría Toledo, René I. González Fernández y Alain Cuadot Álvarez.** Biblioteca Virtual de Salud de Cuba. [En línea] 22 de Abril de 2006. [Citado el: 18 de Febrero de 2007.] [http://bvs.sld.cu/revistas/mgi/vol22\\_4\\_06/mgi14406.htm](http://bvs.sld.cu/revistas/mgi/vol22_4_06/mgi14406.htm).

## Glosario de Términos

**Aplicación o Sistema Informático:** Programas con los cuales el usuario final interactúa a través de una interfaz y que realizan tareas útiles para éste.

**Aplicación Monolítica:** Sistema o modelo de desarrollo de aplicaciones, que se caracteriza por ser un bloque indivisible en el que se gestionan los datos, particularmente su almacenamiento y recuperación, se establecen las reglas de negocio y además, se controla la interfaz que utilizará el usuario. Este tipo de aplicaciones suelen requerir máquinas con gran capacidad de proceso, ya que todo el trabajo se desempeña en el puesto sobre el que operará el cliente.

**Cliente Servidor:** Modelo para construir sistemas de información, que se sustenta en la idea de repartir el tratamiento de la información y los datos por todo el sistema informático, permitiendo mejorar el rendimiento del sistema global de información.

**Código Alfanumérico:** Código que contienen datos alfabéticos (letras) y caracteres especiales (tales como símbolos de puntuación y matemáticos) en adición a números.

**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.

**Concurrencia:** Ejecución simultánea de dos o más actividades durante el mismo intervalo de tiempo.

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

**Deficiencia:** Es toda pérdida o anomalía de una estructura o función psicológica, fisiológica o anatómica.

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

# GLOSARIO DE TÉRMINOS

---

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

**Equipos Básicos de Salud:** Binomio conformado por el médico y enfermera de la familia, que atiende una población geográficamente determinada, que puede estar ubicado en la comunidad, centros laborales o educacionales.

**Interoperabilidad:** Condición necesaria para que los usuarios (humanos o mecánicos) tengan un acceso completo a la información disponible. Entre las iniciativas recientes más destacadas para dotar a la Web de interoperabilidad se encuentran los servicios Web y la Web semántica.

**Internet:** Método de interconexión de redes de computadoras implementado en un conjunto de protocolos denominado TCP/IP y garantiza que redes físicas heterogéneas funcionen como una red (lógica) única.

**Informática:** 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.

**Morbilidad:** Es el estudio de los efectos de una enfermedad en una población en el sentido de la proporción de personas que enferman en un sitio y tiempo determinado.

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

**Policlínico:** Unidad de salud donde se brindan servicios médicos a una población geográficamente determinada perteneciente al nivel asistencial de Atención Primaria de Salud.

**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:** es una colección de protocolos y estándares que sirven para intercambiar datos entre aplicaciones.

# GLOSARIO DE TÉRMINOS

---

**SOAP:** Es un protocolo 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.

**Taxonomía:** Clasificación u ordenación en grupos de cosas que tienen unas características comunes.

**Token:** Es un identificador o palabra reservada que constituye una unidad léxica, en la cual se agrupan un conjunto de lexemas que coincidan con un mismo patrón.

**Unidad de Salud:** Centro de trabajo que pertenece al Ministerio de Salud Pública (MINSAP).