

Universidad de las Ciencias Informáticas
Facultad 7



Título: *Registro de la Clasificación Internacional de Enfermedades y Problemas relacionados con la Salud (RCIE) del Sistema de Información para la Salud.*

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

Autores: Jacqueline Marin Sanchez

Carlos Acosta Montejo

Tutores: Ing. Mirna Cabrera Hernández.

Lic. Yamilka Gómez León

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

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 (UCI) todo aquello que no comprometa dicho acuerdo.

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

Jacqueline Marín Sánchez
Autor

Carlos Acosta Montejó
Autor

Ing. Mirna Cabrera Hernández
Tutor

Lic. Yamilka Gómez León
Tutor

DATOS DE CONTACTO

Mirna Cabrera Hernández (mirna@softel.cu): Graduado 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.

Yamilka Gómez León (yamilkagl@uci.cu): Profesor graduado de Licenciatura en Ciencias de la Computación en el año 2004 en la UCLV. Ha impartido las asignaturas Programación 2, Sistemas de Bases de Datos y Gestión de Software. Posee la categoría docente de Instructor y cursa la maestría de Ciencias de la Computación en la UCI. Forma parte del proyecto de Atención Primaria de Salud de la facultad 7.



Nunca consideres el estudio como una obligación, sino como una oportunidad para penetrar en el bello y maravilloso mundo del saber.

Albert Einstein

AGRADECIMIENTOS

Queremos agradecer a todas aquellas personas, que de una forma u otra contribuyeron al desarrollo del presente trabajo de diploma, aportando su granito de arena para que nuestros sueños se hicieran realidad.

A nuestro Comandante en Jefe y a la Revolución por concedernos la oportunidad de estudiar, y convertirnos en profesionales dispuestos a dar lo mejor de sí. A la Universidad por la confianza depositada en nosotros, por poner a nuestro alcance tantas fuentes de conocimiento y sobre todas las cosas, por la posibilidad de pertenecer a este gran Proyecto y reafirmarnos como “Jóvenes Futuro”.

A Mirna, las gracias por haber contribuido a nuestra formación personal y profesional, por ser, además de tutora y jefa de proyecto, maestra, ejemplo y guía, madre y amiga. Por sus días de dedicación, consejos y confianza en nosotros.

Al equipo de especialistas de Softel, gracias por su apoyo incondicional y su influencia día a día en nuestra correcta formación: Cary, Lucy Domínguez, Denis, Chino, Alfredo, Rosalía, Lucy Cruz, Ana Silvia, Carmenchun. A los profesores que no solo con su sabiduría llenaron nuestras vidas de momentos inolvidables sino también con su desinteresada amistad: Yamilka, Lourdes, Pura y Yovanotti. A nuestros compañeros de proyecto: fue un gusto trabajar junto a ustedes y compartir estos años, gracias a todos por su ayuda.

A nuestros amigos:

Jacqueline: Gracias a mis amigos por apoyarme durante estos años, en especial Maidelys y Yaiser. A Johander agradezco su compañía, su ayuda y el haberme enseñado el camino al optimismo. A mi compañero de tesis por su creatividad, comprensión y confianza.

Carlos: Agradezco a Laritza por ser mi guía todos estos años y soportar mis resabios. A mi compañera de tesis por ser tan exigente y dedicada en la realización de este trabajo.

A nuestras familias, en especial nuestros padres, por la confianza, cariño y educación que siempre nos han brindado, contribuyendo a forjar lo que somos hoy.

DEDICATORIA

A mi familia, en especial a mis padres que me han apoyado en estos 5 años de carrera y en parte son también responsables de estos resultados.

A mis hermanos Alex y Dayán.

A todas las personas que confiaron en mí.

Carlos Acosta Montejo

A las personas que más quiero en el mundo: mi familia.

En especial mis padres y hermana por el apoyo y comprensión que siempre me han brindado.

A mi madre, mi máspreciado tesoro.

A mi tía y mis primas.

A Lisset, mi ejemplo y guía.

A mis amigos, por estar siempre presentes.

A todos lo que han compartido junto a mí estos 5 años.

A todos los que confiaron en mí.

Jacqueline Marín Sánchez

RESUMEN

La presente investigación tiene como objetivo desarrollar una aplicación web, que automatice la gestión de la información de la Clasificación Estadística Internacional de Enfermedades y Problemas Relacionados con la Salud (CIE-10) y brinde la posibilidad de llevar a cabo un proceso de diagnóstico homogéneo en cada una de las aplicaciones informáticas del Sistema Nacional de Salud.

Se estudian los antecedentes históricos e importancia de la Clasificación Internacional de Enfermedades (CIE), los sistemas informáticos existentes a nivel nacional e internacional de su Décima Revisión y se hace una comparación crítica entre los mismos y la solución propuesta. Se exponen las tecnologías y herramientas utilizadas como: servidor web Apache, MySQL, PHP, XML, XSL, todas siguiendo las políticas trazadas por el MINSAP de software libre y código abierto. Además se profundiza en la arquitectura definida: modelo Cliente/Servidor, orientada a servicios y basada en componentes.

La aplicación desarrollada permite realizar búsquedas dinámicas e imprimir los resultados de las mismas, gestionar la información según la estructura de la CIE-10 y brinda servicios a otras aplicaciones integradas al Sistema de Información para la Salud.

Palabras Claves: Sistema de Información para la Salud, Clasificación Internacional de Enfermedades, diagnósticos homogéneos, estadísticas, servicios.

TABLA DE CONTENIDOS

INTRODUCCIÓN	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA	6
1.1 Sistema Nacional de Salud (SNS)	6
1.1.1 Informatización del Sistema Nacional de Salud	7
1.1.2 Infomed.....	10
1.1.3 Registro Informatizado de Salud (RIS)	10
1.1.4 Solución propuesta para la Informatización del SNS	11
1.2 Clasificación Internacional de Enfermedades (CIE)	12
1.2.1 Antecedentes históricos e importancia de la CIE	13
1.2.2 Ejemplos del uso de la CIE-10 en el mundo. Intentos de informatización.....	15
1.2.3 La Clasificación Internacional de Enfermedades en Cuba	16
1.3 Flujo actual de los procesos involucrados en el campo de acción	17
1.4 Situación Problemática.....	17
1.5 Objeto de automatización e información manipulada	18
1.5.1 Análisis comparativo con otras soluciones existentes	19
1.6 Tendencia y tecnologías a considerar	19
XML.....	19
Entornos Distribuidos. Modelo Cliente/Servidor	20
Modelo Cliente/Servidor de tres Capas (Three Tier).....	21
Servidor web Apache	21
Arquitectura Basada en Componentes (CBA)	22
Lenguajes de Programación Web.....	22
PHP.....	23
JavaScript	24
XSL	25
Sistemas de Gestión de Bases de Datos (SGBD)	25
MySQL	26
Desarrollo basado en RUP	26
UML	28

Tabla de Contenidos

Plataforma de Servicio (PLASER)	29
Herramientas a utilizar.....	30
CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA	31
2.1 Modelo Dominio.....	31
2.1.1 Conceptos Fundamentales	32
2.1.2 Diagrama del Modelo de Dominio.....	33
2.2 Propuesta del Sistema	33
2.2.1 Especificación de Requerimientos de Software	34
2.2.1.1 Requerimientos Funcionales	34
2.2.1.2 Requerimientos no Funcionales	37
2.2.1.2.1 Usabilidad.....	37
2.2.1.2.2 Soporte	37
2.2.1.2.3 Portabilidad.....	38
2.2.1.2.4 Seguridad	38
2.2.1.2.5 Apariencia o Interfaz Externa.....	40
2.2.1.2.6 Apariencia o Interfaz Interna.....	40
2.2.1.2.7 Ayuda y Documentación en Línea.....	41
2.2.1.2.8 Software	41
2.2.1.2.9 Hardware	41
2.2.1.2.10 Restricciones en el diseño y la implementación.....	42
2.2.1.2.11 Rendimiento	42
2.2.2 Modelo de Casos de Uso del Sistema.....	42
2.2.2.1 Definición de actores.....	43
2.2.2.2 Diagrama de Casos de Uso	44
2.2.2.3 Descripción textual de los Casos de Uso	44
Conclusiones	48
CAPÍTULO 3: DISEÑO DEL SISTEMA	49
3.1 Modelo de Diseño.....	49
3.1.1 Fundamentación del uso de Patrones	49
3.1.1.1 Modelo Vista Controlador.....	50
3.1.1.2 Alta Cohesión	50

Tabla de Contenidos

3.1.1.3 Bajo Acoplamiento	51
3.1.1.4 Fachada.....	51
3.1.2 Definición de la estructura del diseño.....	52
3.1.3 Diagramas de Clases del Diseño.....	53
3.1.4 Descripción de las clases y atributos.....	60
3.1.4.1 Descripción de páginas clientes.....	60
3.1.4.2 Descripción de páginas servidoras.....	61
3.1.4.3 Descripción de métodos del negocio.....	64
3.1.5 Diagrama de Clases Persistentes.....	68
3.1.6 Modelo de Datos	69
Conclusiones	73
CAPÍTULO 4: IMPLEMENTACIÓN	74
4.1 Modelo de Implementación	74
4.1.1 Diagrama de Componentes	75
4.1.2 Diagrama de Despliegue.....	77
4.2 Descripción de los Métodos (agentes o métodos más complejos).....	78
4.3 Estándares de diseño, codificación y tratamiento de errores	78
4.3.1 Estándares de Diseño	78
4.3.2 Estándares de Codificación	81
4.3.3 Tratamiento de Errores	83
Conclusiones	85
CONCLUSIONES	86
RECOMENDACIONES	87
REFERENCIAS BIBLIOGRÁFICAS	88
BIBLIOGRAFÍA.....	92
ANEXOS.....	94
GLOSARIO DE TÉRMINOS	101

INTRODUCCIÓN

El estado cubano ha asumido totalmente la responsabilidad de la atención a la salud de sus ciudadanos, llevando a cabo acciones económicas y médicas para asegurar la protección de la salud, partiendo entre otras, de los principios de gratuidad y accesibilidad total a los servicios con la calidad requerida.

El constante desarrollo manifestado por la microelectrónica, la informática y las telecomunicaciones ha dado lugar al surgimiento de lo que se conoce hoy en día como Tecnologías de la Información y las Comunicaciones (TIC), las cuales en un proceso acelerado de convergencia se insertan en diversos ámbitos de la vida humana. Cada vez son más las instituciones u organizaciones que optan por incorporar aplicaciones que gestionen su información, priorizando esferas importantes como la educación, la salud y la industria, para lograr una mayor dinámica en sus procesos de negocio.

En el año 1997, en el marco del proceso de informatización de la sociedad cubana, se realiza la planificación estratégica del sector de la salud a mediano plazo, cuya primera etapa abarca hasta el año 2000. En ese período, 1998-2000, se definen los objetivos estratégicos maestros y se elaboran los planes de acción para la capacitación y perfeccionamiento, la Seguridad Informática y la Informatización de los Servicios. [1]

El Ministerio de Salud Pública (MINSAP), organismo rector del Sistema Nacional de Salud (SNS), ha definido un grupo de premisas y requisitos para informatizar sus procesos. A partir del año 2001 se realizaron profundos cambios en métodos y estilos de trabajo que propiciaron un salto cualitativo en el desarrollo estratégico de la informatización. Actualmente se trabaja en proyectos con el empleo de tecnologías basadas en Internet, software libre, documentación de todo el proceso productivo, requisitos de seguridad del software, independencia de las bases de datos y desarrollo en multiplataforma, utilizando estándares internacionales para los productos relacionados con la salud que ayuden a fomentar la producción de soluciones más estables y eficaces, teniendo como infraestructura la Red Telemática de la Salud. [2]

Acorde a las estrategias trazadas por el MINSAP, surge el Proyecto Atención Primaria de la Salud (APS), que encamina sus objetivos a analizar, diseñar y desarrollar un producto de software para dar respuesta a

los procedimientos establecidos por el SNS para el nivel de atención primaria, tomando al policlínico como centro de gestión de información y hacer realidad las ideas del Comandante en Jefe, Fidel Castro cuando expresa: “Una profunda revolución en los servicios de salud tendrá lugar en nuestra Patria”. [3]

La Empresa de Soluciones Informáticas SOFTEL, entidad del Ministerio de Informática y las Comunicaciones (MIC), en el año 2003 reorienta su trabajo hacia el desarrollo de productos y servicios informáticos que eleven la eficiencia del sistema de salud cubano y se ganen un espacio en el mercado internacional. SOFTEL se inserta en la Infraestructura Productiva (IP) de la Universidad de las Ciencias Informáticas (UCI) y se vincula a estudiantes y profesores de la Facultad 7, con el objetivo de cumplir la misión de informatizar el SNS, en el marco del Proyecto APS.

Como parte de este proyecto se concibe el Sistema de Información para la Salud (SISalud), que permitirá integrar la información de diferentes áreas de manera sencilla, pero eficiente; utilizando una arquitectura orientada a servicios y basada en componentes. SiSalud está compuesto por los módulos de los diferentes niveles de atención médica: primaria, secundaria y especializada, y por el Registro Informatizado de Salud (RIS) que incluye los nomencladores nacionales que brinda servicios a las demás aplicaciones informáticas, para obtener una información integral en tiempo real que facilite la toma de decisiones en los diferentes niveles de dirección, la docencia, la investigación y la gestión en salud [4], siendo uno de estos módulos el Registro de la Clasificación Internacional de Enfermedades y Problemas Relacionados con la Salud (RCIE).

La Clasificación Internacional de Enfermedades (CIE), promovida por la Organización Mundial de la Salud (OMS), constituye un estándar de diagnóstico para todos los propósitos epidemiológicos generales y otros de administración, con el fin de apoyar al análisis de la situación general de salud de los grupos de población, el seguimiento de la incidencia y la prevalencia de enfermedades, entre otras variables.

La CIE se da a conocer en agosto de 1900, siendo desde entonces revisada, actualizada y editada cada cierto tiempo. La misma, en la actualidad, se encuentra en su Décima Revisión denominada Clasificación Estadística Internacional de Enfermedades y Problemas Relacionados con la Salud (CIE-10), brindando la posibilidad de recopilar información estadística uniforme para el análisis y la interpretación y comparación de los datos de mortalidad y morbilidad recolectados en diferentes países o áreas.

Cuba comienza a aplicar la CIE-10 a partir del año del 2000, para la emisión de Estadísticas de Mortalidad y a partir del 2001 para las Estadísticas de Morbilidad [5]. Sin embargo no se ha generalizado su uso, producto de la escasez de materiales relacionados con la misma, que deben ser comprados y no se pueden reproducir por estar amparados por el Protocolo 2 de la Convención Universal de Derecho de Autor. En la carrera de Medicina no profundiza en la importancia del uso de dicha clasificación como base para la futura homogenización de los diagnósticos en la salud. Todo lo anterior, influye en el desconocimiento y poco uso de este estándar, por parte del personal de salud, trayendo como consecuencia dificultades en la generación de estadísticas en los diferentes niveles definidos en el SNS.

Este proceso no está apoyado por las ventajas que aporta una aplicación informática integrada a la Red de Salud Nacional, por lo que resulta complejo para los especialistas de las Direcciones de Estadísticas y Registros Médicos la homologación de la información por la CIE-10, la que se realiza de forma manual a partir del trabajo con libros de gran volumen de información y de difícil manipulación.

Por la importancia en la proporción de confiabilidad y apoyo en la toma de decisiones que ofrecen las estadísticas en la salud, se plantea como **Problema a resolver**: ¿Cómo desarrollar un sistema automatizado para la gestión de la información de la Clasificación Estadística Internacional de Enfermedades y Problemas Relacionados con la Salud, que permita de forma estándar y homogénea el proceso de diagnóstico en cada una de las aplicaciones informáticas del Sistema Nacional de Salud?

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

El cual tiene como **Campo de Acción** el proceso de gestión de la información de la Clasificación Estadística Internacional de Enfermedades y Problemas Relacionados con la Salud.

Por lo tanto, el **Objetivo General de la Investigación** es desarrollar una aplicación web que automatice la gestión de la información de la Clasificación Estadística Internacional de Enfermedades y Problemas Relacionados con la Salud y brinde la posibilidad de llevar a cabo un proceso de diagnóstico homogéneo en cada una de las aplicaciones informáticas del Sistema Nacional de Salud.

Planteando como **Idea a defender** que el desarrollo de una aplicación web que automatice la gestión de la información de la Clasificación Estadística Internacional de Enfermedades y Problemas Relacionados con

la Salud, garantizará el diagnóstico homogéneo de las enfermedades y problemas de salud y con ello la emisión de estadísticas estándares en todos los niveles de salud.

Trazando para ello las siguientes **Tareas de la Investigación:**

1. Investigar los conceptos fundamentales asociados al dominio del problema a resolver.
2. Analizar la Arquitectura definida por el MINSAP, Orientada a Servicios y Basada en Componentes (SOA_CBA), para el desarrollo de sus aplicaciones: Plataforma de Servicios PlaSer y Registro Informatizado de Salud (RIS).
3. Identificar y analizar la arquitectura de integración con otros componentes o partes del SNS dentro o fuera de este.
4. Documentar los Flujos de Trabajo:
 - Modelamiento del Negocio
 - Requerimientos
 - Diseño
 - Implementación
5. Describir las herramientas y tecnologías informáticas utilizadas para la implementación.
6. Implementar los componentes (servicios web) del Módulo RCIE.

En sentido general, se puede destacar que con el desarrollo de RCIE se espera obtener un grupo de **beneficios**, los cuales se mencionan a continuación:

- Posibilitará obtener información fiable para las investigaciones científicas y mejorar las condiciones de trabajo para el personal de la salud.
- Facilitará el uso estandarizado de la CIE-10 y la correcta toma de decisiones a todos los niveles del SNS.
- Contribuirá a la elaboración de indicadores e información estadística necesarios para evaluar los resultados en términos de salud y apoyar la vigilancia.
- Permitirá el acceso concurrente de usuarios.
- Facilitará la gestión (inserción, modificación o eliminación) de la información según la estructura de la CIE-10.

- Proporcionará información solicitada, a través de servicios web, a otros módulos que se relacionen con él.

El documento presenta una estructura por capítulos como se muestra a continuación:

CAPÍTULO 1. Fundamentación Teórica: Se describen los conceptos fundamentales asociados al dominio del problema, se expone un estado del arte del tema tratado, tanto a nivel nacional como internacional, se hace un análisis crítico entre las soluciones ya existentes y la propuesta. Explica y justifica, además, las tendencias, tecnologías y herramientas en las que se apoya la solución al problema.

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

CAPÍTULO 3 Diseño del Sistema: En este capítulo se justifican los patrones a usar en el diseño de la aplicación, se definen la estructura y los elementos del diseño, se muestran los diagramas de clases de los casos de uso del sistema y el modelo de datos. Conformándose, finalmente, el Modelo de Diseño, el cual constituye una base para la futura implementación.

CAPÍTULO 4. Implementación: En este capítulo se describe la implementación del sistema propuesto a través de los diagramas de componentes y la distribución del mismo mediante el diagrama de despliegue. Se muestra una descripción detallada de los métodos más complejos en el desarrollo de la solución.

El documento cuenta además con las conclusiones del trabajo, recomendaciones a tener en cuenta para dar continuidad a la investigación y los anexos que permitirán una mejor ilustración del estudio realizado.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

Introducción

El objetivo de este capítulo es abordar los antecedentes de la Clasificación Estadística Internacional de Enfermedades y Problemas Relacionados con la Salud, en su Décima Revisión (CIE-10) y algunos intentos de automatización de la misma. Se expone el marco teórico y conceptual asociado a la problemática a resolver, la estructura organizativa del Sistema Nacional de Salud; así como, el estado del arte de las tecnologías, metodologías y herramientas a utilizar en la presente investigación.

1.1 Sistema Nacional de Salud (SNS)

El sistema nacional de salud como concepto, en cualquier país, es el conjunto de formas y métodos que sirven de base para la organización de la atención a la salud en un país determinado. [6]

Con el triunfo de la Revolución en el año 1959 se produjeron profundas transformaciones sociales, políticas y económicas que permitieron enfrentar los problemas sanitarios existentes y elevar sustancialmente los niveles de vida y salud de la población. En estos primeros años se determinó que era necesario ampliar y perfeccionar el sistema nacional de salud creado, el de atención médica y hospitalaria, desarrollar la medicina preventiva, impulsar la medicina rural, incrementar los estudios de medicina del trabajo y su aplicación al tratamiento de enfermedades profesionales, elevar el nivel de la cultura sanitaria del pueblo y poner énfasis en preservar el medio ambiente y los bienes naturales. Asimismo se plantean los presupuestos metodológicos que configuran a la Escuela Cubana de Medicina, estableciendo la prevención como concepto primordial del sistema sanitario en el cuidado de la salud. [7]

También en los primeros años del gobierno revolucionario se orientó la creación de centros científicos para la atención sistemática de las actividades de la ciencia, cuyas investigaciones respondieran a las necesidades del país, a corto y largo plazo; se trazaron los lineamientos generales y se aseguraron los recursos materiales y humanos para el éxito de estas tareas.

Un gran aporte de la Revolución en salud pública lo constituye la introducción del concepto de universalización de la docencia médica, al integrar a los estudiantes de Medicina y Enfermería durante su

Capítulo 1: Fundamentación Teórica

proceso de formación a las unidades asistenciales docentes, lo que ha permitido, además, alcanzar la masificación de los programas de formación de los recursos humanos en la esfera sanitaria. [8]

En el año 2002, con una población que sobrepasaba los 11 millones de habitantes, el cuadro de salud cubano cambió radicalmente en poco más de cuatro décadas de quehacer a favor de la salud de la población: la mortalidad general no se produce por las llamadas “enfermedades de la pobreza”, sino, como en los países altamente desarrollados, las principales causas la constituyen las afecciones del corazón, el cáncer y los accidentes vasculares, por lo que resulta de gran importancia el diagnóstico médico a partir del uso de estándares, para contar con datos uniformes que ofrezcan una información estadística codificada que permita el análisis, la interpretación y la comparación de los datos emitidos por las diferentes unidades de salud del país. [9]

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

En el 1997 la alta dirección del MINSAP propuso: "desarrollar en el Sistema Nacional de Salud una política única con un modelo integral de informatización a los diferentes niveles del mismo, así como el acceso a la información como proceso que apoye y potencie decisivamente la asistencia médica, la docencia, la investigación, la higiene y la epidemiología, la industria médico farmacéutica, la economía y administración de salud, que se extienda de forma integral a todas las instituciones del país para alcanzar un Sistema Integrado de Gestión que será herramienta básica en la materialización de las estrategias y programas de Salud". [10]

1.1.1 Informatización del Sistema Nacional de Salud

La informatización del SNS comprende un conjunto de métodos, técnicas, procedimientos y actividades gerenciales dirigidas al manejo de la información en salud, la cual comprende la información sobre el estado de salud de la población, la información sobre el conocimiento de las ciencias de la salud y la información en general para la toma de decisiones, clínico-epidemiológicas, operativas y estratégicas. [11]

Capítulo 1: Fundamentación Teórica

El Sistema Nacional de Salud tiene como centro mismo al Policlínico y se propone que las Instituciones del país alcancen un elevado nivel de informatización de las actividades que brindan, partiendo del Sistema de Atención Primaria, para lograr un incremento de la calidad, efectividad y eficiencia de los servicios que se presten a la población, contribuyendo al logro de la satisfacción de los prestadores y usuarios del Sistema Nacional de Salud.

Este proceso de informatización responde a las Políticas y Principios Socialistas cuya actividad es una estrategia vital y prioritaria del Sistema Nacional de Salud que responde a estrategias, planes de desarrollo, políticas de estandarización y proyectos que serán aprobados centralmente, ya que todos los productos y servicios se integrarán a la ciberinfraestructura del sector y se realizarán en lo fundamental sobre sistemas abiertos, arquitectura orientada a los servicios y basadas en componentes (SOA-CBA), utilizando software libre y de calidad. Deben constituirse en componentes modulares y estables, que compartan normas y cooperen entre sí. [12]

Los proyectos que se coordinen con otros organismos y entidades dentro o fuera del país están en la obligación de reconocer y cumplir las políticas e intereses del MINSAP y admitir la evaluación, control y certificación de las soluciones informáticas para el sector de la salud pública.

Entre los principales impactos esperados con la Informatización del SNS se pueden mencionar: [13]

Para la Población:

- Equidad distribuida de acceso a servicios, tecnologías e información de salud independientemente de áreas geográficas, ni niveles de atención.
- Disfrutará la sensación de ser atendida por un personal médico mejor preparado y actualizado, elevando su confianza hacia el sistema de atención.
- Reducción del número de desplazamientos innecesarios entre instituciones de salud con el consecuente impacto en su vida social.
- Reducción de tiempos de esperas para el acceso a servicios especializados con la posibilidad de recibirlos en su propio escenario social.

Capítulo 1: Fundamentación Teórica

Para el SNS:

- Gestión oportuna de una información confiable y actualizada que propiciará una optimización considerable de recursos.
- Elevación de la capacidad y calidad de la toma de decisiones asistenciales y gerenciales por la disposición oportuna de información actualizada para todos los niveles del SNS, que permitirá una rápida transferencia de la información sanitaria de un paciente.
- Disponer de un soporte y herramientas poderosas para la formación y actualización constante de sus miembros desde sus propios escenarios de desempeño, potenciando la investigación científica multicéntrica, nacional e internacional.
- Elevará el papel del Médico y Enfermera de la Familia, incrementando su nivel científico y profesional.

SOFTEL, Empresa del Ministerio de la Informática y las Comunicaciones (MIC), tiene como misión generar soluciones informáticas e implementar un sistema de excelencia para el desarrollo y mantenimiento de productos de software especializados en salud y además organizar un esquema para la prestación de los servicios informáticos a dicho sector.

En la actualidad, SOFTEL, utiliza una estrategia nunca antes concebida en el país en un proceso de desarrollo de software, con una organización del proceso productivo a través de una eficiente gestión de requerimientos, donde participan desde un inicio, médicos y trabajadores de la salud, vinculados directamente a la Atención Primaria en calidad de expertos funcionales y en estrecho vínculo con los especialistas de Informática.

Ejecuta estos objetivos en colaboración con la Universidad de las Ciencias Informáticas (UCI) vinculados a la Facultad 7 que se especializa en software para la salud, logrando incorporar a profesores líderes de proyectos y estudiantes, desde los primeros años de estudio a la producción. De esta integración surge el Proyecto Atención Primaria de la Salud (APS) que encamina sus objetivos a analizar, diseñar y desarrollar un producto de software para dar respuesta a los procedimientos establecidos por el SNS para el nivel de atención primaria.

Capítulo 1: Fundamentación Teórica

1.1.2 Infomed

Es la Red Telemática de Salud en Cuba, que parte de la existencia de una red nacional especializada de centros de información. Surge en 1992 la necesidad de desarrollar las redes académicas, soportadas en la Tecnología de Información como alternativas de acceso a la información relacionada con las ciencias médicas y como una acertada visión acerca de la positiva influencia que podría tener el uso de las nuevas tecnologías para mejorar el acceso a la información de los trabajadores de la salud en Cuba. Su misión es desarrollar una red integrada de acceso y gestión de la información y el conocimiento de salud en Cuba que contribuya al mejoramiento de la atención clínica, la docencia, la investigación y la gestión de salud. [14]

1.1.3 Registro Informatizado de Salud (RIS)

Hasta la fecha la informatización de la salud cubana no ofrece un mecanismo único de integración de los sistemas de información desarrollados, estos en la actualidad se presentan como componentes aislados que brindan solución a determinados problemas, tratándolos como islas de información, por lo que no permite disponer de información única, confiable y en tiempo real para la toma de decisiones en los diferentes niveles de dirección.

El Registro Informatizado de Salud (RIS), sentó las bases para la existencia de un sistema formado por componentes desarrollados con un nivel de cohesión y acoplamiento que le permiten ser capaces de interactuar entre ellos y de esta forma reutilizar la información gestionada por cada componente.

Por definición el RIS: “Es la solución informática integral para la Salud Pública, acorde con los objetivos de la informatización de la sociedad cubana. Constituido por un conjunto de aplicaciones independientes (módulos del sistema) que se interconectan según las necesidades del flujo de información”. [15]

El proyecto APS hereda las características del RIS, y se caracteriza por ser un sistema distribuido de componentes distantes geográficamente, en constante interacción a través de la Red Telemática de Salud de Cuba, INFOMED.

Capítulo 1: Fundamentación Teórica

Por tanto, es necesario desarrollar una arquitectura que garantice la máxima disponibilidad de cada uno de sus componentes y permita la recuperación del sistema ante posibles fallos de conectividad o resolver problemas tales como la recuperación de la información, independientemente de su ubicación. [16]

1.1.4 Solución propuesta para la Informatización del SNS

Para lograr la Informatización en este sector se pretende que todos los componentes, servicios o sistemas que se desarrollen estén integrados en una aplicación nombrada Sistema de Información para la Salud (SISalud), que permita que las aplicaciones informáticas suministren la información adecuada para la atención de pacientes, el control y la administración, y suministre a los niveles de dirección de las unidades de salud y los superiores la información requerida para viabilizar la toma de decisiones, cumpliendo con las indicaciones técnicas y metodológicas que se establezcan centralmente.

Este sistema estará compuesto por el Registro Informatizado de Salud (RIS), el Sistema Informatizado de Atención Primaria (SIAP), el Sistema Informatizado de Gestión Hospitalaria (SIGH) y el Sistema Informatizado de Atención Especializada (SIAE). [17]

Sistema Informatizado de Atención Primaria (SIAP): Se incluirán en la etapa actual los módulos propios de este nivel de atención, que constituirán una nueva herramienta para la transformación de los servicios que se brinda en este nivel.

Sistema Informatizado de Gestión Hospitalaria (SIGH): Se agruparán aquí los módulos que pertenecen al nivel de atención secundario u hospitalario.

Sistema Informatizado de Atención Especializada (SIAE): Se agruparán aquí los módulos que pertenecen al nivel de atención terciario o especializado.

Registro Informatizado de Salud (RIS): Formado por los registros o módulos que son administrados o gestionados a nivel nacional o central.

Capítulo 1: Fundamentación Teórica

1.2 Clasificación Internacional de Enfermedades (CIE)

Para el desarrollo de cualquier aplicación es necesario la definición y adopción de estándares que permitan la integración y homogeneidad de los sistemas, esclareciendo que un estándar es toda regla, requisito o recomendación basada en principios ya probados y en la práctica. Desde la perspectiva de los estándares de atención de salud, esta área se encuentra en flujo constante y se hace necesario estar atento a la evolución de las recomendaciones de los organismos técnicos internacionales y nacionales y las organizaciones profesionales que trabajan en la investigación sobre estándares. La coordinación de este sector y la adopción de estándares comunes para los usuarios, los fabricantes y los proveedores de servicios tienden a fomentar la producción de soluciones más estables y eficaces en función de los costos. [18]

En el sector de la salud existen varias categorías de información, cada una de ellas con estándares propios como se muestra: [19]

Estándares del identificador: Se subdividen en paciente, proveedor, punto de atención y producto. No es sorprendente que no haya aceptación o satisfacción universal con estos sistemas.

Estándares de comunicaciones (formato del mensaje): Aunque los estándares en esta área se encuentran todavía en diversas etapas en el proceso de formulación, están en general en una etapa más avanzada que los estándares de otros grupos.

Estándares de contenido y estructura: El trabajo en esta área está dirigido principalmente a la formulación de estándares para el diseño del registro computadorizado de pacientes y registros dentales.

Estándares internacionales: La Organización Internacional de Normalización (ISO) es una federación mundial de organizaciones de estándares nacionales. Cuenta con 90 países miembros y su finalidad es promover la normalización y las actividades relacionadas en el mundo. Con este objetivo, muchas organizaciones, comités y subgrupos promueven la evolución de los estándares de atención de salud en todo el mundo.

Representaciones de datos clínicos (códigos): Se utilizan en gran medida para documentar diagnósticos y procedimientos. Son más de 150 los sistemas de codificación conocidos, entre los que se encuentra el sistema para la Clasificación Internacional de Enfermedades, promovido por la OMS.

Capítulo 1: Fundamentación Teórica

En la práctica, la CIE se ha convertido en una clasificación diagnóstica internacional, estándar para todos los propósitos epidemiológicos generales y muchos otros de administración de salud. Esto incluye el análisis de la situación general de salud de grupos de población y el seguimiento de la incidencia y prevalencia de enfermedades y otros problemas de salud en relación con otras variables, tales como las características y circunstancias de los individuos afectados. [20]

1.2.1 Antecedentes históricos e importancia de la CIE

La necesidad de registrar y analizar el comportamiento de las enfermedades y de las defunciones según sus causas, ha sido una preocupación constante de la humanidad desde hace muchos siglos. El conocimiento de su frecuencia, su distribución por región, edad, sexo y algunas características socioeconómicas asociadas, así como sus cambios en el tiempo, es usado principalmente para investigar los factores que causan las enfermedades y las muertes, evaluar la situación de salud de la población en diversos períodos, evaluar los resultados de los programas de salud aplicados, planear los servicios de salud que requiere la población y aplicar medidas para prevenir las enfermedades y las muertes prematuras. [21]

Con este propósito el Instituto Internacional de Estadística, encargó a un comité dirigido por el sociólogo francés Jacques Bertillon, en una reunión efectuada en Viena en 1891, la preparación de una clasificación de causas de defunción. Esta clasificación recibió aprobación general y fue acogida por varios países y numerosas ciudades. Su primera aplicación fue en América del Norte y ya en 1898 la Asociación Estadounidense de Salud Pública, recomendó que los registros civiles de Canadá, México y Estados Unidos, adoptaran la Clasificación de Bertillon y sugirió además su revisión cada 10 años. En agosto de 1900 se dispuso de la primera revisión de la Lista Internacional de Causas de Defunción de Bertillon. [22]

Desde entonces la Clasificación Internacional de Enfermedades ha sido revisada, actualizada y editada con el propósito de facilitar el proceso de comparabilidad global. Desde la Sexta Revisión, aprobada en 1948, cada 10 años aproximadamente, la OMS convoca a los países miembros a aprobar una nueva revisión. [23] (Ver Anexo 1)

Capítulo 1: Fundamentación Teórica

La Décima Revisión (CIE-10) está estructurada de la siguiente forma: [24]

Volumen 1. Lista tabular. Contiene el informe de la Conferencia Internacional para la Décima Revisión, la clasificación de tres y de cuatro caracteres, la clasificación para la morfología de los tumores, listas especiales de tabulación para la mortalidad y la morbilidad, las definiciones y el reglamento de nomenclatura.

Volumen 2. Manual de instrucciones. Se agrupan aquí las recomendaciones para la certificación y clasificación anteriormente incluidas en el Volumen 1, y se facilita una buena cantidad de información nueva, así como instrucciones y orientaciones sobre la manera de usar el Volumen 1, sobre las Tabulaciones y sobre la planificación necesaria para emplear la CIE, todo lo cual hizo falta en las revisiones anteriores. Figura aquí también la información histórica que anteriormente formaba parte de la introducción al Volumen 1.

Volumen 3. Índice alfabético. Además del propio índice y su correspondiente introducción, se dan instrucciones amplias sobre cómo utilizarlo.

Cada volumen contiene además orientaciones específicas y otros aspectos relacionados con la CIE y las estadísticas derivadas, los cuales se precisarán al referirse a cada uno de ellos. El codificador debe revisar estos volúmenes para conocer su contenido y aprender su manejo en la codificación. [25]

El diseño de la CIE ha permitido la elaboración de otros sistemas de clasificación relacionados con la atención a la salud, los cuales tienen principios estructurales y funcionales semejantes.

Como ejemplo de ello se tiene la Clasificación Internacional de Procedimientos en Medicina, la Clasificación Internacional de Deficiencias, Discapacidades y Minusvalías, la Clasificación Internacional de Enfermedades para Oncología y otras más que, junto con ella, forman parte de lo que ahora se conoce como familia de clasificaciones, de la cual la CIE es el centro. [26] (Ver Anexo 2)

Capítulo 1: Fundamentación Teórica

1.2.2 Ejemplos del uso de la CIE-10 en el mundo. Intentos de informatización

✓ España

En Cataluña-España, el uso que hacen de la CIE se desarrolla más en el campo clínico, por tanto los problemas diagnósticos también surgen de este uso. Utilizan la CIE-10 para codificar diagnósticos desde la consulta de atención primaria. Dado que el diagnóstico lo realiza el propio clínico, durante la consulta, con el paciente delante, han tenido que solucionar varios problemas para agilizar y adaptar el uso de la clasificación. Se ha tenido que traducir la clasificación al catalán, buscar las equivalencias entre algunos diagnósticos de atención primaria y las definiciones de la CIE-10. En muchas ocasiones los descriptivos de los diagnósticos no facilitan su fácil entendimiento y uso en el seguimiento diario de los pacientes en la consulta. [27]

✓ Colombia

Actualmente en Colombia, el Departamento Administrativo Nacional de Estadística (DANE), conjuntamente con el Ministerio de Salud, están trabajando en un proyecto de mejoramiento tecnológico del Sistema de Estadísticas Vitales, como un componente de un gran sistema de información nacional en Salud. Se pretende realizar la certificación de los nacimientos y las defunciones utilizando formularios electrónicos, lo cual implica la eliminación de los certificados físicos. En lo que respecta al diligenciamiento de las causas de defunción, se espera que el médico seleccione los diagnósticos a través de un software que le permita ir buscando, puede ser por sistemas anatómicos o por enfermedades de aparatos o por capítulos de la CIE-10, de manera que quede seleccionado el diagnóstico y el código correspondiente. Por otra parte, se pretende que el software seleccione automáticamente la causa básica. [28]

✓ TENDON

Es un sistema de auto enseñanza asistida por computadora de la CIE-10 desarrollado por el centro colaborador de la Organización Mundial de la Salud en Londres, Inglaterra. Este paquete sirve para entrenamiento básico de la CIE-10 y una reorientación para codificadores familiarizados con la CIE-9. El software fue desarrollado por Dean Associates of Sheffield y está diseñado para dar completo entendimiento de los conceptos, importancia y métodos de la acertada codificación del diagnóstico, la Clasificación Internacional de Enfermedades y los Problemas Relacionados con la Salud, en su Décima

Capítulo 1: Fundamentación Teórica

Revisión (CIE-10), así como para apoyar la codificación de la morbilidad y mortalidad utilizando la CIE-10. [29]

✓ INTERCOD

Es un sistema de auto-enseñanza asistida por computadora para la Clasificación Internacional de Enfermedades en su Décima Revisión desarrollado por la Organización Panamericana de la Salud. El paquete se desarrolló con la colaboración de la Universidad de Guadalajara, la Dirección General de Estadística e Informática de México y los Centros Colaboradores de CIE de OMS/OPS. Presenta como características la lista tabular de los 15,342 códigos de la CIE-10 en formato PDF, multi-idioma (español, inglés, portugués y francés), decenas de ejercicios para practicar la codificación y sección de evaluación. [30] (Ver Anexo 3)

✓ Argentina

La empresa SoftMédico, Buenos Aires, ha diseñado varias aplicaciones de libre distribución basadas en la CIE:

- CIE-CIAP es una aplicación de escritorio desarrollada en Microsoft Access 2002. Permite buscar a partir de un segmento de texto, los códigos de diagnóstico de la CIE-10 y requiere para su ejecución la instalación del software Access XP, pues no es compatible con versiones anteriores. [31] (Ver Anexo 4)
- CIE-CIAP FULL es una versión más avanzada de CIE-CIAP, la cual presenta un mejor entorno de interfaz, haciendo más amigable la aplicación al usuario, permite búsquedas de diagnósticos y procedimientos dentro de la CIE-9 y cuenta con versiones en español e inglés. [32] (Ver Anexo 5)

1.2.3 La Clasificación Internacional de Enfermedades en Cuba

Cuba adopta la Lista de Bertillon a partir de enero de 1901, recién realizada la primera revisión en Francia en el año 1900. Desde entonces, se han aplicado las sucesivas revisiones y mantenido un reconocido prestigio en el campo de la Certificación de la Muerte y en las Estadísticas de Mortalidad.

Capítulo 1: Fundamentación Teórica

En enero del 2000, el país comienza a aplicar la CIE-10 en las Estadísticas de Mortalidad y un año más tarde, en el 2001, introduce la aplicación de la clasificación en Estadísticas de Morbilidad. [33]

Para apoyar este proceso, la Dirección Nacional de Estadísticas y Registros Médicos utiliza una Multimedia de la CIE-10 que permite realizar búsquedas por distintos criterios devolviendo la respuesta deseada. En algunos hospitales del país, se emplean sistemas creados por las propias entidades en apoyo al proceso de diagnóstico por la CIE-10 dentro de los mismos. El sitio web de Estadísticas de Salud de Cuba, ubicado en la Red Telemática de Salud, brinda la posibilidad de conocer la CIE-10 por su estructura y taxonomía.

1.3 Flujo actual de los procesos involucrados en el campo de acción

En la actualidad el trabajo de codificar las enfermedades y los problemas relacionados con la salud lo realizan los Especialistas de las Direcciones de Estadística y Registros Médicos del MINSAP:

- El Director de la Dirección Nacional de Estadística es el responsable de adquirir los libros de la CIE y controlar las actualizaciones dentro del país.
- Los Especialistas de la Dirección Nacional de Estadística y Registros Médicos reciben las modificaciones de la CIE-10 analizadas y aprobadas por el grupo internacional de expertos de la OMS; realizan las modificaciones (de forma manual sobre el libro) mientras esté vigente la revisión de la CIE. Son los encargados de informar a las Direcciones Provinciales de Estadística y Registros Médicos de las modificaciones, por teléfono o a través de un taller nacional. Informar sobre posibles errores al asesor regional de la Organización Panamericana para la Salud (OPS) o al Centro Colaborador.
- Los Especialistas de las Direcciones Provinciales de Estadística y Registros Médicos reciben las orientaciones de las modificaciones realizadas, las llevan a cabo e informan sobre posibles errores al Especialista principal de la Dirección Nacional de Estadística y Registros Médicos.

1.4 Situación Problemática

La existencia de clasificaciones internacionales de enfermedades contribuyen a que las estadísticas de salud obtenidas por los eventos que ocurren en los individuos y/o en el proceso salud/enfermedad puedan

Capítulo 1: Fundamentación Teórica

ser homogéneas de forma global y que la información manejada sea más entendible y segura. La CIE-10 es reconocida como la principal clasificación para el diagnóstico de los procesos que le ocurren a las personas, la existencia de otras la retroalimentan.

Esta clasificación, existe en varios formatos dentro del país, pero no son extensibles a toda la comunidad médica. Los libros, por ejemplo, deben ser comprados y no se pueden reproducir por estar amparados por el Protocolo 2 de la Convención Universal de Derecho de Autor. Si se argumenta además, que no se profundiza en la importancia de su conocimiento en la docencia de pregrado, la especialidad y cursos de postgrado de la carrera de medicina, se puede deducir a modo general que existe desconocimiento, por parte del personal de la salud, de dicha clasificación, lo que trae consigo que existan limitantes para el diagnóstico por la CIE-10 en los diferentes niveles asistenciales.

La no existencia de diagnósticos homogéneos, dificulta y hace complejo el trabajo de los Especialistas de Estadística de Salud, ya que estos deben homologar por la clasificación haciendo búsquedas manuales en los libros para emitir información estándar a la Dirección Nacional. De la misma forma se modifica manualmente, sobre el primer volumen, al ser propuesta una nueva actualización por la OMS y este cambio es informado posteriormente a las direcciones provinciales, procesos que no son apoyados por las ventajas brindadas por una aplicación informática, integrada a la Red de Salud Nacional, que gestione la información de la CIE-10 y brinde servicios a las demás.

1.5 Objeto de automatización e información manipulada

Teniendo en cuenta los problemas existentes en el país referente a la gestión de la información de la CIE-10 en el Sistema Nacional de Salud, se propone el desarrollo de una aplicación que sea capaz de:

Gestionar la información referente a la taxonomía de la CIE-10: Eliminar, modificar o insertar Capítulo(s), Grupo(s), Categoría(s), Subcategoría(s) o Problema(s) de Salud, posibilitando la actualización de la información con vistas a erradicar el difícil trabajo manual y la descentralización de la misma.

Permitir la realización de búsquedas dinámicas por diferentes criterios, que posibilite a todo profesional que de una forma u otra emita algún diagnóstico, pueda buscar de forma rápida información en la misma, facilitando así la estandarización y homogeneidad de las enfermedades y problemas relacionados con la

Capítulo 1: Fundamentación Teórica

salud en todos los niveles asistenciales. Además de permitir la impresión de la información resultante de las opciones descritas anteriormente que pueda resultar útil como material de apoyo docente y para el aprendizaje de la estructura de la CIE-10.

Brindar servicios a las restantes aplicaciones integradas a la Red Telemática de Salud, contribuyendo a impulsar la informatización de la sociedad cubana, acelerar y mejorar las posibilidades de estudios de tipo comparativo, investigativos y estadísticos, así como el diagnóstico y la toma de decisiones.

La información que se gestionará mediante la aplicación será la contenida en el Volumen 1 de los libros de la CIE-10, 1ra edición, 2da reimpresión.

1.5.1 Análisis comparativo con otras soluciones existentes

Las soluciones existentes a nivel internacional y nacional, no resuelven el problema planteado, debido a que son aplicaciones de escritorio, con arquitecturas monolíticas, por tanto la información no se encuentra centralizada de manera que pueda ser accesible de cualquier lugar. En el sitio de Estadísticas de Salud de Cuba, aunque la información este centralizada, no permite actualizaciones, búsquedas, ni brinda servicios a las demás aplicaciones de la red de salud.

1.6 Tendencia y tecnologías a considerar

XML

XML, es el estándar de Extensible Markup Language (Lenguaje Extensible de Marca). No es más que un conjunto de reglas para definir etiquetas semánticas que nos organizan un documento en diferentes partes. Es un metalenguaje que define la sintaxis utilizada para definir otros lenguajes de etiquetas estructurados. [34]

Su desarrollo se comenzó en 1996 y la primera versión salió a la luz el 10 de febrero de 1998. La primera definición que apareció fue: Sistema para definir, validar y compartir formatos de documentos en la web.

Durante el año 1998 XML tuvo un crecimiento exponencial, en medios de comunicación, menciones en páginas web, soporte software, etc, siendo sus objetivos los siguientes: [35]

Capítulo 1: Fundamentación Teórica

- XML debe ser directamente utilizable sobre Internet.
- XML debe soportar una amplia variedad de aplicaciones.
- Debe ser fácil la escritura de programas que procesen documentos XML.
- El número de características opcionales en XML debe ser absolutamente mínima, idealmente cero.
- Los documentos XML deben ser legibles por humanos y razonablemente claros.
- El diseño de XML debe ser preparado rápidamente.
- El diseño de XML debe ser formal y conciso.
- Los documentos XML deben ser fácilmente creables.
- La concisión en las marcas XML es de mínima importancia.

Entornos Distribuidos. Modelo Cliente/Servidor

Es una arquitectura de procesamiento cooperativo donde uno de los componentes pide servicios a otro. Es un procesamiento de datos de índole colaborativo entre dos o más computadoras conectadas a una red. [36]

El término Cliente/Servidor es originalmente aplicado a la arquitectura de software que describe el procesamiento entre dos o más programas: una aplicación y un servicio de soporte.

IBM define que el modelo Cliente/Servidor "Es la tecnología que proporciona al usuario final el acceso transparente a las aplicaciones, datos, servicios de cómputo o cualquier otro recurso del grupo de trabajo y/o, a través de la organización, en múltiples plataformas. [37]

El modelo soporta un medio ambiente distribuido en el cual los requerimientos de servicio hechos por estaciones de trabajo inteligentes o "clientes", resultan en un trabajo realizado por otros computadores llamados "servidores".

"Es un 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". [38]

Capítulo 1: Fundamentación Teórica

Los elementos principales de la arquitectura Cliente/Servidor son justamente el elemento llamado cliente y el otro elemento llamado servidor. Este modelo es una relación entre procesos corriendo en máquinas separadas. El servidor es un proveedor de servicios y el cliente es un consumidor de estos. [39]

Modelo Cliente/Servidor de tres Capas (Three Tier)

La arquitectura tradicional Cliente/Servidor implica dos niveles o capas: una capa cliente y una servidor. En la arquitectura de tres capas el software de aplicación esta distribuido en tres tipos de máquinas: una maquina de usuario, un servidor de capa intermedia y un servidor final (Backend).

La máquina de usuario es la misma de cliente y las máquinas de capa intermedia son esencialmente pasarelas entre los clientes y una variedad de servidores finales de base de datos, pueden convertir protocolos y traducir un tipo de consulta de base de datos a otro. Además puede mezclar e integrar resultados de distintas fuentes de datos. Por último puede servir como conexión entre aplicaciones de computador de escritorio y antiguas aplicaciones finales actuando de mediadoras entre los dos mundos.

La interacción entre el servidor de capa intermedia y el servidor final también sigue el modelo Cliente/Servidor. De esta forma el sistema de capa intermedia actúa a la vez como cliente y como servidor.

Cada uno de los componentes de la aplicación en una arquitectura de tres capas se separa en una sola entidad. Esto permite implementar componentes de una manera más flexible, es decir, la aplicación tiene que estar preparada para los posibles cambios que el cliente pueda pedir sin tener que reescribir totalmente la aplicación. Este tipo de arquitectura es la más compleja. (Ver Anexo 6)

Servidor web Apache

Es un servidor HTTP de código abierto para plataformas Unix (BSD, GNU/Linux), Windows y otras. Cuando comenzó su desarrollo en 1995 se basó inicialmente en código del popular servidor web NCSA HTTPd 1.3, pero más tarde fue reescrito por completo. Es el servidor Web más utilizado en el mundo. Su costo gratuito, gran fiabilidad y extensibilidad le convierten en una herramienta potente y muy fiable. Es un software que está estructurado en módulos. La configuración de cada módulo se hace mediante la

Capítulo 1: Fundamentación Teórica

configuración de las directivas que están contenidas dentro del módulo. Los módulos del Apache se pueden clasificar en tres categorías: [40]

- **Módulos Base:** Módulo con las funciones básicas del Apache.
- **Módulos Multiproceso:** son los responsables de la unión con los puertos de la máquina, aceptando las peticiones y enviando a los hijos a atender a las peticiones.
- **Módulos Adicionales:** Cualquier otro módulo que le añada una funcionalidad al servidor.

Las funcionalidades más elementales se encuentran en el módulo base, siendo necesario un módulo multiproceso para manejar las peticiones. Se han diseñado varios módulos multiproceso para cada uno de los sistemas operativos sobre los que se ejecuta el Apache, optimizando el rendimiento y la rapidez del código. [41]

Arquitectura Basada en Componentes (CBA)

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

Lenguajes de Programación Web

Los Lenguajes de Programación orientados a la Web se clasifican en lenguajes del lado del cliente y lenguajes del lado del servidor.

Entre los lenguajes que trabajan del lado del servidor se pueden citar algunos como son PERL, ASP, PHP, Java, JSP, que son los más usados en la actualidad. Desarrollan la lógica de negocio, y además se encargan de los accesos a los distintos Sistemas de Gestión de Bases de Datos.

Capítulo 1: Fundamentación Teórica

Del lado del cliente se encuentran principalmente JavaScript (JScript), XSL y el Visual Basic Script (VBScript), este último generalmente usado a la hora de programar en ASP del lado del servidor. Estos lenguajes son los encargados de aportar dinamismo a la aplicación en los navegadores.

En esta investigación se profundizará en los lenguajes de interés para el cliente que estos son los de código abierto o software libre.

PHP

Es un lenguaje de secuencias de comandos diseñado específicamente para la Web, embebido en páginas HTML y ejecutado en el servidor. Las siglas PHP equivalían inicialmente a Personal Home Page (página de inicio personal), pero se modificaron de acuerdo con la convención de designación de GNU y ahora equivale a PHP Hipertext Preprocessor (Preprocesador de hipertexto PHP). [42]

PHP es un producto de código abierto, lo que quiere decir que se puede acceder a su código. Puede ser utilizado, modificado y redistribuido sin coste alguno. [43]

PHP, en el caso de estar montado sobre un servidor Linux o Unix, es más rápido que ASP, dado que se ejecuta en un único espacio de memoria y esto evita las comunicaciones entre componentes COM que se realizan entre todas las tecnologías implicadas en una página ASP.

Algunas de las más importantes capacidades de PHP son: compatibilidad con las bases de datos más comunes, como MySQL, MSSQL, mSQL, Oracle, Informix y ODBC. Incluye funciones para el envío de correo electrónico, upload de archivos, crear dinámicamente en el servidor imágenes en formato GIF, entre otras utilidades. PHP es la gran tendencia en el mundo de Internet.

Este lenguaje corre en 7 plataformas, funciona en 11 tipos de servidores, ofrece soporte para unos 20 Gestores de Bases de Datos y contiene unas 40 extensiones estables sin contar las que se están experimentando, además de que:

- Su sintaxis está inspirada en C, ligeramente modificada para adaptarla al entorno en el que trabaja, de modo que si se está familiarizado con esta sintaxis, le resultará muy fácil aprender PHP, de ahí su simplicidad.

Capítulo 1: Fundamentación Teórica

- Es multiplataforma, es decir, puede ser utilizado en cualquiera de los principales sistemas operativos del mercado actual y es soportado por la mayoría de los servidores Web.
- Es muy rápido. Su integración con la base de datos MySQL y el servidor Apache, le permite constituirse como una de las alternativas más atractivas del mercado.
- Su librería estándar es realmente amplia, lo que permite reducir los llamados "costos ocultos", uno de los principales defectos de ASP.

PHP tiene una de las comunidades más grandes en Internet, por lo que es fácil encontrar ayuda, documentación, artículos, noticias y demás recursos.

JavaScript

Se trata de un lenguaje de programación del lado del cliente, porque es el navegador el que soporta la carga de procesamiento. Gracias a su compatibilidad con la mayoría de los navegadores modernos, es el lenguaje de programación del lado del cliente más utilizado. Con JavaScript se puede crear efectos especiales en las páginas y definir interactividades con el usuario. El navegador del cliente es el encargado de interpretar las instrucciones JavaScript y ejecutarlas para realizar estos efectos e interactividades, de modo que el mayor recurso, y tal vez el único, con que cuenta este lenguaje es el propio navegador. Es un lenguaje de programación bastante sencillo y pensado para hacer las cosas con rapidez, a veces con ligereza. [44]

Entre las acciones típicas que se pueden realizar en JavaScript se tienen dos vertientes. Por un lado los efectos especiales sobre páginas web, para crear contenidos dinámicos y elementos de la página que tengan movimiento, cambio de color o cualquier otro dinamismo. Por el otro, este lenguaje permite ejecutar instrucciones como respuesta a las acciones del usuario, con lo que se puede crear páginas interactivas con programas como calculadoras, agendas o tablas de cálculo. [45]

Brinda la posibilidad de programar pequeños scripts, pero también programas más grandes, orientados a objetos, con funciones, estructuras de datos complejas, etc. Además, pone a disposición del programador todos los elementos que forman la página web, para que éste pueda acceder a ellos y modificarlos dinámicamente. [46]

Capítulo 1: Fundamentación Teórica

Con JavaScript el programador se convierte en el verdadero dueño y controlador de cada cosa que ocurre en la página cuando la está visualizando el cliente. [47]

XSL

XSL (siglas de Extensible Stylesheet Language, expresión inglesa traducible como "lenguaje extensible de hojas de estilo") es una familia de lenguajes basados en el estándar XML que permite describir cómo la información contenida en un documento XML cualquiera debe ser transformada o formateada para su presentación en un medio específico.

Esta familia está formada por tres lenguajes:

- XSLT (siglas de Extensible Stylesheet Language Transformations, lenguaje de hojas extensibles de transformación), que permite convertir documentos XML de una sintaxis a otra (por ejemplo, de un XML a otro o a un documento HTML).
- XSL-FO (lenguaje de hojas extensibles de formateo de objetos), que permite especificar el formato visual con el cual se quiere presentar un documento XML, es usado principalmente para generar documentos PDF.
- XPath, o XML Path Language, es una sintaxis (no basada en XML) para acceder o referirse a porciones de un documento XML. XSL es una familia de lenguajes basados en el estándar XML que permite describir cómo la información contenida en un documento XML cualquiera debe ser transformada o formateada para su presentación en un medio específico.

Sistemas de Gestión de Bases de Datos (SGBD)

Los Sistemas Gestores de Bases de Datos son un tipo de software muy específico, dedicado a servir de interfaz entre las bases de datos y las aplicaciones que la utilizan. El objetivo fundamental de un SGBD consiste en suministrar al usuario las herramientas que le permitan manipular, en términos abstractos, los datos, o sea, de forma que no le sea necesario conocer el modo de almacenamiento de los datos en la computadora, ni el método de acceso empleado.

Un SGBD tiene los siguientes objetivos específicos:

- Independencia de los datos y los programas de aplicación.

Capítulo 1: Fundamentación Teórica

- Disminución de la redundancia.
- Integración y sincronización de las bases de datos.
- Integridad de los datos.
- Seguridad y protección de los datos.
- Facilidad de manipulación de la información.

MySQL

Es un sistema para la administración de bases de datos relacionales (RDBMS) rápido y sólido. Las bases de datos permiten almacenar, buscar, ordenar y recuperar datos de forma eficiente. El servidor de MySQL controla el acceso a los datos para garantizar el uso simultáneo de varios usuarios, para proporcionar acceso a dichos datos y asegurarse que solo obtienen acceso a ellos los usuarios con autorización. Por lo tanto, MySQL es un servidor multiusuario y de subprocesamiento múltiple. Utiliza SQL (Structured Query Language, lenguaje de consulta estructurado), es el lenguaje estándar para la consulta de bases de datos utilizado en todo el mundo. [48]

Ventajas:

- Diseñado con el objetivo de aumentar la velocidad.
- Consume muy pocos recursos de CPU y memoria. Muy buen rendimiento.
- Tamaño del registro sin límite.
- Buena integración con PHP.
- Utilidades de administración (phpMyAdmin).
- Buen control de acceso usuarios-tablas-permisos.

Desarrollo basado en RUP

Cada día la producción de software busca adecuarse más a las necesidades del usuario, esto trae como consecuencia que aumente en tamaño y complejidad.

Para lograr la productividad del software se necesita un proceso que integre las múltiples facetas del desarrollo del mismo.

Capítulo 1: Fundamentación Teórica

El Proceso Unificado de Desarrollo de Software (RUP), es una metodología de desarrollo de software orientada a objetos que proporciona un método disciplinado para asignar las tareas y responsabilidades dentro del equipo de desarrollo. Su objetivo es asegurar la producción de software de alta calidad que resuelva las necesidades del usuario dentro de un cronograma predecible y al menor costo posible. [49]

Un proceso de desarrollo de software es un conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema software. RUP es más que un simple proceso; es un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas software, para diferentes áreas de aplicación, diferentes tipos de organización, diferentes niveles de aptitud y diferentes tamaños de proyecto. [50]

Se caracteriza por ser iterativo e incremental, centrado en la arquitectura y guiado por casos de uso. Incluye artefactos, que son los productos tangibles del proceso, como por ejemplo: el modelo de casos de uso y el código fuente y también incluye roles, que es el papel que desempeña una persona en un determinado momento, pudiendo desempeñar distintos roles a lo largo del proceso. [51]

RUP divide el proceso en ciclos de desarrollo, teniendo un producto al final de cada ciclo. Los ciclos se dividen en fases que finalizan con un hito donde se debe tomar una decisión importante.

Fases definidas por RUP: [52]

- **Inicio:** Se describe el negocio y se delimita el proyecto describiendo sus alcances con la identificación de los casos de uso del sistema.
- **Elaboración:** Se define la arquitectura del sistema y se obtiene una aplicación ejecutable que responde a los casos de uso que la comprometen. A pesar de que se desarrolla a profundidad una parte del sistema, las decisiones sobre la arquitectura se hacen sobre la base de la comprensión del sistema completo y los requerimientos (funcionales y no funcionales) identificados de acuerdo al alcance definido.
- **Construcción:** Se obtiene un producto listo para su utilización que está documentado y tiene un manual de usuario. Se obtiene 1 o varios release del producto que han pasado las pruebas. Se ponen estos release a consideración de un subconjunto de usuarios.

Capítulo 1: Fundamentación Teórica

- **Transición:** El release ya está listo para su instalación en las condiciones reales. Puede implicar reparación de errores. (Ver Anexo 7)

Esta metodología de desarrollo de software utiliza el Lenguaje Unificado de Modelado (UML) para preparar todos los esquemas de un sistema software. RUP y UML están estrechamente relacionados entre sí, pues mientras el primero establece las actividades y los criterios para conducir un sistema desde su máximo nivel de abstracción (la idea en la cabeza del cliente), hasta su nivel más concreto (un programa ejecutándose en las instalaciones del cliente), el segundo ofrece la notación gráfica necesaria para representar los sucesivos modelos que se obtienen en el proceso de refinamiento.

RUP unifica las disciplinas en lo que a desarrollo de software se refiere, incluyendo modelado de negocio, manejo de requerimientos, componentes de desarrollo, ingeniería de datos, manejo y configuración de cambios y pruebas, cubriendo todo el ciclo de vida de los proyectos basado en la construcción de componentes y maximizando el uso del UML.

UML

UML son las siglas de Unified Modeling Language (Lenguaje Unificado para la Construcción de Modelos) se define como un "lenguaje que permite especificar, visualizar y construir los artefactos de los sistemas de software. Es un sistema notacional (que, entre otras cosas, incluye el significado de sus notaciones) destinado a los sistemas de modelado que utilizan conceptos orientados a objeto. Las características más generales de UML son: tecnología de orientación a objetos, viabilidad en la corrección de errores, desarrollo incremental e iterativo y participación del cliente en todas las etapas del proyecto. [53]

Puede usarse para modelar distintos tipos de sistemas: sistemas de software, sistemas de hardware, y organizaciones del mundo real.

UML permite: [54]

- Especificar todas las decisiones de análisis, diseño e implementación, construyéndose modelos precisos y completos.
- Conectarse a lenguajes de programación.
- Ingeniería directa e inversa.

Capítulo 1: Fundamentación Teórica

- Documentar todos los artefactos de un proceso de desarrollo (requerimientos, arquitectura, pruebas, versiones).

UML es un lenguaje expresivo, claro y uniforme, 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 las principales características son:

- Lenguaje unificado para la modelación de sistemas.
- Tecnología orientada a objetos.
- El cliente participa en todas las etapas del proyecto.
- Corrección de errores viables en todas las etapas.
- Aplicable para tratar asuntos de escala inherentes a sistemas complejos de misión crítica, tiempo real y Cliente/Servidor.

Plataforma de Servicio (PLASER)

La Plataforma de Servicios (PLASER) está conformada fundamentalmente por varias clases en PHP. Es una librería que puede ser usada opcionalmente integrar componentes al Registro Informatizado de la Salud (RIS), de no ser usada, la seguridad del sistema corre a cuenta del programador. En esta versión PLASER solo soporta como llamada RCP el protocolo SOAP, pero en futuras versiones se incorporarán otros protocolos de transporte o incluso el acceso local a código a nivel de File System, de forma tal que para el programador sea totalmente transparente si la invocación del proceso es remoto, local, por SOAP, directamente a código, etc.

Está concebido 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, con la ventaja de que el programador no tiene que preocuparse por implementar la seguridad del sistema, ya que esta es una de las tareas que asume PLASER, 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;

Capítulo 1: Fundamentación Teórica

pero desde el punto de vista de implementación solo trabaja con las bases de datos soportadas por el componente DBX, ya que PLASER encapsula a dicho componente y lo utiliza para el acceso a bases de datos.

Herramientas a utilizar

Se decidió utilizar la herramienta Case: Rational Rose Enterprise Edition 2003, para sustentar la documentación y como lenguaje de modelado a UML (Unified Modeling Language), para la confección de los diagramas que serán ilustrados en el documento.

Para el diseño de las páginas Web será utilizado en una primera etapa, Dreamweaver 8 de la familia Macromedia. Se utilizará además, el Stylus Studio 5.1, para que los desarrolladores creen ficheros XSL a través de escenarios.

Para la edición del código PHP se utilizará el NuSphere PHPEd 3.3, el cual es un editor para programadores con soporte para múltiples formatos. Esta herramienta facilita el trabajo de programación con numerosas características 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, incluye un cliente de FTP y un servidor Web integrado, totalmente configurables según las necesidades de trabajo.

Conclusiones

En este capítulo se expusieron los requisitos y premisas definidos para la informatización del sector de la salud en el país. Se profundizó en el conocimiento de algunos conceptos necesarios para la comprensión de la presente investigación, así como en los antecedentes históricos e importancia de la CIE y en los intentos de informatizarla, tanto en el ámbito nacional como internacional. Además, se realizó un análisis completo de las tecnologías que serán utilizadas a lo largo del desarrollo del sistema propuesto siguiendo las políticas trazadas por el MINSAP en el marco de la informatización del SNS; fundamentando así las elecciones de los lenguajes, el sistema gestor de bases de datos, la metodología a utilizar, entre otros.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

Introducción

En el presente capítulo se describen las características del sistema a construir. Debido a la poca definición de los procesos de negocio se hace necesaria la definición de conceptos y sus relaciones, agrupados en un Modelo de Dominio o Modelo Conceptual. Este sirve de apoyo para la especificación de las condiciones, capacidades y cualidades que el sistema debe tener; las que, se especifican en términos de Requerimientos y Casos de Uso, mostrándose estos últimos en un Diagrama de Casos de Uso en conjunto con los actores que interactúan con los mismos.

2.1 Modelo Dominio

El Modelo del Dominio o Modelo Conceptual, permite de manera visual mostrar al usuario los principales conceptos que se manejan en el dominio del sistema en desarrollo. Un modelo del dominio es una representación de las clases conceptuales del mundo real, no de componentes software. No se trata de un conjunto de diagramas que describen clases software, u objetos software con responsabilidades, el modelo del dominio podría considerarse como un diccionario visual de las abstracciones relevantes, vocabulario e información del dominio. Aprovechando las bondades de los diagramas UML para representar cosas y conceptos, el diagrama del modelo del dominio se presenta en forma de diagrama de clases donde figuran los principales conceptos y roles del sistema analizado. [55]

En el caso del Registro de la Clasificación Internacional de Enfermedades y Problemas Relacionados con la Salud (RCIE) se define un Modelo de Dominio, debido a la poca estructuración de los procesos de negocio en el flujo actual de los eventos, el cual permitirá a los usuarios, clientes, desarrolladores y otros interesados, utilizar un vocabulario común enmarcado en el contexto del sistema y contribuirá a una correcta captura de los requerimientos, que apoyarán la solución del problema.

Capítulo 2: Características del Sistema

2.1.1 Conceptos Fundamentales

Capítulo: Concepto definido en la CIE-10. Corresponde al primer nivel de la estructura jerárquica del codificador. Es un conjunto mayor de categorías ordenadas en grupos. El primer carácter del código se asocia con una letra y cada letra se asocia con un capítulo en particular, con sus excepciones.

Categoría: Concepto definido en la CIE-10. Corresponde al tercer nivel de la estructura jerárquica del codificador. Clave o código de tres caracteres que corresponde a una enfermedad o término diagnóstico. Dentro de cada grupo, algunas de las categorías de tres caracteres corresponden a afecciones únicas, seleccionadas según su frecuencia, gravedad o vulnerabilidad a las acciones de salud pública.

Grupo: Concepto definido en la CIE-10. Corresponde al segundo nivel de la estructura jerárquica del codificador. Es la división en bloques o grupos homogéneos de categorías de tres caracteres, afines o relacionadas con algún tipo de enfermedad u órgano del cuerpo dentro de un capítulo.

Subcategoría: Concepto definido en la CIE-10. Corresponde al cuarto nivel de la estructura jerárquica del codificador. La mayoría de las categorías de tres caracteres están subdivididas por medio de un carácter numérico, permitiendo hasta 10 subcategorías a las cuales les corresponde también un término diagnóstico.

Enfermedad o Problema de Salud: Estamos en presencia de un problema de salud o enfermedad cuando un paciente acude en busca de atención médica en cualquier nivel de atención.

Especialista de la Dirección Nacional de Estadística y Registros Médicos: Es la persona que recibe, realiza las actualizaciones de la CIE por los documentos que envía el Grupo Internacional de Expertos de la CIE de la OMS (de forma manual sobre el libro) mientras esté vigente la revisión e informa a las Direcciones Provinciales de Estadísticas y Registros Médicos. Además utilizan los códigos del libro de la CIE para codificar las causas de muerte de los certificados de defunción y para la morbilidad. Emite o informa al Grupo de Expertos de la CIE Regional o Mundial sobre posibles errores existentes en la versión o la inclusión/exclusión de códigos, enfermedades o problemas de salud.

Especialista de la Dirección Provincial de Estadística y Registros Médicos: Es la persona que recibe las orientaciones de las actualizaciones realizadas a la CIE, utilizan los códigos del libro de la CIE para

Capítulo 2: Características del Sistema

codificar las causas de muerte de los certificados de defunción y para la morbilidad e informa sobre posibles errores al Especialista principal de la Dirección Nacional de Estadística del MINSAP

Especialista de la Dirección de Estadística y Registros Médicos: Usuario que recibe las actualizaciones emitidas por la OMS de la CIE-10, consulta y actualiza de forma manual los libros de la CIE-10 e informa sobre errores o cambios a instancias internacionales.

2.1.2 Diagrama del Modelo de Dominio

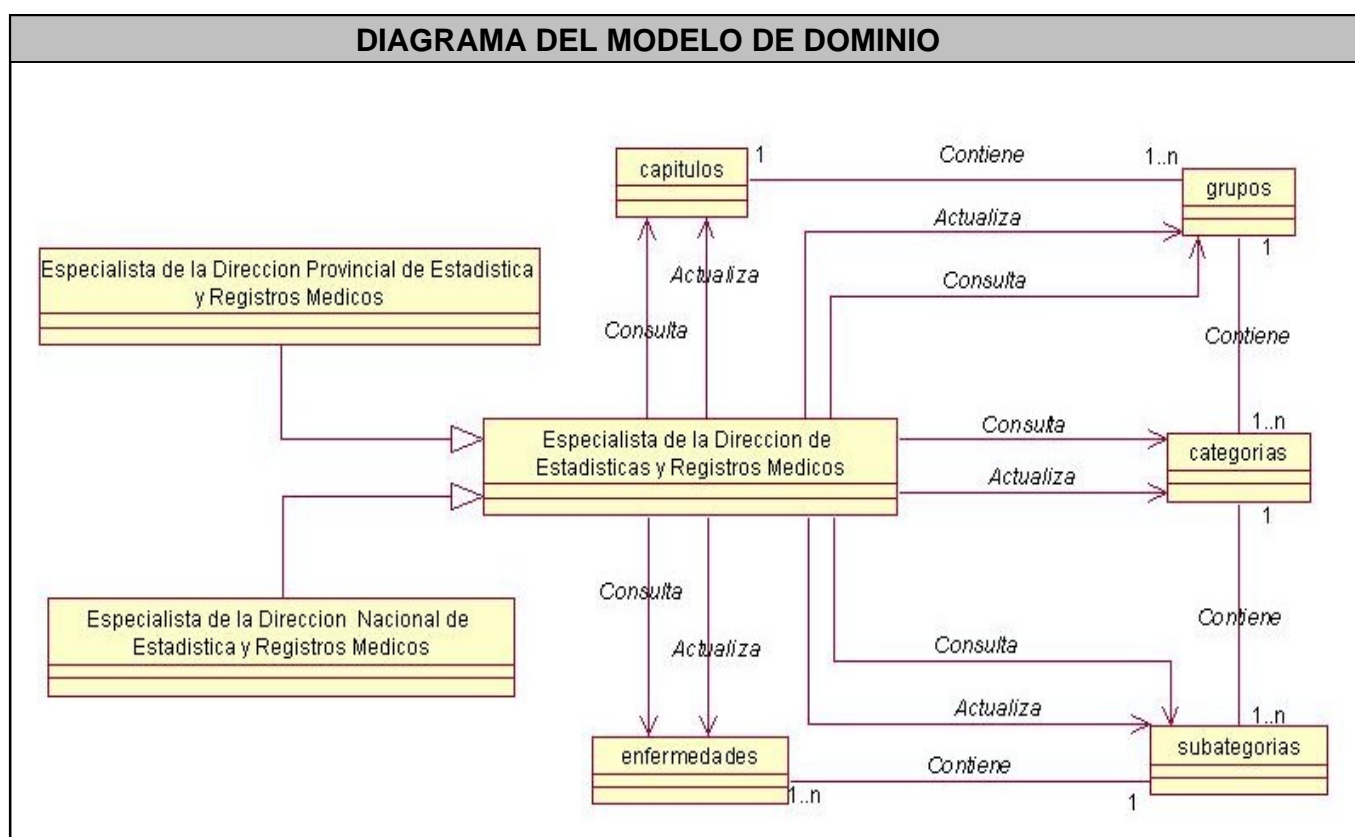


Figura 2.1 Diagrama del Modelo de Dominio

2.2 Propuesta del Sistema

Para la solución del problema planteado se debe partir de la definición de la Clasificación Internacional de Enfermedades o Problemas de Salud, para poder entender de una manera más amena el funcionamiento

Capítulo 2: Características del Sistema

del sistema. Una clasificación de enfermedades puede definirse como un sistema de categorías, a las que se asignan entidades morbosas de acuerdo con criterios establecidos y facilitan la recopilación de estadísticas. Una clasificación estadística de enfermedades debe abarcar toda la gama de estados morbosos dentro de un número manuable de categorías.

Por lo antes expuesto, se decidió que el RCIE mantenga la estructura lo más semejante al formato del libro, para permitir familiaridad a los distintos clientes que harán uso del mismo. La CIE-10 está estructurada de la siguiente manera: un capítulo está formado por grupos de categorías, las categorías están divididas en subcategorías y están agrupadas en conjuntos de problemas de salud según su codificación.

El sistema contará con un diseño de interfaces agradables y de fácil navegación, con ayuda y manual de usuario online. Permitirá la gestión y la realización de búsquedas eficientes de información, así como la impresión de la misma. Brindará servicios a las demás aplicaciones del Sistema de Información para la Salud, lo que apoyará a la homogeneidad de los diagnósticos en la salud.

2.2.1 Especificación de Requerimientos de Software

El propósito fundamental del flujo de trabajo de requerimientos es guiar el desarrollo hacia el sistema correcto. Esto se consigue mediante una descripción de los requerimientos del sistema (es decir, las condiciones o capacidades que el sistema debe cumplir) suficientemente buena como para que pueda llegarse a un acuerdo entre el cliente (incluyendo a los usuarios) y los desarrolladores sobre qué debe y qué no debe hacer el sistema. Los requerimientos deben ser especificados por escrito, descritos como una característica del sistema a construir y posibles de probar o verificar. [56]

2.2.1.1 Requerimientos Funcionales

Los requerimientos funcionales son capacidades o condiciones que el sistema debe cumplir sin alterar la funcionalidad del producto, esto quiere decir que se mantienen invariables sin importar con qué propiedades o cualidades se relacionan. Su principal tarea consiste en la generación de especificaciones correctas que describan con claridad, sin ambigüedades, en forma consistente y compacta, el comportamiento del sistema. [57]

Capítulo 2: Características del Sistema

A continuación se describen los requerimientos funcionales del sistema propuesto:

RF1 Gestionar la taxonomía de la Clasificación Internacional de Enfermedades y Problemas Relacionados con la Salud:

RF1.1 Gestionar (adicionar, modificar y eliminar) la información de los **Capítulos**:

- 1.1.1 Código
- 1.1.2 Descripción
- 1.1.3 Incluye
- 1.1.4 Excluye
- 1.4.5 Observaciones

RF1.2 Gestionar (adicionar, modificar y eliminar) la información de los **Grupos**:

- 1.2.1 Código
- 1.2.2 Descripción
- 1.2.3 Incluye
- 1.2.4 Excluye
- 1.2.5 Observaciones

RF1.3 Gestionar (adicionar, modificar y eliminar) la información de las **Categorías**:

- 1.3.1 Código
- 1.3.2 Descripción
- 1.3.3 Incluye
- 1.3.4 Excluye
- 1.3.5 Observaciones

RF1.4 Gestionar (adicionar, modificar y eliminar) la información de las **Subcategorías**:

- 1.4.1 Código
- 1.4.2 Descripción
- 1.4.3 Incluye
- 1.4.4 Excluye
- 1.4.5 Observaciones

RF1.5 Gestionar (adicionar, modificar y eliminar) la información de los **Problemas de Salud**:

- 1.5.1 Código

Capítulo 2: Características del Sistema

1.5.2 Descripción

RF2 Listar información de la taxonomía de la Clasificación Internacional de Enfermedades y Problemas Relacionados con la Salud:

RF2.1 Listar información de los **Capítulos**:

2.1.1 Código

2.1.2 Descripción

RF2.2 Listar información de los **Grupos**:

2.2.1 Código

2.2.2 Descripción

RF2.3 Listar información de las **Categorías**:

2.3.1 Código

2.3.2 Descripción

RF2.4 Listar información de las **Subcategorías**:

2.4.1 Código

2.4.2 Descripción

RF2.5 Listar información de los **Problemas de Salud**:

2.5.1 Código

2.5.2 Descripción

RF3 Imprimir información de la taxonomía de la Clasificación Internacional de Enfermedades y Problemas Relacionados con la Salud (Capítulos, Grupos, Categorías, Subcategorías y Problemas de Salud).

RF4 Permitir búsquedas dinámicas de acuerdo a criterios seleccionados por el usuario (Código, Descripción o ambos).

RF5 Mostrar información obtenida de las búsquedas.

RF6 Mostrar datos siguiendo la taxonomía de la Clasificación Internacional de Enfermedades y Problemas Relacionados con la Salud (Capítulos contiene a Grupos; Grupos contiene a Categorías; Categorías contiene a Subcategorías; y Subcategorías contiene a Problemas de Salud).

Capítulo 2: Características del Sistema

2.2.1.2 Requerimientos no Funcionales

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

Estos requerimientos 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 pueden marcar la diferencia entre un producto bien aceptado y uno con poca aceptación.

2.2.1.2.1 Usabilidad

Estos requerimientos describen los niveles apropiados de usabilidad, dados los usuarios finales del producto, para ello deben revisarse las especificaciones de los perfiles de usuarios y definir las clasificaciones de sus niveles de experiencia.

RNF1 Cada usuario que se autentique solo tendrá acceso a la información que le corresponde, ya sea editor o visualizador a nivel nacional, provincial, municipal o de unidad de salud.

2.2.1.2.2 Soporte

Abarcan todas las acciones a tomar una vez que se ha terminado el desarrollo del software con motivos de asistir a los clientes de éste, así como lograr su mejoramiento progresivo y evolución en el tiempo. Pueden incluir: Pruebas, Extensibilidad, Adaptabilidad, Mantenimiento, Compatibilidad, Configuración, Servicios, Instalación e Internacionalización.

RNF2 El personal que trabaja con el módulo debe contar con el nivel técnico requerido mediante adiestramiento de servicio.

Capítulo 2: Características del Sistema

2.2.1.2.3 Portabilidad

Estos requerimientos especifican los atributos que debe presentar el software para facilitar su traslado a otras plataformas u entornos.

RNF3 Permitir que el sistema se ejecute sobre el Sistema Operativo Windows 98 o superior o Linux.

2.2.1.2.4 Seguridad

Este es quizás el tipo de requerimiento más difícil, que provocará los mayores riesgos si no se maneja correctamente. La seguridad puede ser tratada en tres aspectos diferentes:

I. Confidencialidad: La información manejada por el sistema está protegida de acceso no autorizado y divulgación.

RNF4 Disponer de un mecanismo de seguridad basado en el modelo de Autenticación, Autorización y Auditoría (AAA).

RNF4.1 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 conocimiento exclusivo de la persona que se autentica.

RNF4.1.1 Si el usuario autenticado no se encuentra registrado se debe reportar un error de acceso.

RNF4.1.2 Si el usuario autenticado se encuentra registrado se autoriza su acceso y se crea un certificado digital.

RNF4.1.2.1 El certificado digital constará de:

- Un identificador único (token) de 32 caracteres generado de manera aleatoria en el momento de la autorización.
- El identificador del usuario.
- El nivel de acceso del usuario (Nacional, Provincial, Municipal o Unidad de Salud).
- El identificador del nivel de acceso.

Capítulo 2: Características del Sistema

- Un listado de los módulos que el usuario tiene acceso y el tipo de acceso en cada uno de ellos (Editor o Visualizador).

RNF4.1.2.2 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.

RNF4.2 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.

RNF4.2.1 Si el token se encuentra registrado y la petición se corresponde con los derechos del usuario se procederá a ejecutar la petición solicitada.

RNF4.2.2 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.

RNF4.3 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.

II. Integridad: La información manejada por el sistema será objeto de cuidadosa protección contra la corrupción y estados inconsistentes, de la misma forma será considerada igual a la fuente o autoridad de los datos. Pueden incluir también mecanismos de chequeo de integridad y realización de auditorías.

RNF5 Prevenir posibles fallos y recuperarse ante ellos.

RNF5.1 La información manejada por el sistema será objeto de cuidadosa protección contra la corrupción y estados inconsistentes. Deberán existir mecanismos de chequeo de integridad.

RNF5.2 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.

RNF5.3 Se permitirá la creación de copias de respaldo que puedan restaurar el sistema en caso de fallo crítico o pérdida total de la información.

Capítulo 2: Características del Sistema

III. Disponibilidad: Significa que a los usuarios autorizados se les garantizará el acceso a la información y que los dispositivos o mecanismos utilizados para lograr la seguridad no ocultarán o retrasarán a los usuarios para obtener los datos deseados en un momento dado.

RNF6 La seguridad no implicará lentitud o retaso en la respuesta dada por el sistema, por lo que se debe minimizar y reducir el tiempo de respuesta, así como optimizar el código.

2.2.1.2.5 Apariencia o Interfaz Externa

Este tipo de requerimiento describe la apariencia del producto. Es importante destacar que no se trata del diseño de la interfaz en detalle, sino que especifican cómo se pretende que sea la interfaz externa del producto.

RNF7 Todas las funciones se realizarán desde el portal principal y una vez realizada la opción, se vuelve a él.

RNF8 Diseñar una interfaz donde el personal que trabaja con la aplicación, seleccione y organice la información que desee mostrar con sólo hacer clic.

2.2.1.2.6 Apariencia o Interfaz Interna

Describen la apariencia del producto internamente, como deben ser sus componentes, su relación entre ellos, así como el funcionamiento interno del software.

RNF9 Todos los componentes del sistema deben desarrollarse siguiendo el principio de máxima cohesión y mínimo acoplamiento.

RNF10 Los componentes reutilizables entre los módulos que componen el Sistema de Información para la Salud deberán ser desarrollados como servicios web XML que interactúan a través de SOAP con otros componentes.

Capítulo 2: Características del Sistema

2.2.1.2.7 Ayuda y Documentación en Línea

Describen los requisitos que provee a los usuarios y clientes documentación para la utilización y manipulación del sistema, así como para entender el mismo.

RNF11 Disponer de instrucciones en una opción de ayuda.

RNF12 Disponer de una documentación del sistema, documentada con alguna metodología de desarrollo.

2.2.1.2.8 Software

Describen los programas que se deben disponer, para que el sistema funcione correctamente. Indican si hay que integrar el producto con otros productos de software o si para su correcta utilización hay que asociar otros programas al mismo.

RNF13 Los clientes tendrán acceso al Sistema de Información para la Salud a través de cualquier navegador web, recomendado Mozilla 1.5, Internet Explorer 5.0 o superior.

RNF14 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.

2.2.1.2.9 Hardware

Estos requisitos especifican las características lógicas para cada interfaz entre el producto y los componentes de hardware del sistema. Se incluirán características de configuración.

RNF15 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.

RNF16 Impresora local o de red para imprimir los reportes solicitados.

Capítulo 2: Características del Sistema

2.2.1.2.10 Restricciones en el diseño y la implementación

Este tipo de requerimiento especifica o restringe la codificación o construcción de un sistema, son restricciones que han sido ordenadas y deben ser cumplidas estrictamente. Ejemplos de ellas son: estándares requeridos, lenguajes de programación a ser usados para la implementación, uso obligatorio de ciertas herramientas de desarrollo, restricciones en la arquitectura o el diseño, bibliotecas de clases, entre otras.

RNF17 La lógica de presentación constituirá una capa independiente de la lógica de negocio, centrando su función en la interfaz de usuario y validaciones simples de los datos de entrada.

RNF18 Validar el proceso de la captación de datos para evitar entradas inadecuadas.

RNF19 Se utilizarán herramientas de desarrollo que garanticen la calidad de todo el ciclo de desarrollo del producto.

2.2.1.2.11 Rendimiento

Especifican de los requisitos relacionados con la carga que se espera tenga que soportar el sistema. Por ejemplo, el número de terminales, el número esperado de usuarios simultáneamente conectados, número de transacciones por segundo que deberá soportar el sistema, etc. Todos estos requisitos deben ser medibles. Debe indicarse en porcentaje (%) relacionado con su tiempo.

RNF20 El sistema debe soportar una conexión simultánea de más de 60 000 usuarios.

RNF21 El sistema debe soportar un tiempo de respuesta menor o igual a 30 segundos.

RNF22 El sistema debe soportar respuestas SOAP que no deben exceder los 50 Kbyte en el tiempo de respuesta al usuario.

2.2.2 Modelo de Casos de Uso del Sistema

El modelo de casos de uso permite que los desarrolladores de software y los clientes lleguen a un acuerdo sobre los requerimientos, es decir, sobre las condiciones y posibilidades que debe cumplir el sistema. El

Capítulo 2: Características del Sistema

mismo contiene actores, casos de uso y sus relaciones y constituye una entrada para el análisis, el diseño y las pruebas. [59]

2.2.2.1 Definición de actores

El modelo de casos de uso describe lo que hace el sistema para cada tipo de usuario. Cada uno de estos se representa mediante uno o más actores (personas). Por lo tanto, los actores representan terceros fuera del sistema que colaboran con el mismo. A continuación se describen los actores del sistema propuesto:

ACTORES DEL SISTEMA	JUSTIFICACIÓN
Especialista de la Dirección de Estadística y Registros Médicos	Es el encargado de la gestión de la información de la CIE-10 (Insertar, modificar y eliminar), además de tener los permisos de un usuario normal.
Usuario.	Usuario (todos los que soliciten CIE desde cualquier lugar de la red de Infomed, no necesitan ser autenticados). Es el encargado de listar, hacer búsquedas y recibir información.

Tabla 2.1 Justificación de Actores del Sistema

Capítulo 2: Características del Sistema

2.2.2.2 Diagrama de Casos de Uso

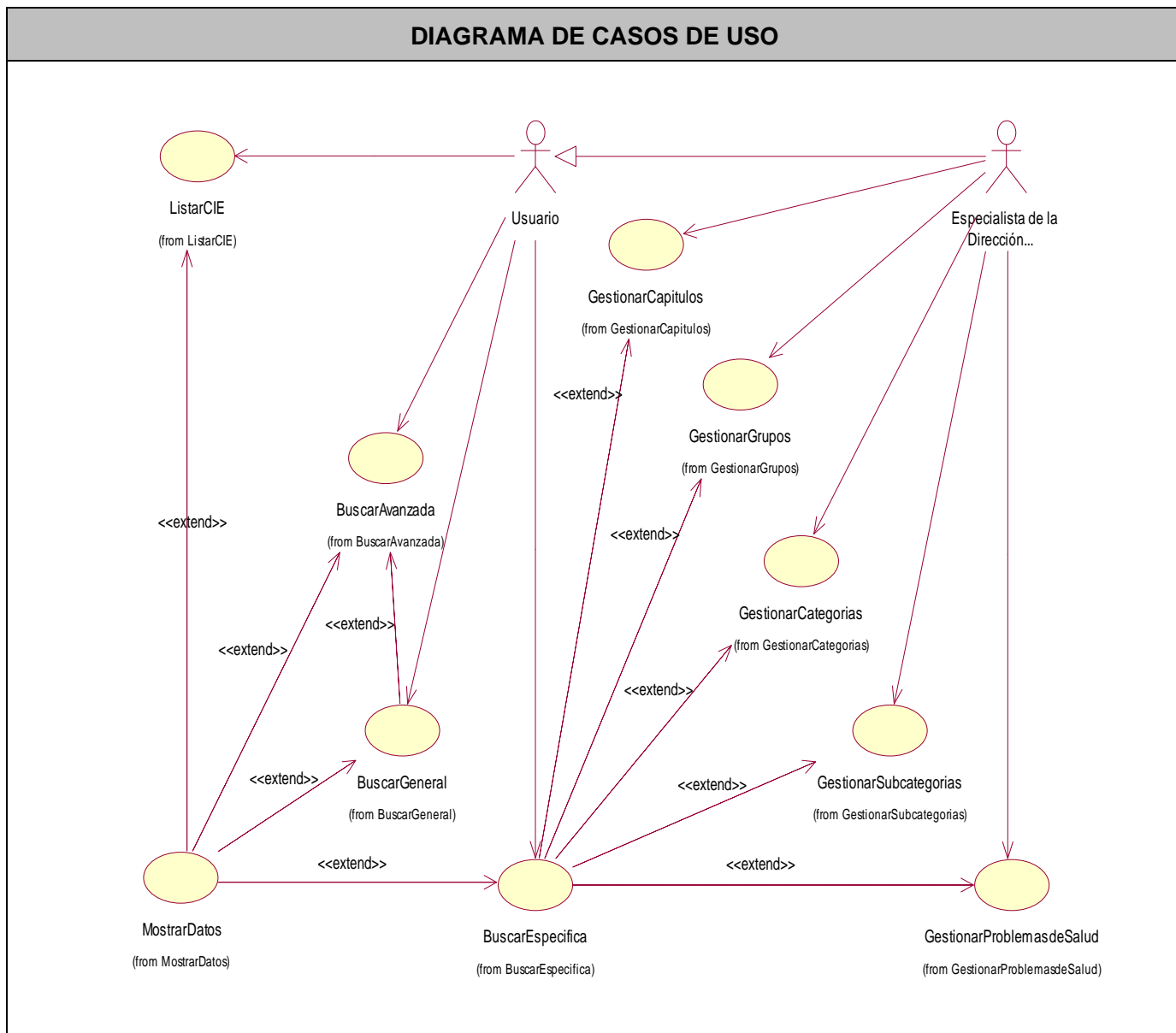


Figura 2.2 Diagrama de Casos de Uso del Sistema

2.2.2.3 Descripción textual de los Casos de Uso

CUS-1: ListarCIE	
Actores:	Usuario
Breve descripción:	El Caso de Uso comienza cuando el usuario desea obtener información asociada

Capítulo 2: Características del Sistema

<p>Precondiciones: Poscondiciones: Requisitos: Prioridad</p>	<p>a los Capítulos, Grupos, Categorías, Subcategorías o Problemas de Salud. El sistema hace una búsqueda y muestra la página con los códigos y descripciones de la opción seleccionada. El usuario tiene la posibilidad de imprimir la información mostrada, además de la opción de visualizar los datos de cada uno de los elementos, llamando al Caso de Uso extendido “MostrarDatos”, culminando así el Caso de Uso.</p> <p>RF2, RF3, CU MostrarDatos (extend) Crítico</p>
---	--

Tabla 2.2 Descripción textual CUS-1: ListarCIE

CUS-2: BuscarGeneral	
Actores:	Usuario
Breve descripción:	El Caso de Uso comienza cuando el usuario desea buscar información, introduce un código, una descripción o ambos y el sistema devuelve los resultados encontrados. El usuario tiene la posibilidad de imprimir dicha información, además de la opción de visualizar los datos de cada elemento encontrado, llamando al Caso de Uso extendido “MostrarDatos”, culminando así el Caso de Uso.
Precondiciones: Poscondiciones:	El sistema retorna los datos de Capítulos, Grupos, Categorías, Subcategorías y Enfermedades buscados al proceso que haya solicitado la búsqueda.
Requisitos: Prioridad	RF3,RF4,RF5, CU MostrarDatos (extend) Crítico

Tabla 2.3 Descripción textual CUS-2: BuscarGeneral

CUS-3: BuscarAvanzada	
Actores:	Usuario
Breve descripción:	El Caso de Uso comienza cuando el usuario desea hacer búsquedas restringidas especificando el Capítulo del cual desea obtener información, dentro del mismo restringe el Grupo, dentro de este la Categoría y luego la Subcategoría, el sistema devuelve la información encontrada. El usuario tiene la posibilidad de imprimir dicha información, además de la opción de visualizar los datos de cada elemento encontrado, llamando al Caso de Uso extendido “MostrarDatos”, culminando así el Caso de Uso.
Precondiciones: Poscondiciones:	El sistema retorna los datos de Capítulos, Grupos, Categorías, Subcategorías y Problemas de Salud buscados al proceso que haya solicitado la búsqueda.
Requisitos: Prioridad	RF3,RF4,RF5, CU MostrarDatos (extend) Crítico

Tabla 2.4 Descripción textual CUS-3: BuscarAvanzada

Capítulo 2: Características del Sistema

CUS-4: BuscarEspecifica	
Actores:	Usuario
Breve descripción:	El Caso de Uso comienza cuando el usuario desea hacer búsquedas específicas de Capítulos, Grupos, Categorías, Subcategorías o Problemas de Salud. El sistema devuelve la información encontrada. El usuario tiene la posibilidad de imprimir dicha información, además de la opción de visualizar los datos de cada elemento encontrado, llamando al Caso de Uso extendido “MostrarDatos”, culminando así el Caso de Uso.
Precondiciones:	
Poscondiciones:	El sistema retorna los datos de Capítulos, Grupos, Categorías, Subcategorías y Enfermedades buscados al proceso que haya solicitado la búsqueda.
Requisitos:	RF3,RF4,RF5, CU MostrarDatos (extend)
Prioridad	Crítico

Tabla 2.5 Descripción textual CUS-4: BuscarEspecifica

CUS-5: GestionarCapitulos	
Actores:	Codificador de la Dirección de Estadística del MINSAP.
Propósito:	Mantener actualizada la información referente a los Capítulos dentro del RCIE.
Breve descripción:	El Caso de Uso comienza cuando se desea eliminar, insertar o modificar la información referente a un/varios Capítulo(s), el actor inserta los datos del nuevo Capítulo o modifica los ya existentes en caso de ser necesario. Si se desea eliminar un/varios Capítulo(s), se verifica si tiene información asociada, en este caso no se realiza la eliminación; en caso contrario se procede a la eliminación física. El caso de uso finaliza con la actualización de la información referente al/los Capítulo(s).
Precondiciones:	El usuario debe haber sido autenticado en el Sistema.
Poscondiciones:	La Base de Datos queda actualizada.
Requisitos:	RF1.1, RF3, RF4, RF5
Prioridad	Crítico

Tabla 2.6 Descripción textual CUS-5: GestionarCapitulos

CUS-6: GestionarGrupos	
Actores:	Codificador de la Dirección de Estadística del MINSAP.
Propósito:	Mantener actualizada la información referente a los Grupos dentro del RCIE.
Breve descripción:	El Caso de Uso comienza cuando se desea eliminar, insertar o modificar la información referente a un/varios Grupo(s), el actor inserta los datos del nuevo Grupo o modifica los ya existentes en caso de ser necesario. Si se desea eliminar un/varios Grupo(s), se verifica si tiene información asociada, en este caso no se realiza la eliminación; en caso contrario se procede a la eliminación física. El caso de uso finaliza con la actualización de la información referente al/los Grupo(s).
Precondiciones:	El usuario debe haber sido autenticado en el Sistema.
Poscondiciones:	El sistema deja actualizada la información de Grupos del CIE.
Requisitos:	RF1.2, RF3, RF4, RF5
Prioridad	Crítico

Tabla 2.7 Descripción textual CUS-6: GestionarGrupos

Capítulo 2: Características del Sistema

CUS-7: Gestionar Categorías	
Actores:	Codificador de la Dirección de Estadística del MINSAP.
Propósito:	Mantener actualizada la información referente a las Categorías dentro del RCIE.
Breve descripción:	El Caso de Uso comienza cuando se desea eliminar, insertar o modificar la información referente a una/varias Categoría(s), el actor inserta los datos de la nueva Categoría o modifica los ya existentes en caso de ser necesario. Si se desea eliminar una/varias Categoría(s), se verifica si tiene información asociada, en este caso no se realiza la eliminación; en caso contrario se procede a la eliminación física. El caso de uso finaliza con la actualización de la información referente a la/las Categoría(s).
Precondiciones:	El usuario debe haber sido autenticado en el Sistema.
Poscondiciones:	El sistema deja actualizada la información de Categorías del CIE.
Requisitos:	RF1.3, RF3, RF4, RF5
Prioridad	Crítico

Tabla 2.8 Descripción textual CUS-7: GestionarCategorías

CUS-8: Gestionar Subcategorías	
Actores:	Codificador de la Dirección de Estadística del MINSAP.
Propósito:	Mantener actualizada la información referente a las Subcategorías dentro del RCIE.
Breve descripción:	El Caso de Uso comienza cuando se desea eliminar, insertar o modificar la información referente a una/varias Subcategorías(s), el actor inserta los datos de la nueva Subcategoría o modifica los ya existentes en caso de ser necesario. Si se desea eliminar una/varias Subcategoría(s), se verifica si tiene información asociada, en este caso no se realiza la eliminación; en caso contrario se procede a la eliminación física. El caso de uso finaliza con la actualización de la información referente a la/las Subcategoría(s).
Precondiciones:	El usuario debe haber sido autenticado en el Sistema.
Poscondiciones:	El sistema deja actualizada la información de Subcategorías del CIE.
Requisitos:	RF1.4, RF3, RF4, RF5
Prioridad	Crítico

Tabla 2.9 Descripción textual CUS-8: GestionarSubcategorías

CUS-9: Gestionar Problemas de Salud	
Actores:	Codificador de la Dirección de Estadística del MINSAP.
Propósito:	Mantener actualizada la información referente a los Problemas de Salud dentro del RCIE.
Breve descripción:	El Caso de Uso comienza cuando se desea eliminar, insertar o modificar la información referente a un/varios Problema(s) de Salud, el actor inserta los datos del nuevo Problema de Salud o modifica los ya existentes en caso de ser necesario. Si se desea eliminar un/varios Problema(s) de Salud, se verifica si tiene información asociada, en este caso no se realiza la eliminación; en caso contrario se procede a la eliminación física. El caso de uso finaliza con la actualización de la información referente al/los Problema(s) de Salud.
Precondiciones:	El usuario debe haber sido autenticado en el Sistema.

Capítulo 2: Características del Sistema

Poscondiciones:	El sistema deja actualizada la información de los Problemas de Salud del CIE.
Requisitos:	RF1.5, RF3, RF4, RF5
Prioridad	Crítico

Tabla 2.10 Descripción textual CUS-9: GestionarProblemasdeSalud

CUS-10: MostrarDatos (extend)	
Actores:	Usuario
Propósito:	Brindarle al usuario la posibilidad de ver todos los datos de algún Capítulo, Grupo, Categoría, Subcategoría o Problema de Salud específico, así como su taxonomía.
Breve descripción:	El Caso de Uso comienza después de que el actor escogió algún elemento resultante de la búsqueda, el sistema muestra los datos del elemento especificado y su taxonomía, culminando así el Caso de Uso.
Precondiciones:	El usuario debe haber dado clic en el Código o la Descripción de algún Capítulo, Grupo, Categoría, Subcategoría o Problema de Salud.
Poscondiciones:	
Requisitos:	RF6
Prioridad	Secundario

Tabla 2.11 Descripción textual CUS-10: MostrarDatos (extend)

Conclusiones

En este capítulo se definieron las características de la aplicación a construir, se especifican en términos de requerimientos funcionales y no funcionales, siendo modelados gráficamente mediante un Diagrama de Casos de Uso del Sistema. Además se identificaron y describieron los actores que interactuarán con las funcionalidades previstas para cada uno de estos casos de usos. Con el desarrollo de este capítulo quedaron sentadas las bases que darán paso al diseño y construcción del RCIE.

CAPÍTULO 3: DISEÑO DEL SISTEMA

Introducción

Con el modelado del diseño, se adquiere una mayor comprensión de los aspectos relacionados con los requerimientos funcionales, no funcionales y de las restricciones necesarias para el sistema. Este Flujo de Trabajo contribuye a la definición de una arquitectura estable y sólida, creando un plano del modelo de implementación. El propósito del presente capítulo es definir la estructura y elementos del diseño, describir los casos de uso en términos de clases y sus objetos, representándolos gráficamente en Diagramas de Clases, Así como fundamentar los patrones empleados y estructurar el Modelo de Datos.

3.1 Modelo de Diseño

Un Modelo de Diseño es una abstracción del Modelo de Implementación y su código fuente, el cual fundamentalmente se emplea para representar y documentar su diseño. Es usado como entrada esencial en las actividades relacionadas a implementación. Representa a los casos de uso en el dominio de la solución. Este Modelo puede contener: los diagramas, las clases, paquetes, subsistemas, relaciones, colaboraciones, atributos, las realizaciones de los casos de uso, entre otros que se puedan considerar para el sistema en desarrollo. [60]

3.1.1 Fundamentación del uso de Patrones

Un patrón es una solución recurrente para un problema en un contexto o la respuesta al problema dentro de un contexto que ayuda a resolver las dificultades. Los Patrones de Diseño son directrices y principios estructurados que describen un problema común y entregan una buena solución ya probada a la que se le da un nombre. La utilización de los mismos en la concepción de RCIE, ayudará a diseñar correctamente en menos tiempo, a construir clases reutilizables, facilitará la documentación y conducirá a la definición de una arquitectura pequeña, simple y comprensible.

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 se realiza un diseño que mezcle los componentes de interfaz y de negocio, entonces los cambios que se realicen en las interfaces producirán modificaciones en los componentes de negocio, lo que proporcionará mayor trabajo y más riesgo de errores.

En el diseño de RCIE se separa la vista del modelo, con la finalidad de mejorar la reusabilidad. De esta forma las modificaciones en las vistas/interfaces impactarán en menor medida en la lógica de negocio o de datos, reduciendo el impacto del cambio.

Elementos del patrón:

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

En la propuesta de solución (RCIE):

- El **modelo:** Formado por la base de datos de RCIE.
- El **controlador:** Métodos del negocio.
- Las **vistas:** Constituidas por los ficheros XSL y JavaScript, clientes PHP de la capa de presentación y FachadaCIE.

3.1.1.2 Alta Cohesión

La cohesión es la medida de la fuerza que une a las responsabilidades de una clase. Una clase con baja cohesión es aquella que hace muchas cosas no afines o muchas tareas, lo que trae como consecuencias dificultades para entender, reutilizar y conservarla. Son delicadas y las afectan constantemente los cambios. Una clase con alta cohesión mejora la claridad y la facilidad de su uso, su mantenimiento se simplifica y es fácil de reutilizar. A menudo se genera un bajo acoplamiento.

RCIE se diseñó asignando responsabilidades de modo que la cohesión sea alta, por lo que los componentes del mismo tienen el mínimo de tareas en el sistema, ejemplo de esto lo representan los métodos del negocio, ya que tienen bien delimitadas sus responsabilidades, no recargando en ningún caso sus funcionalidades, permitiendo mayor eficiencia y que el tiempo de respuesta y ejecución no exceda lo estimado, generando por ende bajo acoplamiento.

3.1.1.3 Bajo Acoplamiento

El acoplamiento es la medida de la fuerza con que una clase está conectada a otras clases, a las que conoce y recurre a ellas. Una clase con bajo (o débil) acoplamiento no depende de muchas otras. Una clase con alto (o fuerte) acoplamiento recurre a muchas otras, lo que trae consigo que los cambios en una clase ocasionen cambios en otras, sean más difíciles de entender y de reutilizar porque se requiere la presencia de otras clases de las que dependen.

El diseño del módulo RCIE está regido por la creación de dependencias escasas, un ejemplo de esto lo constituye la definición de métodos del negocio poco dependientes de otros, lo que reduce el impacto del cambio y favorece la reutilización.

3.1.1.4 Fachada

El patrón fachada trata de simplificar la interfaz entre dos sistemas o componentes de software ocultando un sistema complejo detrás de una clase que hace de pantalla o fachada. Denominado por algunos como, patrón “no converses con extraños”, en el diseño de RCIE se define su uso para abstraer a la capa de presentación de la interacción con las clases de la librería PLASER, para esto se definió la implementación de una clase Fachada en la capa de presentación la cual interactúa con una clase Fachada de PLASER, ocultando la complejidad de sus acciones internas. Este patrón brinda, además, escalabilidad y portabilidad al sistema, reduciendo el impacto del cambio.

3.1.2 Definición de la estructura del diseño.

Se ha definido que la aplicación posea una estructura centrada en la flexibilidad ante los cambios y la reutilización. Mediante la conservación de dicha estructura durante la implementación, se obtendrá un sistema mantenible como un todo: será flexible a los cambios en los requerimientos e incluirá elementos que podrán ser reutilizados cuando se construyan sistemas similares.

El siguiente diagrama muestra el diseño de la estructura de RCIE:

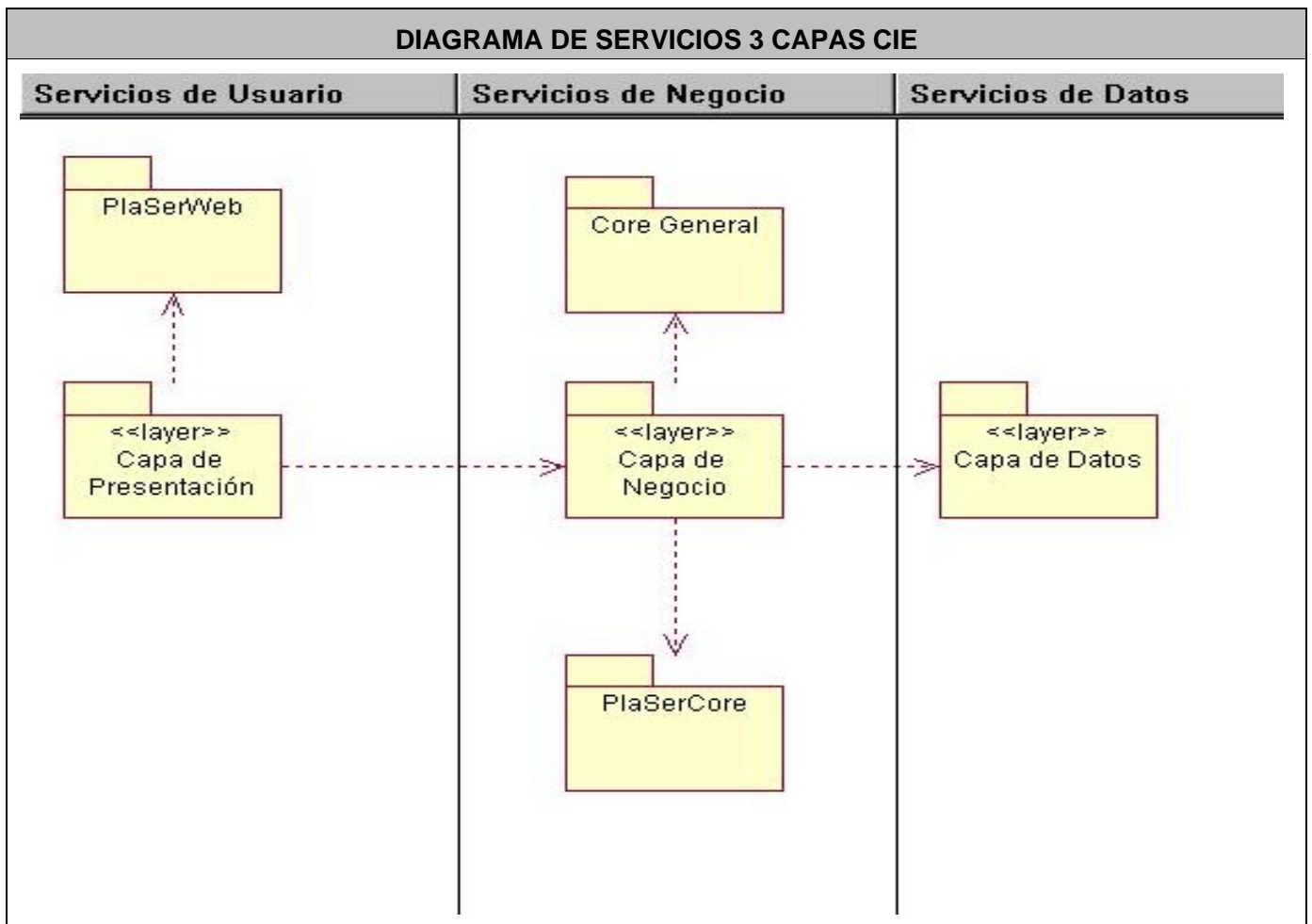


Figura 3.1 Diagrama de Servicios 3 Capas CIE

Capítulo 3: Diseño del Sistema

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

Capa de Presentación: Esta capa encapsula la lógica de navegación y controla los eventos de las interfaces. Es la encargada de interactuar con el usuario y hacer transparente la complejidad del sistema a los mismos, estando conformada por: DHTML, XML, PHP y XSL.

Capa de Negocio: Esta capa encapsula la lógica de negocio. Establece comunicación entre la capa de presentación y la de datos, recibiendo y respondiendo peticiones. Los métodos que la conforman reciben solicitudes de la capa superior, se comunican con los ficheros de acceso a datos y actualizan o retornan información.

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

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

PlaSerWeb: Este paquete contiene la clase CFachada de la cual hereda la clase FachadaCIE ubicada en la capa de presentación del módulo, la misma es la encargada de transformar las páginas xsl en html mediante la función transforma, para esto incluye las clases PIASER_xml, PLASER_xslt y de acceder al negocio incluyendo a PLASER_Client.

PlaSerCore: Este paquete contiene la clase de acceso a datos DBz y la clase encargada de brindar la dirección física de los componentes, PLASER_Server, la que incluye para su funcionamiento al fichero confplaser_server.

3.1.3 Diagramas de Clases del Diseño

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

Capítulo 3: Diseño del Sistema

Al ser RCIE una aplicación web, la misma se modelará con estereotipos definidos para este tipo de aplicaciones, como son los estereotipos web, lo que proporciona una mayor comprensión de las funcionalidades del sistema y el poder distinguir qué atributos, operaciones y relaciones están activas en el servidor y cuáles están activas cuando el usuario está interactuando con la página en el navegador cliente.

A continuación se brinda una explicación de cómo son usados estos estereotipos en el diseño de la propuesta del sistema y qué representa cada cual:



sp_<NombreClaseServidora>

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



cl_<NombreClaseCliente>

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



<nombre>

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

Capítulo 3: Diseño del Sistema

Relaciones utilizadas entre clases:

Hasta Desde	Client Page	Form	Server Page
Client Page		<<contine>>	
Form			<<submit>>
Server Page	<<build>>		<<redirect>> <<call>> <<redirect>> <<include>>

Tabla 3.1 Relaciones entre clases

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

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

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

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

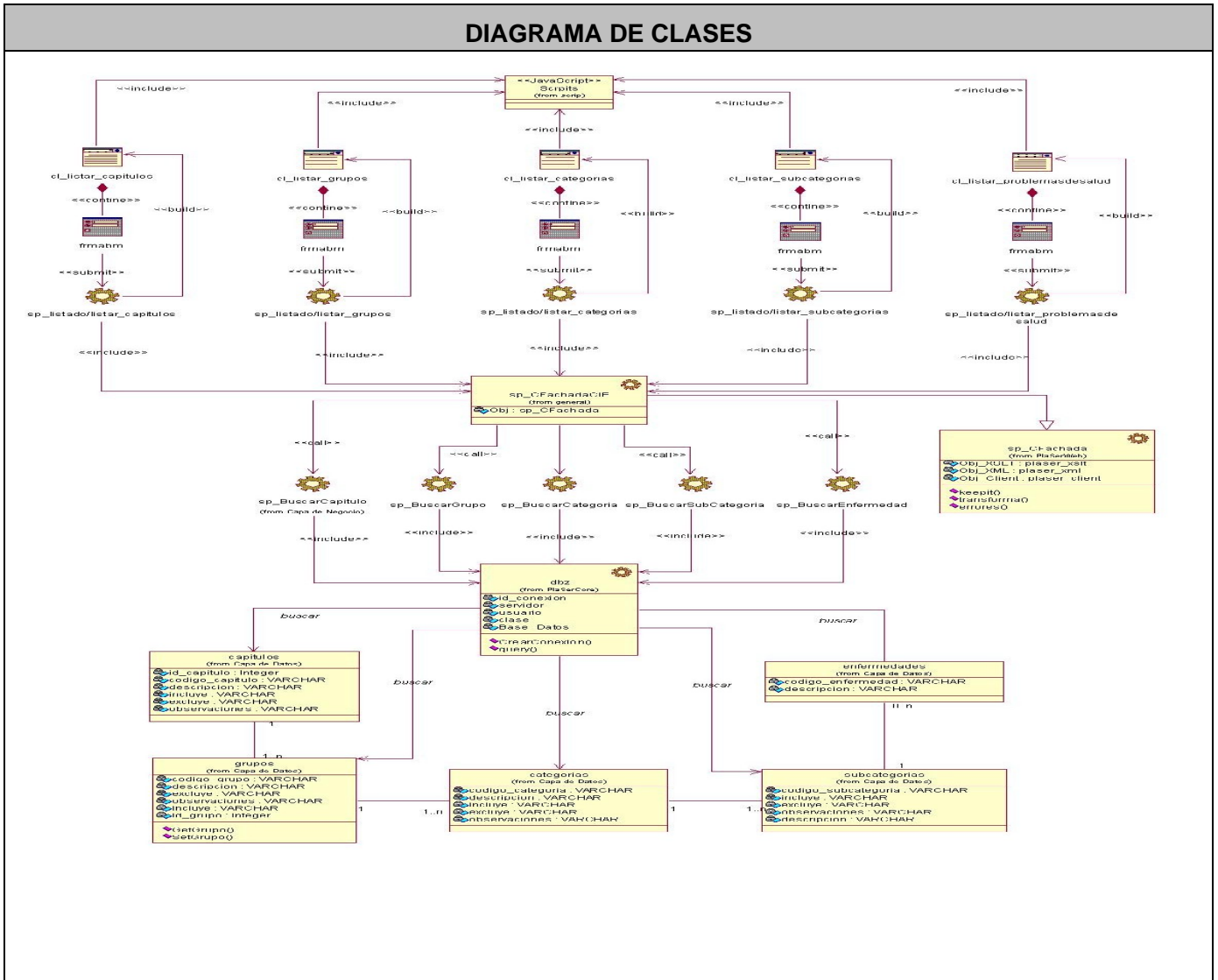


Figura 3.2 Diagrama de Clases del Diseño CUS-1: ListarCIE

Capítulo 3: Diseño del Sistema

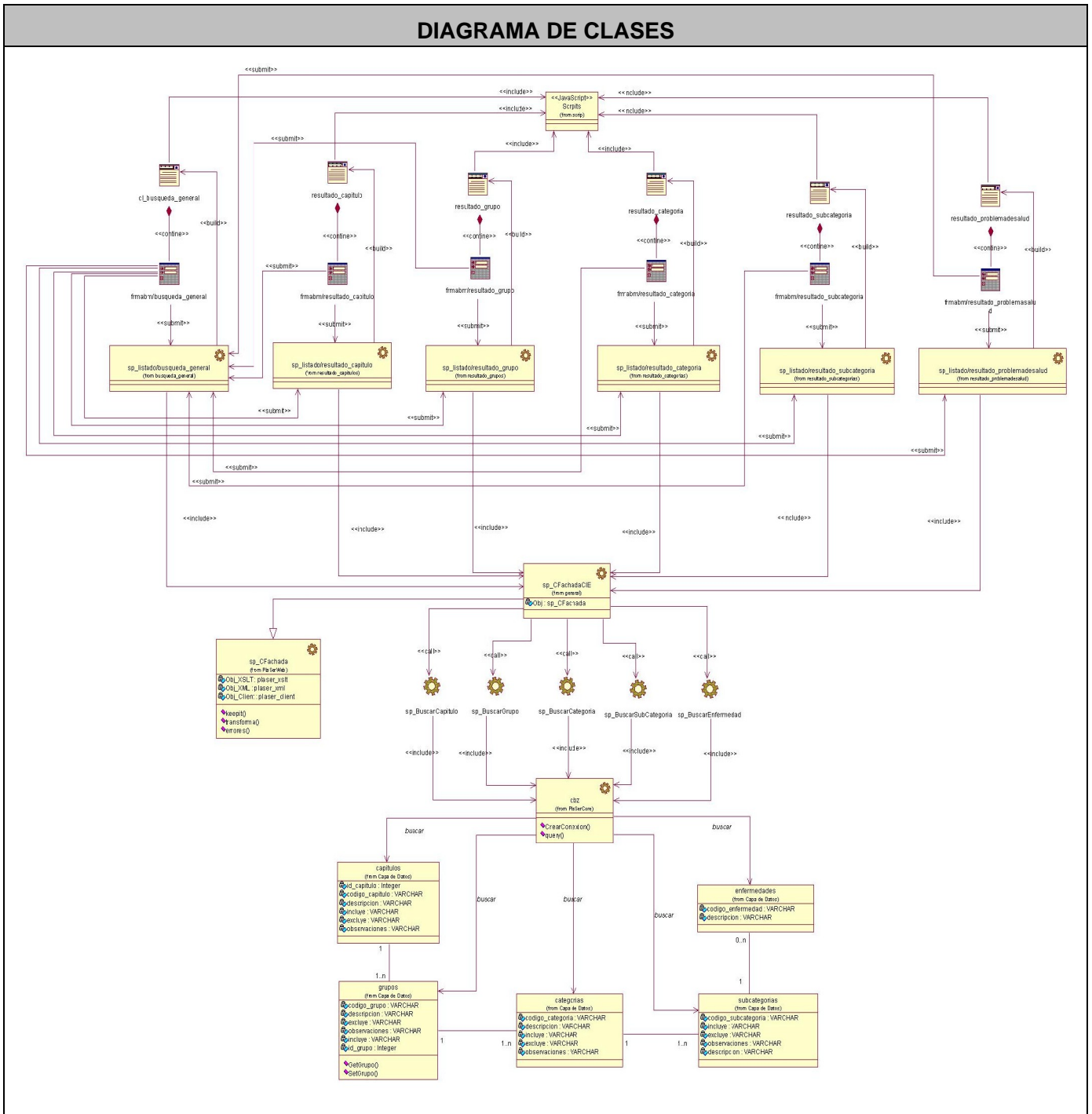


Figura 3.3 Diagrama de Clases del Diseño CUS-2: BuscarGeneral

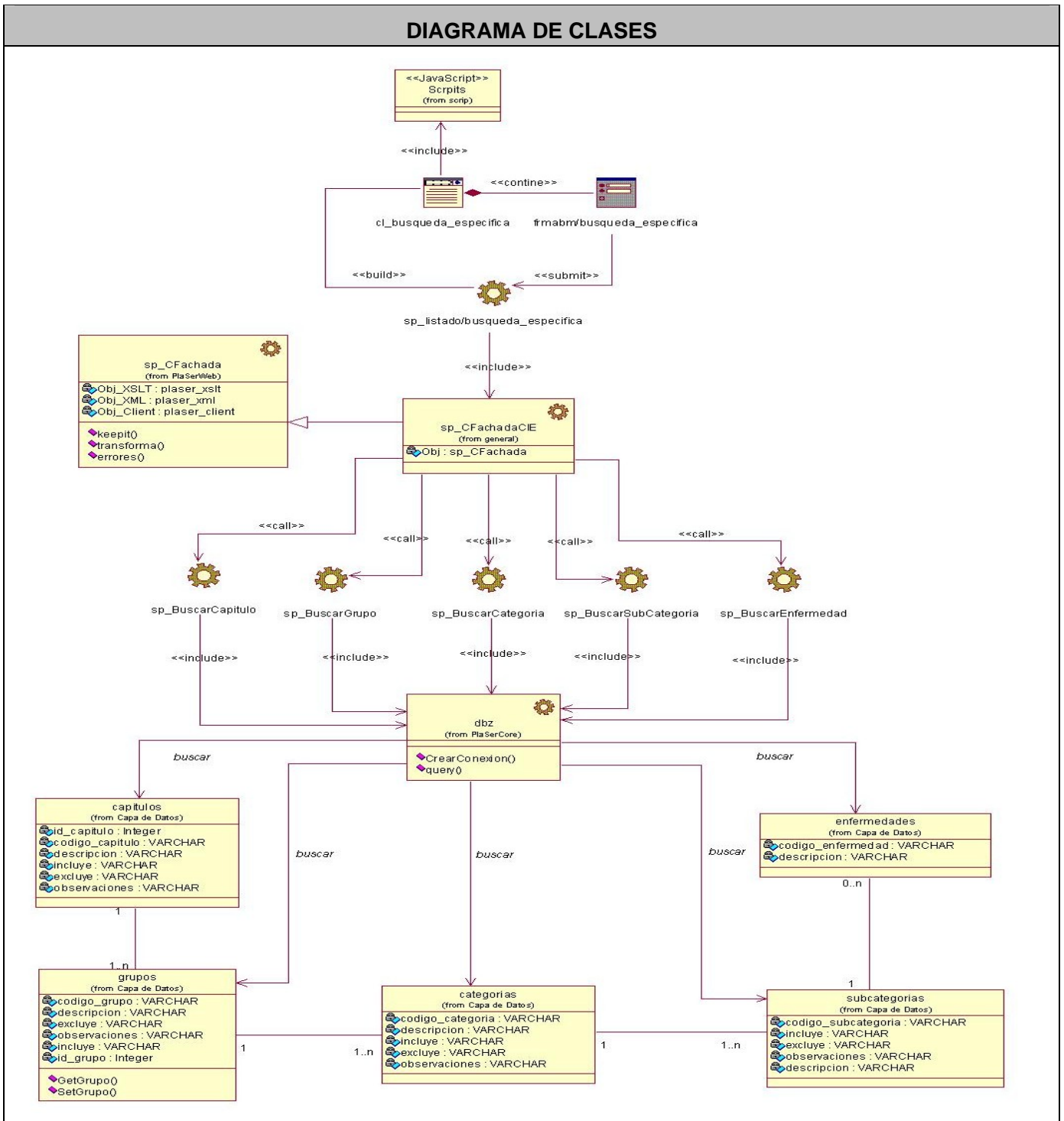


Figura 3.4 Diagrama de Clases del Diseño CUS-4: BuscarEspecifica

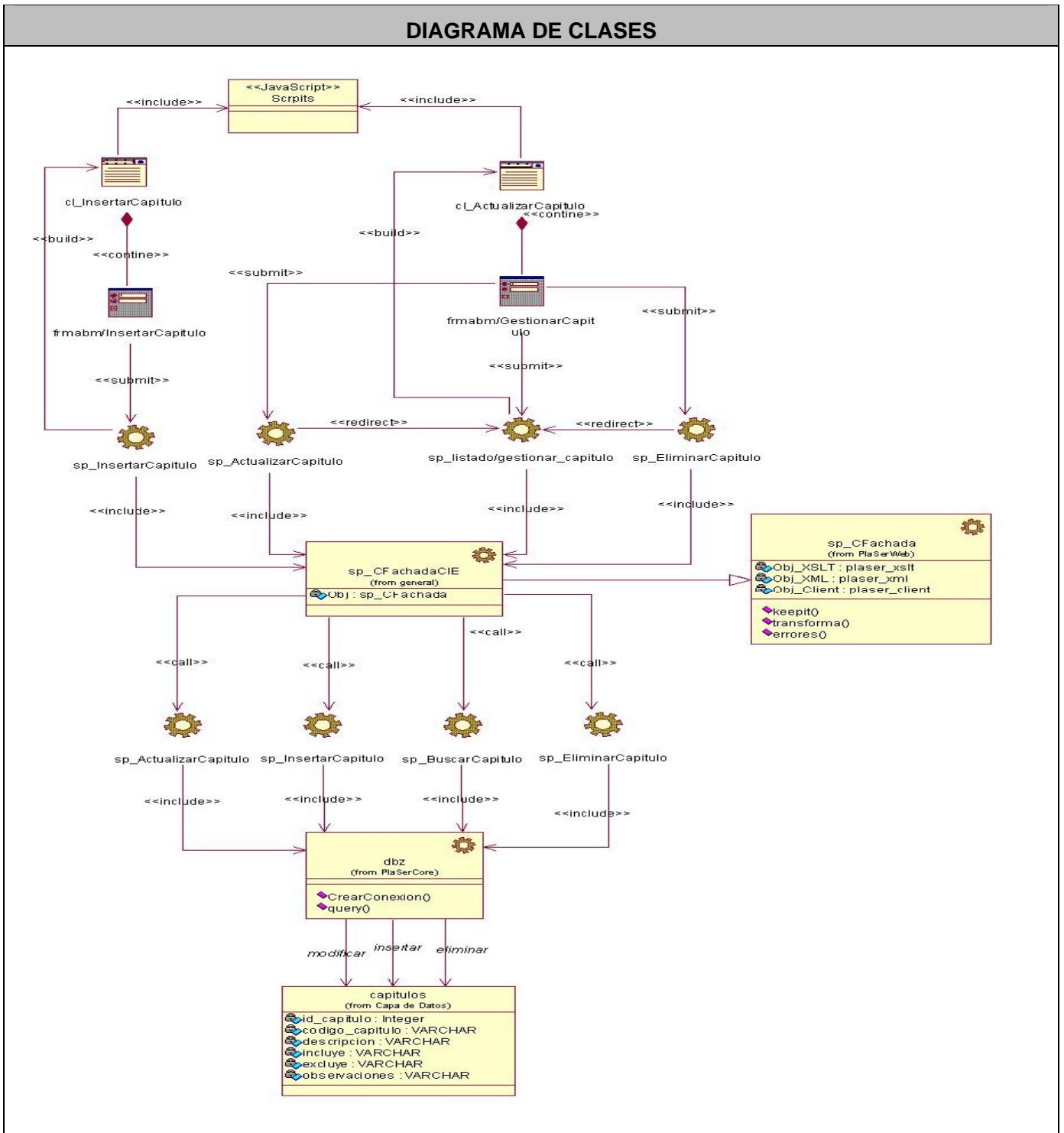


Figura 3.5 Diagrama de Clases del Diseño CUS-5: GestionarCapitulos

3.1.4 Descripción de las clases y atributos

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

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

3.1.4.1 Descripción de páginas clientes.

Nombre: CI_listado
Tipo de clase: Client Page
Descripción General: La clase <i>CI_listado</i> es una página Web que se ejecuta del lado del cliente sobre un navegador Web. Permite a los Visualizadores o Editores Provinciales, Municipales o de Unidad de Salud la realización de listado y búsquedas de información específica permitiendo generar documentos Portable Document Format (.pdf) y Microsoft Office Excel (.xls). A través de esta página se puede visualizar la información y el resultado de las búsquedas es paginada permitiendo la movilidad por tales resultados, accediendo directamente a la última. Es utilizada en los siguientes casos de uso: <ul style="list-style-type: none">✓ ListarCIE✓ BusquedaGeneral✓ BusquedaAvanzada✓ BusquedaEspecifica

Tabla 3.2 Descripción de la página cliente CI_listado

Nombre: CI_Actualizar
Tipo de clase: Client Page
Descripción General: La clase <i>CI_Actualizar</i> es una página Web que se ejecuta del lado del cliente sobre un navegador Web. Permite a los Editores Nacionales de Estadísticas la realización de listado y búsquedas de información específica permitiendo generar documento Portable Document Format (.pdf) y Microsoft Office Excel (.xls). A través de esta página se puede modificar o eliminar la información almacenada o simplemente visualizar información. El resultado de las búsquedas es paginada cada 10 elementos, permitiendo la movilidad por tales resultado, además de ir directamente a la última página. Es utilizada en los siguientes casos de uso:

Capítulo 3: Diseño del Sistema

- ✓ GestionarCapitulos
- ✓ GestionarGrupos
- ✓ GestionarCategorias
- ✓ GestionarSubcategorias
- ✓ GestionarProblemasdeSalud

Tabla 3.3 Descripción de la página cliente CI_Actualizar

Nombre: CI_Insertar
Tipo de clase: Client Page
Descripción General: La clase <i>CI_Insertar</i> 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 del Registro, posee un conjunto de validaciones en JavaScript que no permiten realizar peticiones innecesarias y por lo tanto se incrementa su usabilidad. Es utilizada en los siguientes casos de uso: <ul style="list-style-type: none">✓ GestionarCapitulos✓ GestionarGrupos✓ GestionarCategorias✓ GestionarSubcategorias✓ GestionarProblemasdeSalud

Tabla 3.4 Descripción de la página cliente CI_Insertar

3.1.4.2 Descripción de páginas servidoras

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

Tabla 3.5 Descripción de la página servidora SP_CFachada

Capítulo 3: Diseño del Sistema

Nombre: SP_CFachadaCIE
Tipo de clase: Server Page
Descripción General: La clase <i>SP_CFachadaCIE</i> es una clase que representa a la clase <i>SP_CFachada</i> dentro del módulo y hereda de ella. Cada uno de los módulos contiene la suya. Su función principal es modelar la comunicación con la capa de negocio.

Tabla 3.6 Descripción de la página servidora SP_CFachadaCIE

Nombre: SP_listado
Tipo de clase: Server Page
Descripción General: La clase <i>SP_listado</i> es una clase que se ejecuta del lado del servidor en la Capa de Presentación. Su actividad es construir una página cliente. Aplica un documento (XSL) a otro documento XML para transformarlo y mostrarlo al usuario en formato XHTML. Es utilizada en los siguientes casos de uso: <ul style="list-style-type: none">✓ ListarCIE✓ BusquedaGeneral✓ BusquedaAvanzada✓ BusquedaEspecifica✓ GestionarCapitulos✓ GestionarGrupos✓ GestionarCategorias✓ GestionarSubcategorias✓ GestionarProblemasdeSalud

Tabla 3.7 Descripción de la página servidora SP_listado

Capítulo 3: Diseño del Sistema

Nombre: SP_Insertar
Tipo de clase: Server Page
Descripción General: La clase <i>SP_Insertar</i> es una clase que se ejecuta del lado del servidor en la Capa de Presentación. Su actividad es construir la página cliente <i>CI_Insertar</i> . Aplica un documento (XSL) a otro documento XML para transformarlo y mostrarlo al usuario en formato XHTML para posteriormente recibir los valores introducidos por el mismo y enviarlos hacia la Capa de Negocio a través de la clase <i>SP_FachadaCIE</i> la cual invoca al método del negocio para la inserción de los datos involucrados y una vez concluida la ejecución de sus responsabilidades devuelve una respuesta a la clase <i>SP_Insertar</i> . Es utilizada en los siguientes casos de uso: <ul style="list-style-type: none">✓ GestionarCapitulos✓ GestionarGrupos✓ GestionarCategorias✓ GestionarSubcategorias✓ GestionarProblemasdeSalud

Tabla 3.8 Descripción de la página servidora SP_Insertar

Nombre: SP_Actualizar
Tipo de clase: Server Page
Descripción General: La clase <i>SP_Actualizar</i> es una clase que se ejecuta del lado del servidor en la Capa de Presentación. Su actividad es recibir los valores de la página cliente <i>CI_Actualizar</i> para posteriormente enviarlos hacia la Capa de Negocio a través de la clase <i>SP_FachadaCIE</i> la cual invoca al método del negocio para la modificación de los datos involucrados y una vez concluida la ejecución de sus responsabilidades devuelve una respuesta a la clase <i>SP_Actualizar</i> la cual redirecciona su ejecución a la <i>clase SP_Listado</i> . Es utilizada en los siguientes casos de uso: <ul style="list-style-type: none">✓ GestionarCapitulos✓ GestionarGrupos✓ GestionarCategorias✓ GestionarSubcategorias✓ GestionarProblemasdeSalud

Tabla 3.9 Descripción de la página servidora SP_Actualizar

Nombre: SP_Eliminar
Tipo de clase: Server Page
Descripción General: La clase <i>SP_Eliminar</i> es una clase que se ejecuta del lado del servidor en la Capa de Presentación. Su actividad es recibir los valores de la página cliente <i>CI_Actualizar</i> para posteriormente enviarlos hacia la Capa de Negocio a través de la clase <i>SP_FachadaCIE</i> la cual invoca al método del negocio para la eliminación de los datos involucrados y una vez concluida la ejecución de sus responsabilidades devuelve una respuesta a la clase <i>SP_Eliminar</i> la cual redirecciona su ejecución a la <i>clase SP_listado</i> . Es utilizada en los siguientes casos de uso: <ul style="list-style-type: none"> ✓ GestionarCapitulos ✓ GestionarGrupos ✓ GestionarCategorias ✓ GestionarSubcategorias ✓ GestionarProblemasdeSalud

Tabla 3.10 Descripción de la página servidora SP_Eliminar

3.1.4.3 Descripción de métodos del negocio

Nombre: SP_BuscarCapitulo	
Tipo de clase: Server Page	
Descripción General: La clase <i>SP_BuscarCapitulo</i> es una clase que se ejecuta del lado del servidor en la Capa de Negocio. Su actividad es devolverle al usuario un listado de los capítulos según los parámetros solicitados.	
Parámetros de Entrada	
Descripción	Tipo
idCapitulo	int
CodigoCapitulo	string
Descripcion	string
Incluye	string
Excluye	string
Observaciones	string
Offset	int

Capítulo 3: Diseño del Sistema

Cantidad	int
----------	-----

Tabla 3.11 Descripción de la página servidora SP_BuscarCapitulo

Nombre: SP_BuscarGrupo	
Tipo de clase: Server Page	
Descripción General: La clase <i>SP_BuscarGrupo</i> es una clase que se ejecuta del lado del servidor en la Capa de Negocio. Su actividad es devolverle al usuario un listado de los grupos según los parámetros solicitados.	
Parámetros de Entrada	
Descripción	Tipo
idGrupo	int
idCapitulo	int
CodigoGrupo	string
Descripcion	string
Incluye	string
Excluye	string
Observaciones	string
Offset	int
Cantidad	int

Tabla 3.12 Descripción de la página servidora SP_BuscarGrupo

Nombre: SP_BuscarCategoria	
Tipo de clase: Server Page	
Descripción General: La clase <i>SP_BuscarCategoria</i> es una clase que se ejecuta del lado del servidor en la Capa de Negocio. Su actividad es devolverle al usuario un listado de las categorías según los parámetros solicitados.	
Parámetros de Entrada	
Descripción	Tipo
idGrupo	int
CodigoCategoria	string
Descripcion	string
Incluye	string
Excluye	string
Observaciones	string
Offset	int

Capítulo 3: Diseño del Sistema

Cantidad	int
----------	-----

Tabla 3.13 Descripción de la página servidora SP_BuscarCategoria

Nombre: SP_BuscarSubCategoria	
Tipo de clase: Server Page	
Descripción General: La clase <i>SP_BuscarSubCategoria</i> es una clase que se ejecuta del lado del servidor en la Capa de Negocio. Su actividad es devolverle al usuario un listado de las subcategorias según los parámetros solicitados.	
Parámetros de Entrada	
Descripción	Tipo
CodigoCategoria	string
CodigoSubCategoria	string
Descripcion	string
Incluye	string
Excluye	string
Observaciones	string
Offset	int
Cantidad	int

Tabla 3.14 Descripción de la página servidora SP_BuscarSubCategoria

Nombre: SP_BuscarEnfermedad	
Tipo de clase: Server Page	
Descripción General: La clase <i>SP_BuscarEnfermedad</i> es una clase que se ejecuta del lado del servidor en la Capa de Negocio. Su actividad es devolverle al usuario un listado de las enfermedades según los parámetros solicitados.	
Parámetros de Entrada	
Descripción	Tipo
CodigoSubCategoria	string
CodigoEnfermedad	string
Descripcion	string
Offset	int

Capítulo 3: Diseño del Sistema

Cantidad	int
----------	-----

Tabla 3.15 Descripción de la página servidora SP_BuscarEnfermedad

Nombre: SP_InsertarCapitulo	
Tipo de clase: Server Page	
Descripción General: La clase <i>SP_InsertarCapitulo</i> es una clase que se ejecuta del lado del servidor en la Capa de Negocio. Su actividad es insertar en la base de datos los parámetros que recibe de la Capa de Presentación.	
Parámetros de Entrada	
Descripción	Tipo
codigo_capitulo	string
descripcion	string
incluye	string
excluye	string
observaciones	string

Tabla 3.16 Descripción de la página servidora SP_InsertarCapitulo

Nombre: SP_EliminarCapitulo	
Tipo de clase: Server Page	
Descripción General: La clase <i>SP_EliminarCapitulo</i> es una clase que se ejecuta del lado del servidor en la Capa de Negocio. Su actividad es eliminar de la base de datos los parámetros que recibe de la Capa de Presentación.	
Parámetros de Entrada	
Descripción	Tipo
idCapitulo	int
CodigoCapitulo	string
Cascada	int

Tabla 3.17 Descripción de la página servidora SP_EliminarCapitulo

Capítulo 3: Diseño del Sistema

Nombre: SP_ActualizarCapitulo	
Tipo de clase: Server Page	
Descripción General: La clase <i>SP_ActualizarCapitulo</i> es una clase que se ejecuta del lado del servidor en la Capa de Negocio. Su actividad es actualizar en la base de datos los parámetros que recibe de la Capa de Presentación.	
Parámetros de Entrada	
Descripción	Tipo
idCapitulo	int
CodigoCapitulo	string
Descripcion	string
Incluye	string
Excluye	string
Observaciones	string

Tabla 3.18 Descripción de la página servidora SP_ActualizarCapitulo

3.1.5 Diagrama de Clases Persistentes

Para el diseño de la base de datos del sistema, se parte de un diagrama de clases persistentes, definiendo la persistencia como la capacidad de un objeto de mantener su valor en el espacio y en el tiempo. Una vez conformado dicho diagrama se puede obtener el modelo de datos.

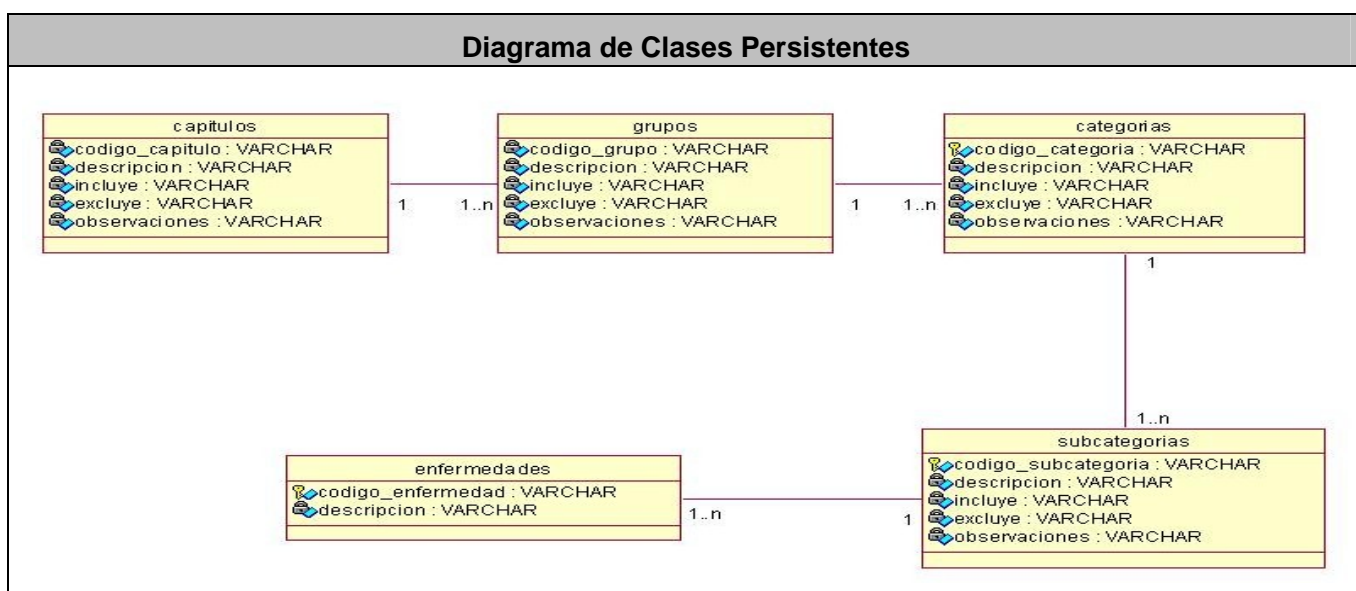


Figura 3.6 Diagrama de Clases Persistentes

Capítulo 3: Diseño del Sistema

3.1.6 Modelo de Datos

Un modelo de datos es un sistema formal y abstracto que permite describir los datos de acuerdo con reglas y convenios predefinidos. Es formal, pues los objetos del sistema se manipulan siguiendo reglas perfectamente definidas y utilizando exclusivamente los operadores definidos en el sistema, independientemente de lo que estos objetos y operadores puedan significar.

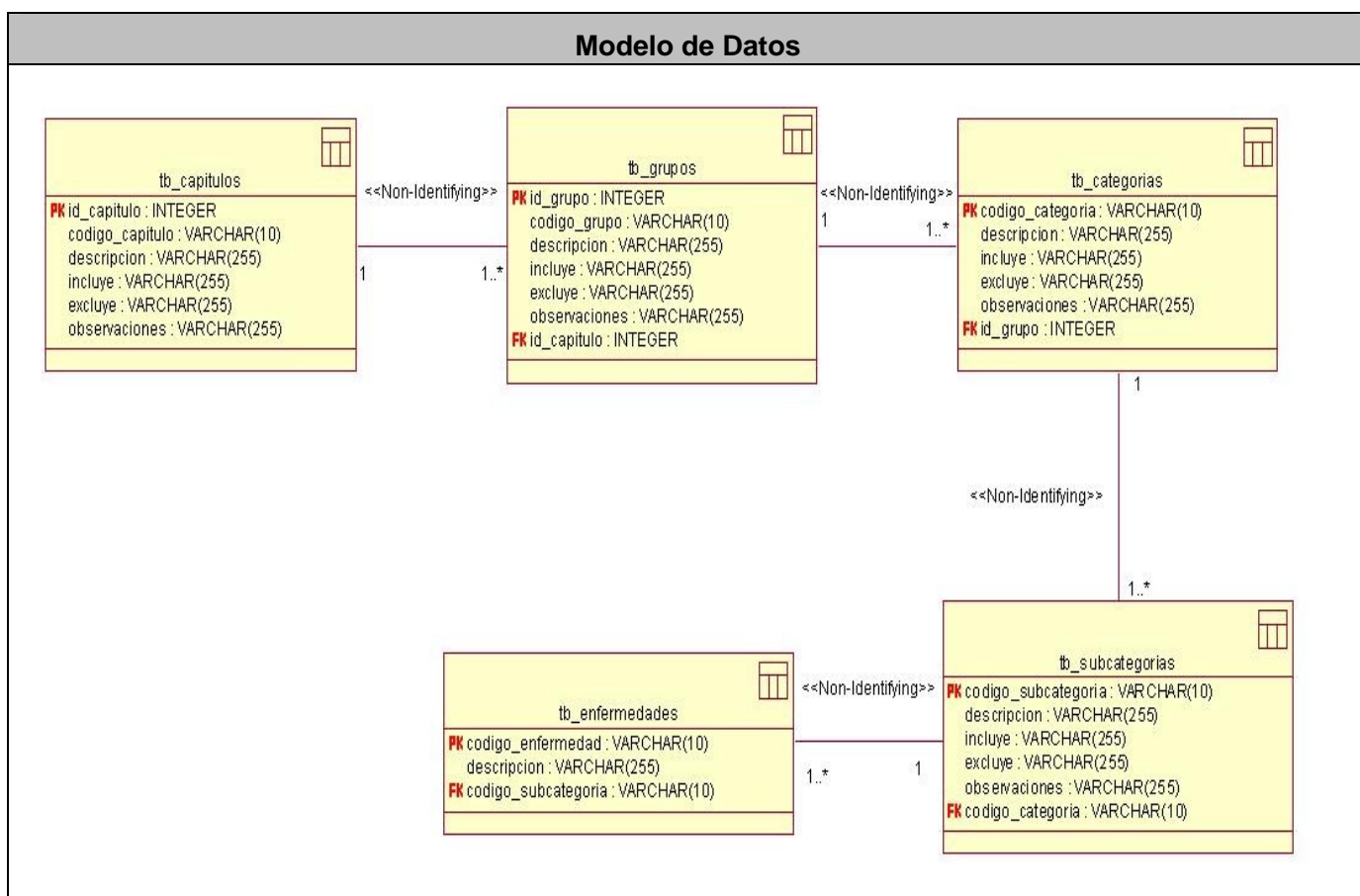


Figura 3.7 Diagrama de Modelo de Datos

Capítulo 3: Diseño del Sistema

3.1.7 Descripción de tablas y atributos

Nombre: tb_capitulos				
Descripcion: Tabla que almacenará los datos de todos los capítulos que existan en la CIE-10.				
Atributo	Tipo	Nulo	Tipo Llave	Descripción
id_capitulo	int	no	Primary Key	En este atributo se almacena el identificador del Capítulo, el cual sirve para garantizar unicidad en la Base de Datos y como es auto incremento manejados por la base de datos hace que la búsqueda sea más rápida.
codigo_capitulo	varchar(10)	no	Unique Key	Este atributo es el código que tiene el Capítulo en la CIE (Clasificación Internacional de Enfermedades).
descripcion	varchar (255)	no	Unique Key	Este atributo contiene el nombre Capítulo.
incluye	varchar (255)			Este atributo contiene información adicional de un Capítulo aparte de su Código y la Descripción.
excluye	varchar (255)			Este atributo contiene información adicional de un Capítulo aparte de su Código y la Descripción.
observaciones	varchar (255)			Este atributo contiene información adicional de un Capítulo aparte de su Código y la Descripción.

Tabla 3.19 Descripción de tb_capitulos

Nombre: tb_grupos				
Descripcion: Tabla que almacenará los datos de todos los grupos que existan en la CIE-10.				
Atributos	Tipo	Nulo	Tipo Llave	Descripción
id_grupo	int	no	Primary Key	En este atributo se almacena el identificador de Grupo, el cual sirve para garantizar unicidad en la Base de Datos y como es auto incremento manejados por la base de datos hace que la búsqueda sea más rápida.
codigo_grupo	varchar (10)	no	Unique Key	Este atributo es el código que tiene el Grupo en la CIE

Capítulo 3: Diseño del Sistema

				(Clasificación Internacional de Enfermedades).
descripcion	varchar(255)	no	Unique Key	Este atributo almacena el nombre del Grupo.
incluye	varchar(255)			Este atributo contiene información adicional de un Grupo aparte de su código y descripción.
excluye	varchar(255)			Este atributo contiene información adicional de un Grupo aparte de su código y descripción.
observaciones	VARCHAR(255)			Este atributo contiene información adicional de un Grupo aparte de su código y descripción.
id_capitulo	int	no	Foreign Key	En este atributo se almacena el identificador del Capítulo, el cual sirve para garantizar unicidad en la Base de Datos y como es auto incremento manejados por la base de datos hace que la búsqueda sea más rápida.

Tabla 3.20 Descripción de tb_grupos

Nombre: tb_categorias				
Descripción: Tabla que almacenará los datos de todas las categorías que existan en la CIE-10.				
Atributos	Tipo	Nulo	Tipo Llave	Descripción
codigo_categoria	varchar(10)	no	Primary Key	En este atributo se almacena el identificador de la Categoría y además es el código que tiene Categorías en la CIE (Clasificación Internacional de Enfermedades).
descripcion	varchar(255)	no	Unique Key	Este atributo contiene el nombre de la Categoría.
incluye	varchar(255)			Este atributo contiene información adicional de una Categoría aparte de su código y descripción.
excluye	varchar(255)			Este atributo contiene información adicional de una Categoría aparte de su código y descripción.
observaciones	varchar(255)			Este atributo contiene

Capítulo 3: Diseño del Sistema

				información adicional de una Categoría aparte de su código y descripción.
id_grupo	int	no	Foreign Key	En este atributo se almacena el identificador de Grupo, el cual sirve para garantizar unicidad en la Base de Datos y como es auto incremento manejados por la base de datos hace que la búsqueda sea más rápida.

Tabla 3.21 Descripción de tb_categorias

Nombre: tb_subcategorias				
Descripción: Tabla que almacenará los datos de todas las subcategorías que existan en la CIE-10.				
Atributos	Tipo	Nulo	Tipo Llave	Descripción
codigo_subcategoria	varchar(10)	no	Primary Key	En este atributo se almacena el identificador de la Subcategoría y además es el código que tiene Subcategorías en la CIE (Clasificación Internacional de Enfermedades).
descripcion	varchar(255)	no	Unique Key	Este atributo almacena el nombre de la Subcategoría.
incluye	varchar(255)			Este atributo contiene información adicional de una Subcategoría aparte de su código y descripción.
excluye	varchar(255)			Este atributo contiene información adicional de una Subcategoría aparte de su código y descripción.
observaciones	varchar(255)			Este atributo contiene información adicional de una Subcategoría aparte de su código y descripción.
codigo_categoria	varchar(10)	no	Foreign Key	En este atributo se almacena el identificador de la Categoría y además es el código que tiene Categorías en la CIE (Clasificación Internacional de Enfermedades).

Tabla 3.22 Descripción de tb_subcategorias

Capítulo 3: Diseño del Sistema

Nombre: tb_enfermedades				
Descripción: Tabla que almacenará los datos de todos los problemas de salud que existan en la CIE-10.				
Atributos	Tipo	Nulo	Tipo Llave	Descripción
codigo_enfermedad	varchar(10)	no	Primary Key	En este atributo se almacena el identificador del Problema de Salud y además es el código que tiene el Problema de Salud en la CIE (Clasificación Internacional de Enfermedades).
descripcion	varchar(255)	no	Unique Key	Este atributo almacena el nombre del Problema de Salud.
codigo_subcategoria	varchar(10)	no	Foreign Key	En este atributo se almacena el identificador de la Subcategoría y además es el código que tiene Subcategorías en la CIE (Clasificación Internacional de Enfermedades).

Tabla 3.23 Descripción de tb_enfermedades.

Conclusiones

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

CAPÍTULO 4: IMPLEMENTACIÓN

Introducción

El propósito fundamental de este capítulo es definir cómo desarrollar la arquitectura prevista durante el diseño, representando el sistema como un todo. Se implementan las clases y subsistemas definidos en el capítulo anterior en términos de componentes. Se modelan los diagramas de componentes y de despliegue, quedando conformado el modelo de implementación del sistema, dando una visión de cómo quedará construida y distribuida la aplicación. Además,, se proporciona una detallada explicación de los estándares de diseño, codificación y tratamiento de errores a utilizar.

4.1 Modelo de Implementación

El Modelo de Implementación es comprendido por un conjunto de componentes y subsistemas que constituyen la composición física de la implementación del sistema. Este modelo describe la relación que existe desde los paquetes y clases del modelo de diseño a subsistemas y componentes físicos. cómo se organizan estos últimos de cuerdo con los mecanismos de estructuración, al lenguajes o lenguajes utilizados y cómo dependen unos de otros.

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

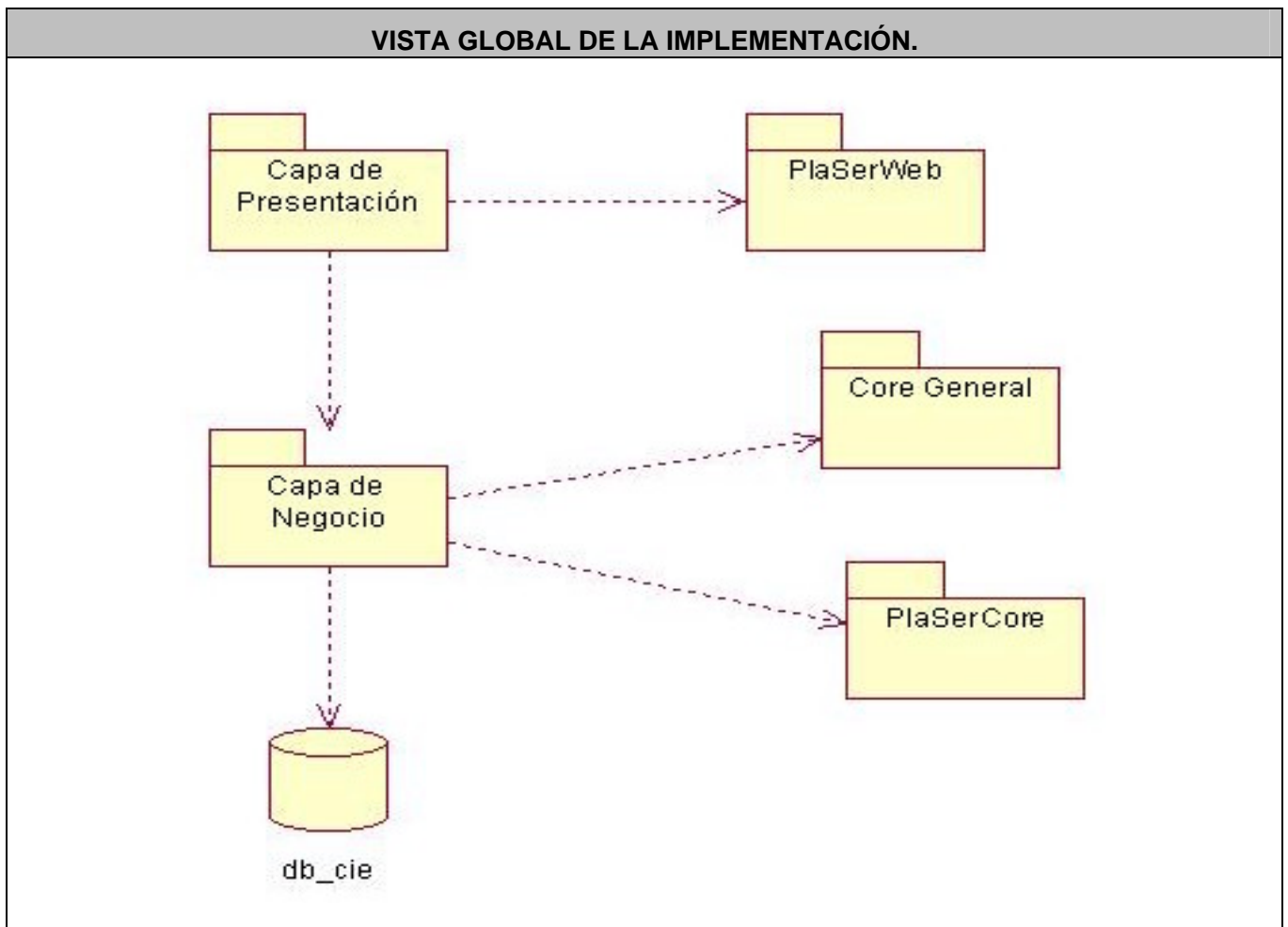


Figura 4.1 Vista Global de la Implementación

4.1.1 Diagrama de Componentes

Un componente es el empaquetamiento físico de los elementos de un modelo, como son las clases del modelo de diseño, los mismos son creados, modificados o eliminados en el proceso de implementación y constituyen la versión del producto.

El siguiente diagrama muestra los diferentes componentes, separados por capas, y sus relaciones del Caso de Uso BuscarGeneral:

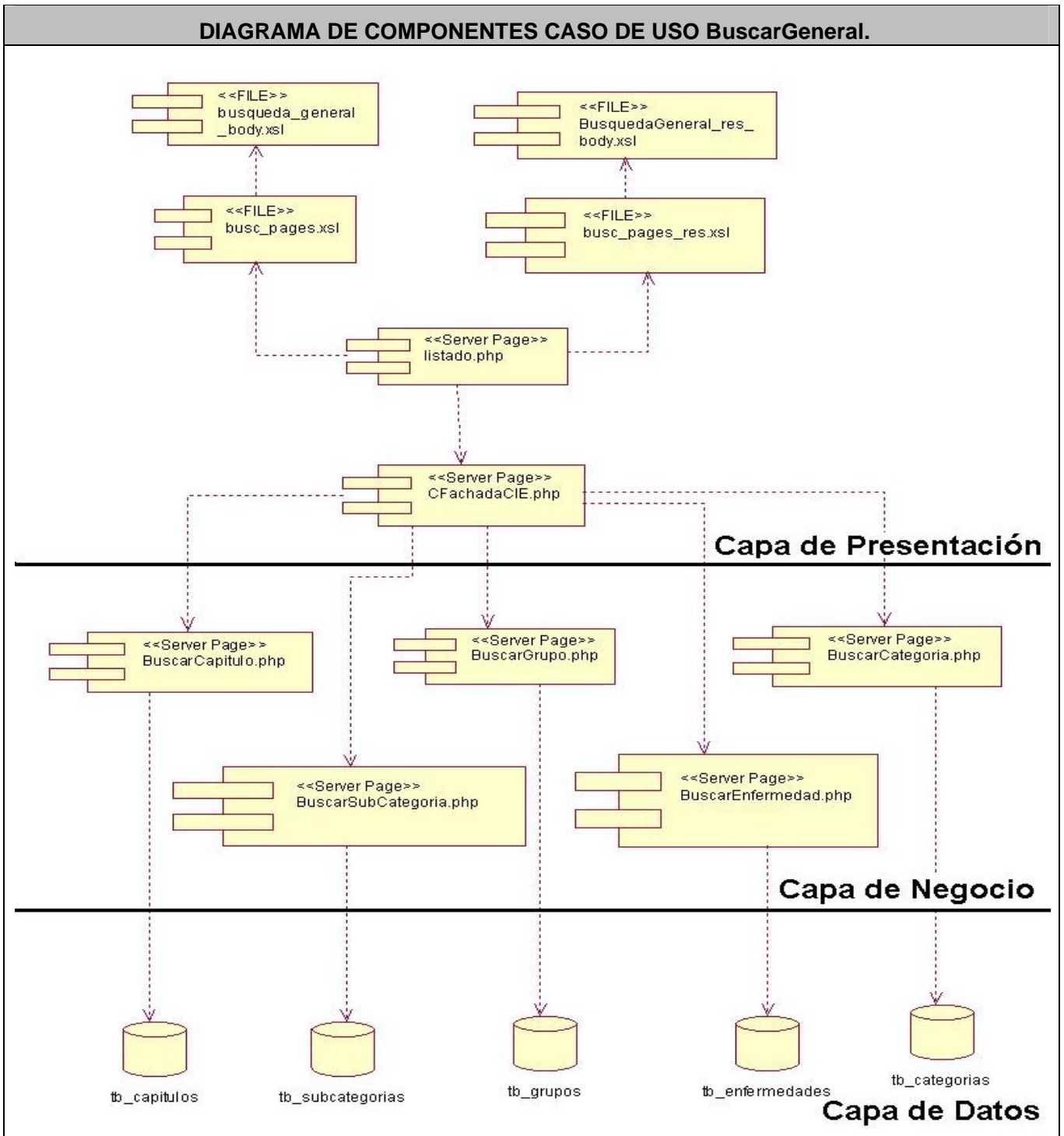


Figura 4.2 Diagrama de Componentes Caso de Uso BuscarGeneral

4.1.2 Diagrama de Despliegue.

El diagrama de despliegue describe la distribución física del sistema en términos de cómo se distribuye la funcionalidad entre los nodos. Los nodos representan recursos de cómputos: procesadores o dispositivos de hardware.

De manera general la distribución de RCIE quedará de la siguiente manera:

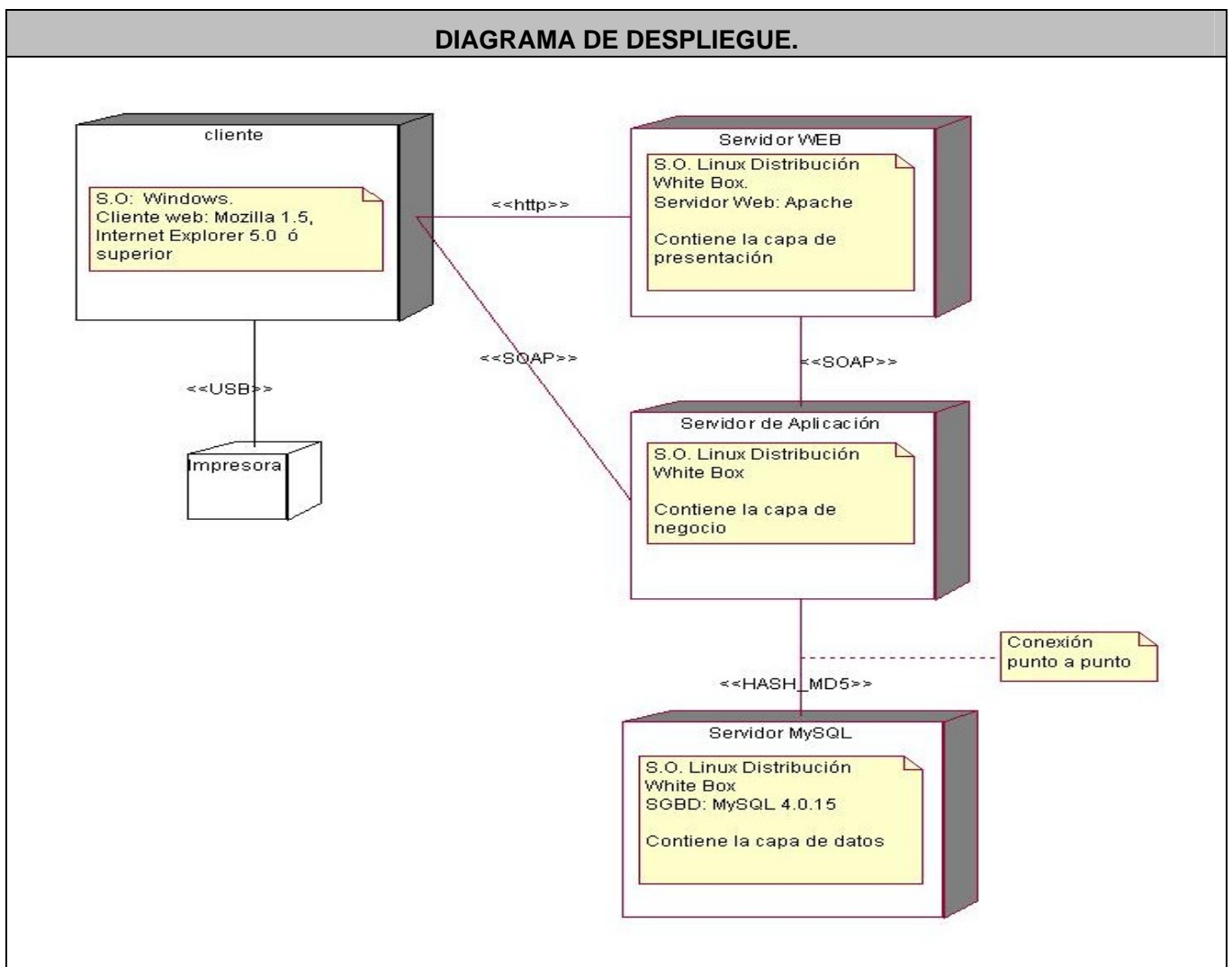


Figura 4.3 Diagrama de Despliegue

Capítulo 4: Implementación

4.2 Descripción de los Métodos (agentes o métodos más complejos)

Nombre: Datos.php	
Tipo de clase: Server Page	
Descripción General: Este método permitirá conocer dado un id ya sea de un Capítulo, Grupo, Categoría, Subcategoría o Enfermedad los datos referentes al mismo y además devolverá toda la dependencia superior que tenga cada elemento solicitado.	
Parámetros de Entrada	
Descripción	Tipo
id_capitulo	int
Id_grupo	int
Id_categoria	string
Id_subcategoría	string
Id_enfermedad	string


Tabla 4.1 Descripción del método Datos.php.

4.3 Estándares de diseño, codificación y tratamiento de errores

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

4.3.1 Estándares de Diseño

Para el diseño de las páginas web de la aplicación del módulo RCIE se siguió las pautas y el mismo diseño que en el Registro Informatizado de Salud (RIS). Esto trae como ventajas una uniformidad en la estructura de las páginas web de todos los módulos integrados al RIS. Para lograr esta uniformidad se debe aplicar una hoja de estilo en cascada en inglés (Cascading Style Sheets CSS), todas las páginas deben tener una resolución de 800X600 pixel, incluir una serie de imágenes estándares que define el RIS y seguir una serie de pautas, que algunas de estas son:

- En caso de existir varias opciones de búsqueda se pone siempre en la primera fila de la tabla una imagen ( searchPersonal.gif) seguido por el texto **Buscar por:** (hipervínculo hecho en

Capítulo 4: Implementación

JavaScript que permite recoger o expandir la tabla, aunque siempre aparece expandido al abrir la página).


En la fila de los botones:

Parte derecha: Con estilo **grisclaro**.

Parte izquierda: Con estilo **verdelimon**.


Expandido:

🔍 Búsqueda General

 Buscar por:	
Código <input type="text"/>	Descripción <input type="text"/>
<input type="button" value="Buscar"/>	
Introduzca algún criterio de búsqueda y presione "Buscar".	

Recogido:

🔍 Búsqueda General

 Buscar por:
Introduzca algún criterio de búsqueda y presione "Buscar".

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

Capítulo 4: Implementación

🔍 Resultados - SUBCATEGORÍAS

Código	Descripción
A00.0	Cólera debido a Vibrio cholerae O1, biotipo cholerae
A00.1	Cólera debido a Vibrio cholerae O1, biotipo El Tor
A00.9	Cólera, no especificado

1- 3/3

Exportar a ▼

Volver

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

🔍 Resultados - SUBCATEGORÍAS

Código	Descripción
A00.0	Cólera debido a Vibrio cholerae O1, biotipo cholerae
A00.1	Cólera debido a Vibrio cholerae O1, biotipo El Tor
A00.9	Cólera, no especificado
B00.0	Eczema herpético
B00.1	Dermatitis vesicular herpética
B00.2	Gingivostomatitis y faringoamigdalitis herpética
B00.3+	Meningitis herpética (G02.0*)
B00.4+	Encefalitis herpética (G05.1*)
B00.5	Oculopatía herpética(B00.5+)
B00.7	Enfermedad herpética diseminada

1- 10/117

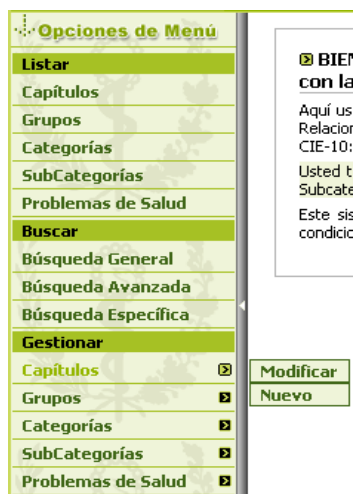
Exportar a ▼

Página # 1 ▼

Próximo Último

Volver

- El menú estará hecho en JavaScript, desplegándose hacia la derecha, de la siguiente forma:



4.3.2 Estándares de Codificación

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

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

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

Se hará uso de los arreglos predefinidos para el manejo de los valores enviados por el usuario `$_GET`, `$_POST`, `$_FILES` evitando el uso de `$_REQUEST`.

Capítulo 4: Implementación

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: BuscarCapitulo

Para comentar el código se utilizará, en el caso de una línea, al final de la misma el carácter “//” y seguido el comentario y en el caso de un bloque se utilizará los caracteres “/* */”.

El idioma de las clases auxiliares como sesión y error, será el inglés para garantizar la homogeneidad con las programadas en este ámbito en el mundo, en el caso de los Servicios Web y la interface de administración se usará el español para esclarecer los objetivos de cada método o script a utilizar.

Se utilizará el operador “?” para sentencias cortas, preferiblemente que ocupen una sola línea. La sentencia switch siempre tendrá la opción default y se evitará el uso de continue y break, ya que podrían perder la vista lógica del código fuente.

El almacenamiento de la información será en scripts SQL para construir la base de datos e interactuar con ella desde las aplicaciones.

Las palabras correspondientes a las sentencias SQL y sus parámetros deben ir en mayúsculas.

En las consultas de inserción se debe colocar siempre el nombre de los campos en los cuales desea escribir en la tabla.

Los nombres de las tablas deben ir en minúsculas y cada palabra separada por línea abajo "_". (Ejemplo: Id_nombre_tabla)

En el caso de los XSL será con el mismo nombre que el fichero de la capa de presentación.

Capítulo 4: Implementación

Los controles seguirán el siguiente tratamiento:

Control	Prefijo	Ejemplo
Botón	Btn	btnAceptar
Etiqueta	Lbl	lblNombre
Lista/Menú	Mn	mnPrincipal
Campo de Texto	Txt	TxtFecha
Botón de Opción	Opt	OptSexo
Casilla de Verificación	Chx	ChxBorrar

Tabla 4.2 Estándares para los controles

Las páginas HTML se harán sin incluir código y todas las funciones JavaScript que se usarán se escribirán dentro de ficheros “.js”.

Para la capa de datos se tiene que nombrar la base de datos poniendo el identificador del proyecto “APS” seguido del carácter “_” y del nombre del módulo. Ejemplo: bd_rcie

Las tablas se identificarán con el acrónimo tb_Nombre, ejemplo: tb_capitulos.

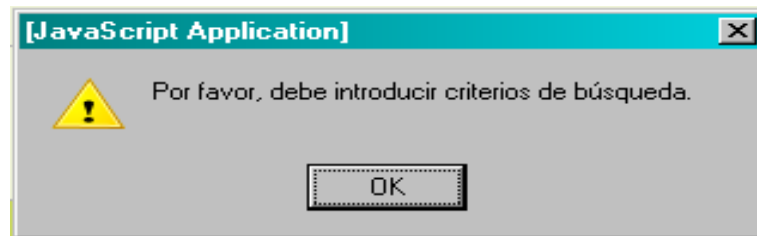
Los campos de la base de datos se nombrarán igual que las variables.

Cada módulo definirá un espacio de nombre (namespace), siguiendo la siguiente estructura <http://APS/RCIE/NombreMetodo>.

4.3.3 Tratamiento de Errores

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

Capítulo 4: Implementación

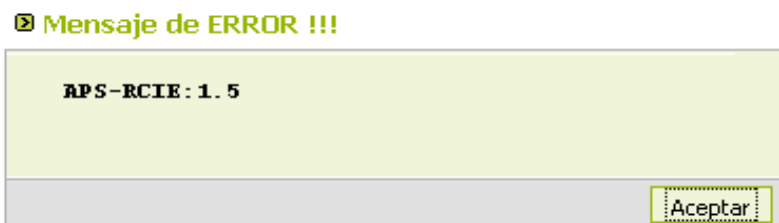


Otros errores en la capa de negocio serán tratados devolviendo un SOAP_FAULT, cuyos elementos FaultCode, FaultString, FaultAutor se describirán a continuación:

FaultCode:

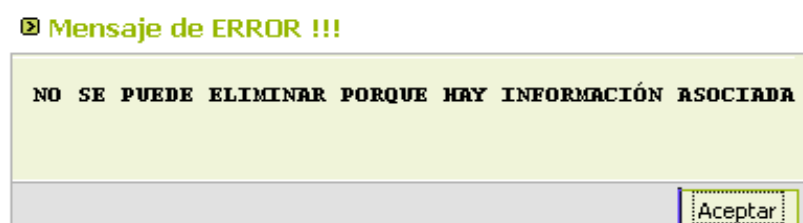
Código de texto utilizado para indicar la clase de error, codificado de la siguiente manera.

Código del proyecto-código del modulo (:) número del método (.) número del error. Ejemplo: APS-RCIE: 1.5 que indica error 5 en el método 1 del módulo Registro de la Clasificación Internacional de Enfermedades y Problema Relacionado con la Salud perteneciente al Proyecto APS.



FaultString:

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

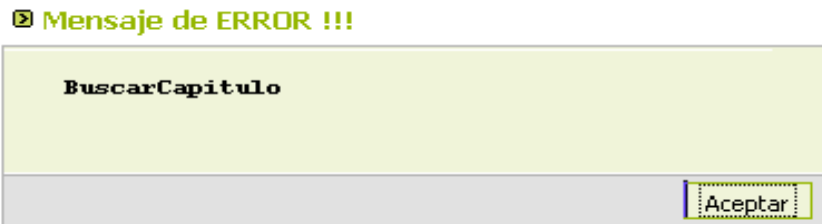


Capítulo 4: Implementación

FaultActor:

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

Ejemplo:



Conclusiones

En este capítulo se mostraron los resultados obtenidos durante la etapa de implementación; quedando conformados los diagramas de componentes y despliegue, que representan cómo construir y distribuir el sistema, lo que permite la obtención de un producto que cumpla con las funcionalidades propuestas. Se detallan los métodos de implementación más complejos, así como el tratamiento de errores, los principios de codificación y de diseño empleados.

CONCLUSIONES

Luego de un estudio realizado sobre los procesos de gestión de la información del Sistema Nacional de Salud y en aras de apoyar la informatización de dicho sector, así como el cumplimiento de los objetivos y tareas trazados, se obtuvieron las siguientes conclusiones:

- Se realizó un estudio detallado del proceso de gestión de la CIE-10 en el Sistema Nacional de Salud, definiendo los principales problemas existentes en el país, así como los conceptos asociados al dominio de los mismos.
- Se investigó sobre sistemas existentes a nivel internacional y nacional, que pudieran resolver el problema planteado, realizando un análisis comparativo entre los mismos. Además, se estudiaron las tendencias, tecnologías y herramientas a utilizar para el desarrollo de la solución propuesta.
- Se documentaron los flujos de trabajo: Modelamiento de Negocio, Requerimientos, Diseño e Implementación.
- Se diseñó una aplicación que cumple con los requerimientos trazados y con la arquitectura definida por el MINSAP.
- Se implementó una aplicación web que permite búsquedas dinámicas, gestionar la información atendiendo a la estructura y taxonomía de la CIE-10 y brindar servicios a las demás aplicaciones integradas a SISalud. Esta se encuentra en fase de despliegue en INFOMED. Posibilitando con su uso, que la Dirección Nacional de Estadísticas del MINSAP cuente con una herramienta que permita el proceso de diagnóstico homogéneo en cada una de las aplicaciones informáticas del Sistema Nacional de Salud.

RECOMENDACIONES

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

- Implementar una nueva opción de búsqueda en el sistema, que muestre la estructura de la CIE-10 según su taxonomía.
- Habilitar la opción “Conozca CIE” diseñada en la página principal de SISalud, para que cualquier usuario del dominio de salud, pueda adquirir conocimientos relacionados con la CIE-10 y su estructura.
- Publicar los servicios web de RCIE para que puedan ser consumidos por cualquier aplicación informática que necesite de los mismos.
- Incluir en la concepción del componente de avisos de SISalud los mensajes generados por actualizaciones que se realicen en el RCIE.
- Poner en práctica el plan de capacitación necesario para que el personal de salud de las diferentes unidades asistenciales se familiaricen con el sistema.

REFERENCIAS BIBLIOGRÁFICAS

- [1]. **Delgado Ramos, Ariel y Vidal Ledo, María.** <http://bvs.sld.cu>. [En línea] 2006. [Citado el: 02 de febrero de 2007.] http://bvs.sld.cu/revistas/spu/vol32_3_06/spu15306.htm#cargo.
- [2]. Idem a la Referencia 1.
- [3]. **Castro Ruz, Fidel.** La idea esencial es acercar los servicios primarios a los ciudadanos. *Periódico Granma*. 2003.
- [4]. **Delgado Ramos, Ariel, Cabrera Hernández, Mirna y Juncal, Virginia.** <http://www.sld.cu>. [En línea] 28 de septiembre de 2006. [Citado el: 06 de febrero de 2007.] <http://www.sld.cu/sitios/dne/temas.php?idv=7252>.
- [5]. **Martínez Morales, Miguel Ángel, y otros.** <http://www.sld.cu>. [En línea] 2005. [Citado el: 04 de febrero de 2007.] http://www.sld.cu/galerias/pdf/sitios/dne/nro2_estudio-puente.pdf.
- [6]. **Marín Díaz, Miguel Enrique.** *Fundamentos del Sistema de Salud Pública en Cuba para estudiantes de Informática*. La Habana : s.n., 2006.
- [7]. **de la Osa, Jose A.** La semilla del desarrollo de la salud pública en Cuba. *Periódico Granma*. 2004.
- [8]. Idem a la Referencia 7.
- [9]. Idem a la Referencia 7.
- [10]. **Dotres Martínez, Carlos.** <http://www.infomed.sld.cu>. [En línea] junio de 1997. [Citado el: 12 de febrero de 2007.] <http://www.infomed.sld.cu/discursos/telem.html>.
- [11]. **Delgado Ramos, Ariel.** *Informatización del Sistema Nacional de Salud*. La Habana : Dirección Nacional de Registros Médicos y Estadísticas de Salud. Ministerio de Salud Pública, 2006.
- [12]. Idem a la Referencia 11.
- [13]. **Durand Llamas, Catherine, Reyna Soler, Luis Mariano.** *Centro de Control para el Sistema de Información para la Salud*. Trabajo de Diploma para optar por el Título de Ingeniero Informático. Instituto Superior Politécnico José Antonio Echeverría. La Habana, Cuba. Julio 2005.
- [14]. **INFOMED.** <http://www.sld.cu>. [En línea] 1999-2000. [Citado el: 3 de febrero de 2007]. <http://www.sld.cu/acerca/acercade.html#que>
- [15] Idem a la Referencia 13.
- [16]. **Cabrera Hernandez, Mirna, y otros.** *Propuesta de Esquema Sistema Integral de Salud (SiSalud)*. La Habana : s.n., 2006.
- [17]. Idem a la Referencia 16.

Referencias Bibliográficas

- [18]. **Colectivo de Autores OPS**. <http://www.virtual.epm.br>. [En línea] OPS. [Citado el: 03 de febrero de 2007.] http://www.virtual.epm.br/material/healthcare/spanish/parte_f.htm.
- [19]. Idem a la Referencia 18.
- [20]. **Cuellar Álvarez, Dra. Rosabel, Martínez Morales, Dr. Miguel Ángel, Zacca Peña, Dr. Eduardo**. La Clasificación Internacional de Enfermedades y Problemas Relacionados con la Salud (CIE). <http://www.uvs.sld.cu>. [En línea] Universidad Virtual de Salud. Cuba, 18 de Mayo de 2006. [Citado el: 7 de Febrero de 2007.] <http://www.uvs.sld.cu/profesores/supercursos/plonearticlemultipage.2006-05-18.8206031544/>
- [21]. **Castillo Salgado, Dr. Carlos**. Antecedentes e importancia de la CIE. <http://ais.paho.org>. [En línea] 2003. [Citado el: 3 de febrero de 2007.] <http://ais.paho.org/cie/index.htm?xml=antecedents.htm>
- [22]. Idem a la Referencia 21.
- [23]. Idem a la Referencia 21.
- [24]. **Castillo Salgado, Dr. Carlos**. Diseño de la CIE. <http://ais.paho.org>. [En línea] 2003. [Citado el: 3 de febrero de 2007.] <http://ais.paho.org/cie/index.htm?xml=ciedesign.htm>
- [25]. Idem a la Referencia 24.
- [26]. **OPS**. PROPÓSITO Y APLICABILIDAD DE LA CIE. <http://ais.paho.org>. [En línea] 2003. [Citado el: 16 de febrero de 2007.] <http://ais.paho.org/cie/index.htm?xml=purpose.htm>.
- [27]. **Derivet Thureaux, Denis**. *Modelación de un Sistema Informático para la gestión de la Clasificación Internacional de Enfermedades*. La Habana : s.n., 2006.
- [28]. Idem a la Referencia 27.
- [29]. **Castillo Salgado, Dr. Carlos**. <http://ais.paho.org>. [En línea] 2003. [Citado el: 26 de enero de 2007.] <http://ais.paho.org/cie/index.htm?xml=tendon.htm>.
- [30]. **Castillo Salgado, Dr. Carlos**. <http://ais.paho.org>. [En línea] 2003. [Citado el: 26 de enero de 2007.] <http://ais.paho.org/cie/index.htm?xml=intercod.htm>.
- [31]. **Svetliza, Javier**. <http://www.softmedico.com>. [En línea] [Citado el: 02 de marzo de 2007.] <http://www.softmedico.com/CIE-CIAP.html>.
- [32]. **Svetliza, Javier**. <http://www.abcdatos.com>. [En línea] 30 de mayo de 2007. [Citado el: 01 de junio de 2007.] <http://www.abcdatos.com/programas/programa/z6951.html>.
- [33]. **Martínez Morales, Miguel Ángel, y otros**. <http://www.sld.cu>. [En línea] 2005. [Citado el: 4 de febrero de 2007.] http://www.sld.cu/galerias/pdf/sitios/dne/nro2_estudio-puente.pdf.

Referencias Bibliográficas

- [34]. **García Arenas, María Isabel.** <http://geneura.ugr.es>. [En línea] [Citado el: 4 de febrero de 2007.] <http://geneura.ugr.es/~maribel/xml/introduccion/index.shtml#12>.
- [35]. Idem a la Referencia 34.
- [36]. **Guillermo Valle, Jose.** <http://www.monografias.com>. [En línea] 2005. [Citado el: 3 de febrero de 2007.] <http://www.monografias.com/trabajos24/arquitectura-cliente-servidor/arquitectura-cliente-servidor.shtml#intro>.
- [37]. Idem a la Referencia 36.
- [38]. Idem a la Referencia 36.
- [39]. Idem a la Referencia 36.
- [40]. **Cuenca, Carlos Luis.** <http://www.desarrolloweb.com>. [En línea] [Citado el: 06 de febrero de 2007.] <http://www.desarrolloweb.com/articulos/1112.php>.
- [41]. Idem a la Referencia 40.
- [42]. **Luke Welling, Laura Thomson.** *Desarrollo web con PHP y MySQL*. s.l. : ANAYA Multimedia, 2003.
- [43]. Idem a la Referencia 42.
- [44]. **Alvarez, Miguel Angel.** <http://www.desarrolloweb.com>. [En línea] [Citado el: 06 de febrero de 2007.] <http://www.desarrolloweb.com/articulos/25.php>.
- [45]. Idem a la Referencia 44.
- [46]. Idem a la Referencia 44.
- [47]. Idem a la Referencia 44.
- [48]. Idem a la Referencia 42.
- [49]. **Rational Corporation.** *Rational Unified Process* 2003, Rational Corporation.
- [50]. Idem a la Referencia 49.
- [51]. Idem a la Referencia 49.
- [52]. Idem a la Referencia 49.
- [53]. **clikear.com.** <http://www.clikear.com>. *Desarrollo Orientado a Objetos con UML*. [En línea] 2004. [Citado el: 06 de 03 de 2007.] <http://www.clikear.com/manuales/uml>.
- [54]. Idem a la Referencia 53.
- [55]. **Jacobson, Ivar, Booch, Grady y Rumbaugh, James.** *El Proceso Unificado de Desarrollo de Software*. La Habana : Felix Varela, 2004.
- [56]. Idem a la Referencia 55.
- [57]. Idem a la Referencia 55.

Referencias Bibliográficas

[58]. **James JACOBSON, Ivar y BOOCH, Grady.** *El lenguaje unificado de modelado.* 2000.

[59]. Idem a la Referencia 55.

[60]. **MeRinde.** <http://merinde.rinde.gob.ve>. [En línea] 2007. [Citado el: 24 de abril de 2007.]

http://merinde.rinde.gob.ve/index.php?option=com_content&task=view&id=92&Itemid=296.

BIBLIOGRAFÍA

1. **Alvarez, Miguel Angel.** <http://www.desarrolloweb.com>. [En línea] [Citado el: 06 de febrero de 2007.] <http://www.desarrolloweb.com/articulos/25.php>.
2. **Cabrera Hernandez, Mirna, y otros.** *Propuesta de Esquema Sistema Integral de Salud (SiSalud)*. La Habana : s.n., 2006.
3. **Castillo Salgado, Dr.Carlos.** Diseño de la CIE. <http://ais.paho.org>. [En línea] 2003. [Citado el: 3 de febrero de 2007.]
4. **Castro Ruz, Fidel.** La idea esencial es acercar los servicios primarios a los ciudadanos. *Periódico Granma*. 2003.
5. **Cuellar Álvarez, Dra. Rosabel, Martínez Morales, Dr. Miguel Ángel y Zacca Peña, Dr. Eduardo.** La Clasificación Internacional de Enfermedades y Problemas relacionados con la Salud (CIE). <http://www.uvs.sld.cu>. [En línea] Universidad Virtual de Salud. Cuba, 18 de Mayo de 2006. [Citado el: 7 de Febrero de 2007.] <http://www.uvs.sld.cu/profesores/supercursos/plonearticlemultipage.2006-05-18.8206031544/>.
- 6.. **Cuenca, Carlos Luis.** <http://www.desarrolloweb.com>. [En línea] [Citado el: 06 de febrero de 2007.] <http://www.desarrolloweb.com/articulos/1112.php>.
7. **clikear.com.** <http://www.clikear.com>. *Desarrollo Orientado a Objetos con UML*. [En línea] 2004. [Citado el: 06 de 03 de 2007.] <http://www.clikear.com/manuales/uml>.
8. **de la Osa, Jose A.** La semilla del desarrollo de la salud pública en Cuba. *Periódico Granma*. 2004.
9. **Delgado Ramos, Ariel.** *Informatización del Sistema Nacional de Salud*. La Habana : Dirección Nacional de Registros Médicos y Estadísticas de Salud. Ministerio de Salud Pública, 2006.
10. **Derivet Thaux, Denis.** *Modelación de un Sistema Informático para la gestión de la Clasificación Internacional de Enfermedades*. La Habana : s.n., 2006.
11. **Dotres Martínez, Carlos.** <http://www.infomed.sld.cu>. [En línea] junio de 1997. [Citado el: 12 de febrero de 2007.] <http://www.infomed.sld.cu/discursos/telem.html>.
12. **Durand Llamos, Catherine, Reyna Soler, Luis Mariano.** *Centro de Control para el Sistema de Información para la Salud*. Trabajo de Diploma para optar por el Título de Ingeniero Informático. Instituto Superior Politécnico José Antonio Echeverría. La Habana, Cuba. Julio 2005.

13. **García Arenas, María Isabel.** <http://geneura.ugr.es>. [En línea] [Citado el: 4 de febrero de 2007.] <http://geneura.ugr.es/~maribel/xml/introduccion/index.shtml#12>.
14. **Guillermo Valle, Jose.** <http://www.monografias.com>. [En línea] 2005. [Citado el: 3 de febrero de 2007.] <http://www.monografias.com/trabajos24/arquitectura-cliente-servidor/arquitectura-cliente-servidor.shtml#intro>.
15. **INFOMED.** <http://www.sld.cu>. [En línea] 1999-2000. [Citado el: 3 de febrero de 2007.] <http://www.sld.cu/acerca/acercade.html#que>.
16. **Jacobson, Ivar, Booch, Grady y Rumbaugh, James.** *El Proceso Unificado de Desarrollo de Software*. La Habana : Felix Varela, 2004.
17. **James JACOBSON, Ivar y BOOCH, Grady.** *El lenguaje unificado de modelado*. 2000.
18. **Lago, Ramiro.** [Http://www.proactiva-calidad.com](http://www.proactiva-calidad.com). *Patrón "Modelo-Vista-Controlador"*. [En línea] abril de 2007. [Citado el: 23 de mayo de 2007.] <http://www.proactiva-calidad.com/java/patrones/mvc.html>.
19. **Luke Welling, Laura Thomson.** *Desarrollo web con PHP y MySQL*. s.l. : ANAYA Multimedia, 2003.
20. **MeRinde.** <http://merinde.rinde.gob.ve>. [En línea] 2007. [Citado el: 24 de abril de 2007.] http://merinde.rinde.gob.ve/index.php?option=com_content&task=view&id=92&Itemid=296.
21. **Martínez Morales, Miguel Ángel, y otros.** <http://www.sld.cu>. [En línea] 2005. [Citado el: 4 de febrero de 2007.] http://www.sld.cu/galerias/pdf/sitios/dne/nro2_estudio-puente.pdf.
22. **Molpeceres, Alberto.** <http://www.javahispano.com>. *Diseño de software con patrones (parte 4)*. [En línea] [Citado el: 23 de mayo de 2007.] http://www.javahispano.org/download/articulos/patrones_4.pdf;jsessionid=12932BD7FDED8F8266E3661B55FAD557.
23. **Morea, Lucas Morea.** <http://www.monografias.com>. [En línea] 1997. [Citado el: 5 de 03 de 2007.] <http://www.monografias.com/trabajos5/insof/insof.shtml>.
24. **Pompas Sourd, Frank, y otros.** *Sistema Informático para la Atención Primaria de Salud (APUS)*. La Habana : s.n., 2001.
25. **Svetliza, Javier.** <http://www.abcdatos.com>. [En línea] 30 de mayo de 2007. [Citado el: 01 de junio de 2007.] <http://www.abcdatos.com/programas/programa/z6951.html>.

ANEXOS

Anexo 1. Revisiones de la Clasificación Internacional de Enfermedad.



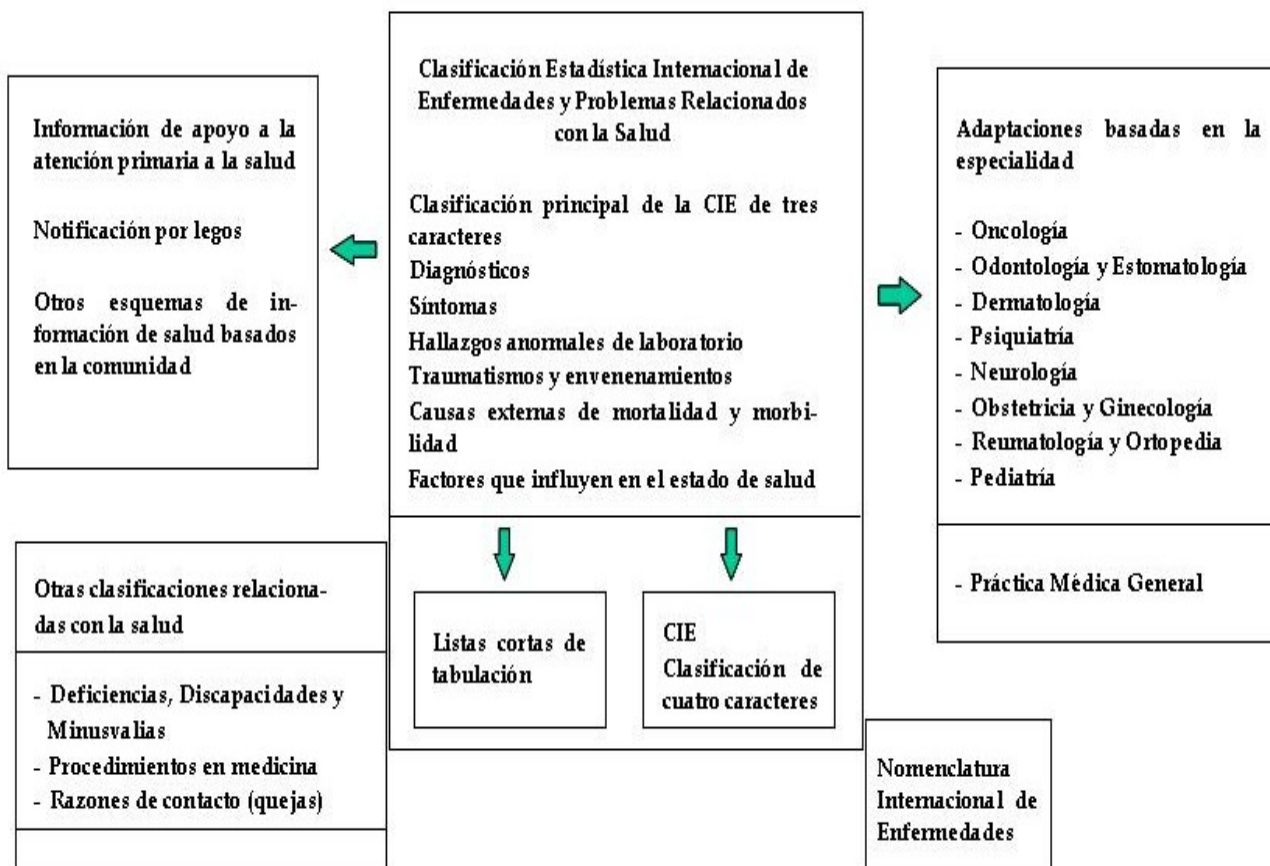
REVISIONES DE LA CLASIFICACIÓN INTERNACIONAL DE ENFERMEDADES, AÑO DE ADOPCIÓN Y CATEGORÍAS.

Revisión CIE	Año Adopción	Enfermedades	Causas Externas	Total
I	1900	157	22	179
II	1909	157	32	189
III	1920	166	39	205
IV	1929	164	36	200
V	1938	164	36	200
VI	1948	769	153	1010
VII	1955	800	153	1041
VIII	1965	858	182	1088
IX	1975	909	182	1178
X	1989	1575	373	2032

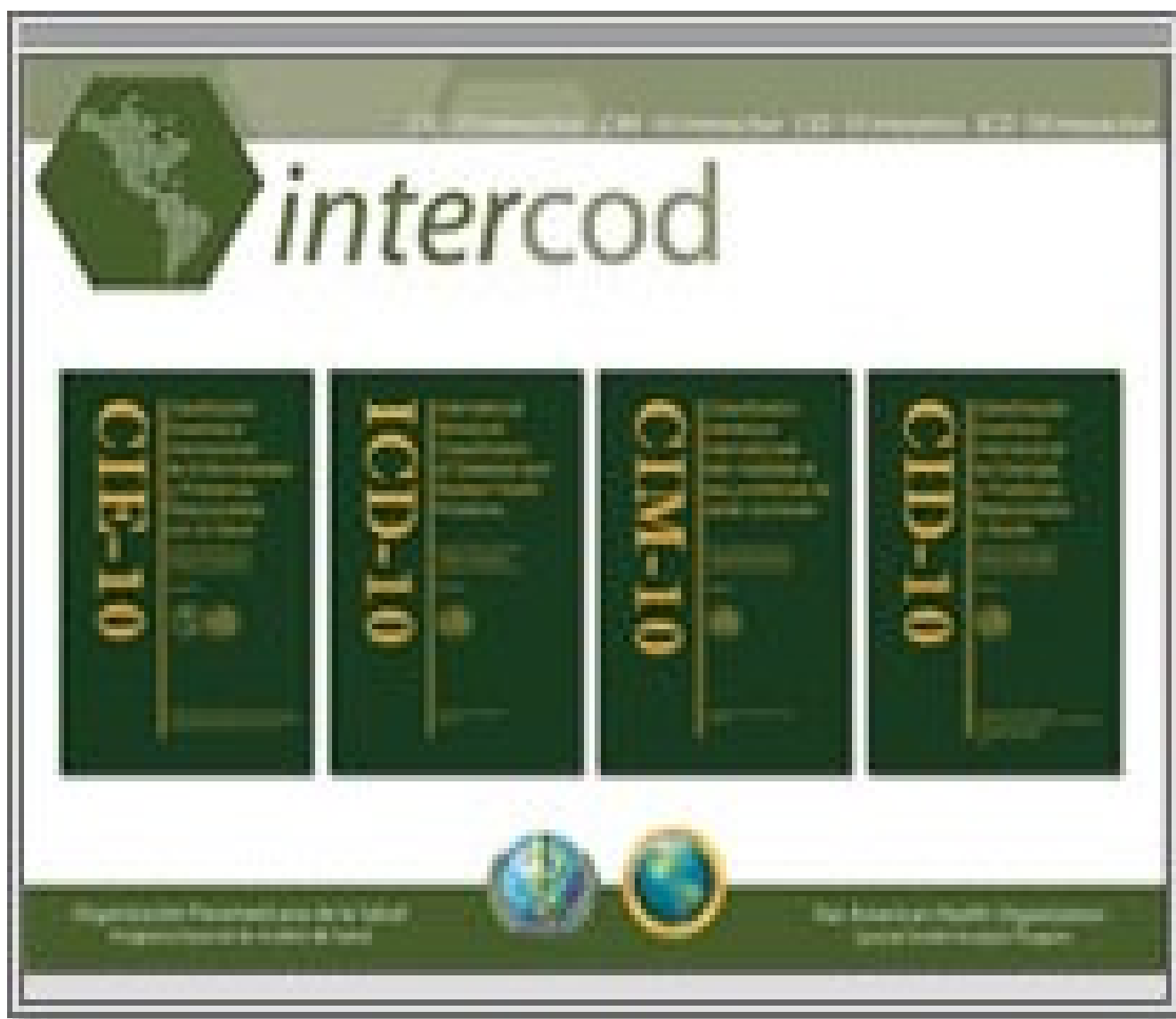


Anexo 2. Familia de Clasificaciones.

FAMILIA DE CLASIFICACIONES



Anexo 3. INTERCOD



Anexo 4. CIE-CIEAP

*** BUSCARÁS VALORES DENTRO DEL CIE 10 ***

BUSQUEDA ALFABETICA DE DIAGNOSTICO

INGRESE EL TERMINO A BUSCAR CIE 10

CODIGO	DIAGNOSTICO
D63.0 *	anemia en enfermedad neoplasica (c00d48+)
G13.2 *	atrofia sistematica que afecta primariamente el sistema nervioso central en el mixedema (e00.1+ e03.+)
G53.1 *	paralisis multiples de los nervios craneales en enfermedades infecciosas y parasitarias clasificadas en otra parte (a00b99+)
G53.3 *	paralisis multiple de los nervios craneales, en enfermedades neoplasicas (c00d48+)
G63.3 *	polineuropatia en otras enfermedades endocrinas y metabolicas (e00e07+ e15e16+ e20e34+ e70e89+)
G63.6 *	polineuropatia en otros trastornos osteomusculares (m00m25+,m40m96+)
G73.2 *	otros sindromes miastenicos en enfermedad neoplasica (c00d48+)
G94.0 *	hidrocefalo en enfermedades infecciosas y parasitarias clasificadas en otra parte (a00b99+)
G94.1 *	hidrocefalo en enfermedad neoplasica (c00d48+)
H19.1 *	queratitis y queratoconjuntivitis por herpes simple (b00.5 I)

Anexo 5. CIE-CIEAP FULL

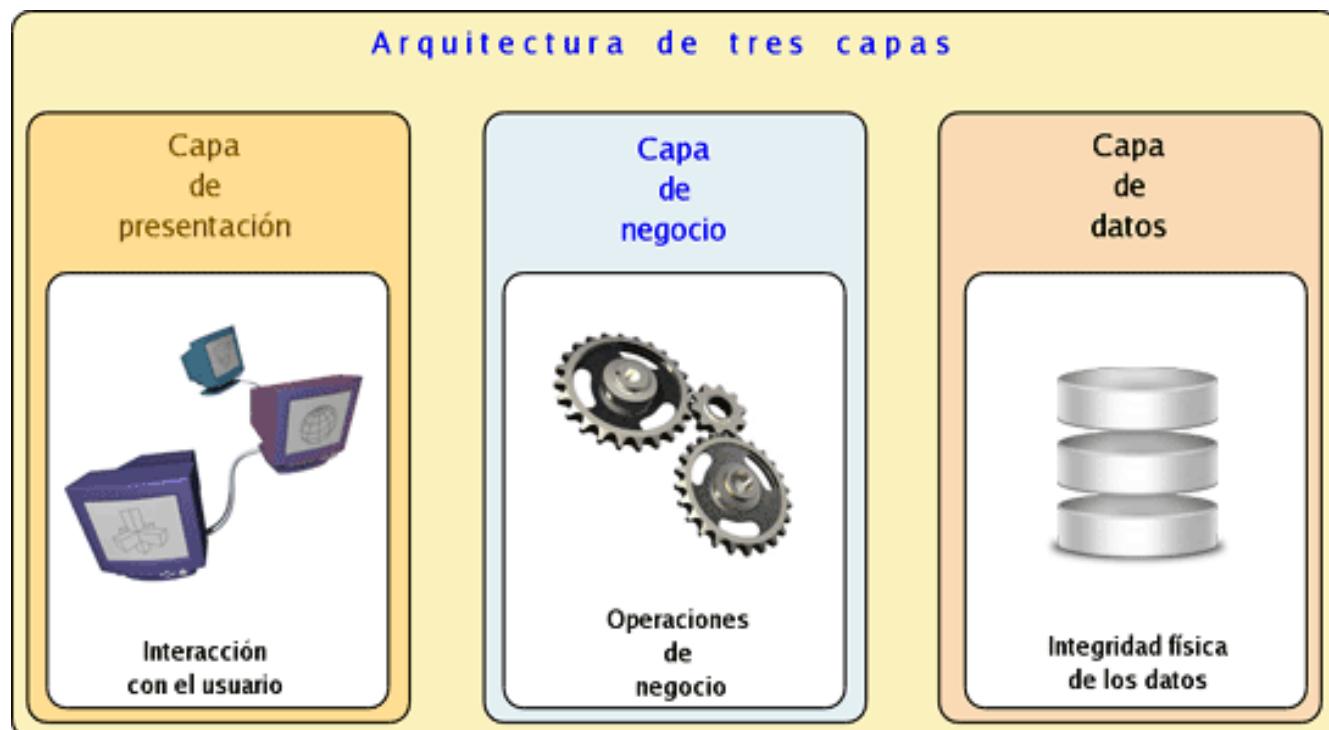
BÚSQUEDA UNIVERSAL DE CODIGOS INTERNACIONALES DE DIAGNOSTICO Y PRÁCTICA

Búsqueda de Códigos de Diagnóstico en el CIE-10

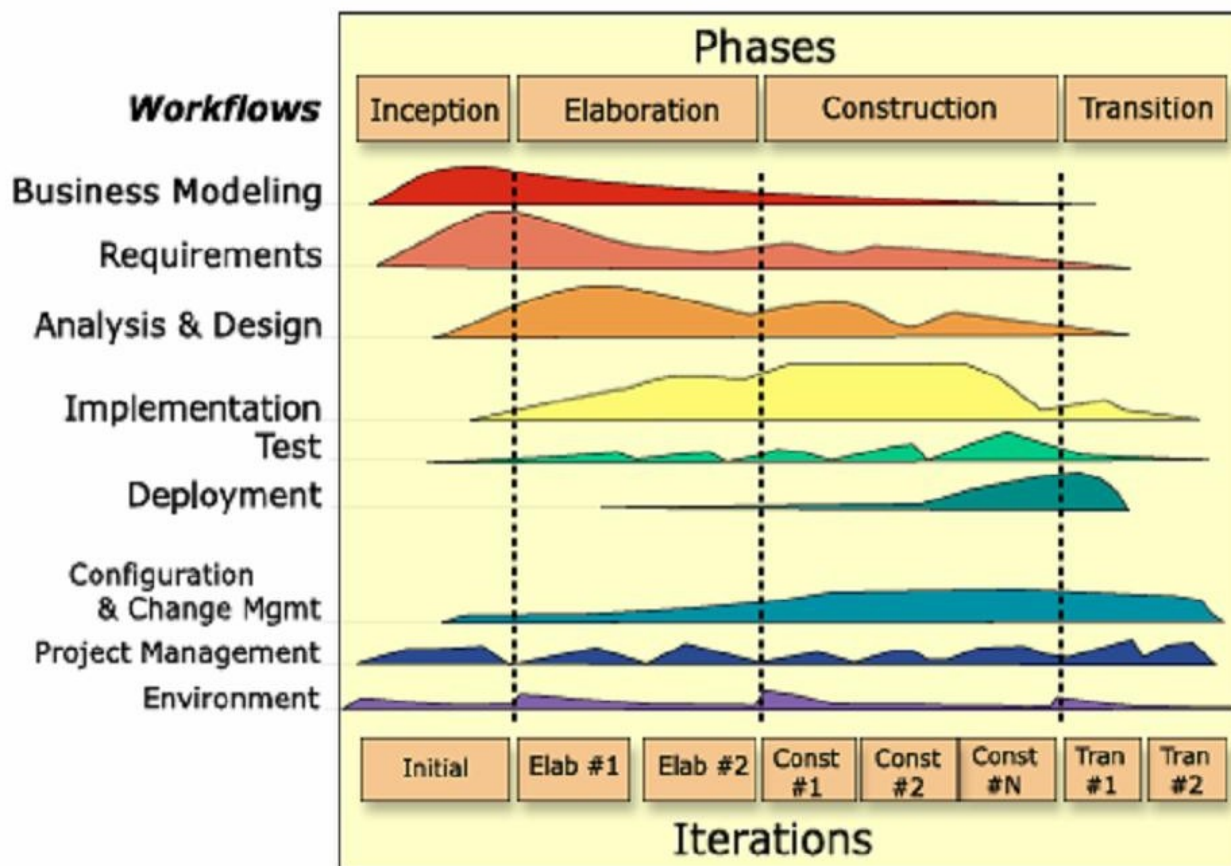
Ingrese el Término a buscar y oprima "ENTER"

CODIGO	DIAGNOSTICO
D63.0 *	anemia en enfermedad neoplasica (c00d48+)
G13.2 *	atrofia sistematica que afecta primariamente el sistema ner
G53.1 *	paralisis multiples de los nervios craneales en enfermeda
G53.3 *	paralisis multiple de los nervios craneales, en enfermeda
G63.3 *	polineuropatia en otras enfermedades endocrinas y metal
G63.6 *	polineuropatia en otros trastornos osteomusculares (m00n
G73.2 *	otros sindromes miastenicos en enfermedad neoplasica (
G94.0 *	hidrocefalo en enfermedades infecciosas y parasitarias c
G94.1 *	hidrocefalo en enfermedad neoplasica (c00d48+)
H19.1 *	queratitis y queratoconjuntivitis por herpes simple (b00.5 I
L54.0 *	eritema marginado en la fiebre reumatica aguda(i00+)
M36.1 *	artropatia en enfermedad neoplasica (c00d48+)

Anexo 6. Arquitectura en Tres Capas



Anexo 7. Fases y Flujos de Trabajos de RUP



GLOSARIO DE TÉRMINOS

Aplicación (Sistema): Sistema que ofrece a un usuario final un conjunto coherente de casos de uso.

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

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

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

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

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

Capa de Datos: Es donde residen los datos. Está formada por uno o más Sistemas Gestores de Bases de Datos (SGBD) que realiza todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

Capa de Negocio: Es donde residen los programas que son ejecutados mediante 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. 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 al gestor de base de datos almacenar o recuperar datos de él.

Glosario de Términos

Capa de Presentación: Es la que ve el usuario, presenta el sistema al usuario, le comunica la información y captura la información del usuario dando un mínimo de proceso (realiza un filtrado previo para comprobar que no hay errores de formato). Esta capa se comunica únicamente con la capa de negocio.

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

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

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.

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

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

Epidemiología: Es la parte de la medicina que se dedica al estudio de la distribución, frecuencia, determinantes, relaciones, predicciones y control de factores relacionados con la salud y enfermedad en poblaciones humanas determinadas, así como la aplicación de este estudio a los problemas de salud.

Incidencia: La incidencia de una enfermedad se define como el número de casos nuevos de esa enfermedad que afecta a una población durante un periodo determinado.

Internet: Es un 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.

Glosario de Términos

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

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

Microelectrónica: Es la aplicación de la ciencia electrónica a componentes y circuitos de dimensiones muy pequeñas, microscópicas y hasta de nivel molecular para producir dispositivos y equipos electrónicos de dimensiones reducidas pero altamente funcionales.

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.

Prevalencia: La prevalencia de una enfermedad o infección es la proporción de personas de una población que tiene esa enfermedad o infección en un momento dado.

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

Red de Computadoras o Red Informática: Es un conjunto de computadoras y/o dispositivos conectados por enlaces de un medio físico (medios guiados) ó inalámbricos (medios no guiados) y que comparten información (archivos), recursos (CD-ROM, impresoras, etc.) y servicios (e-mail, chat, juegos), etc.

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: Siglas de 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.

Telecomunicaciones: Es una técnica consistente en transmitir un mensaje desde un punto a otro, normalmente con el atributo típico adicional de ser bidireccional.

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 del Ministerio de Salud Pública (MINSAP).

This document was created with Win2PDF available at <http://www.daneprairie.com>.
The unregistered version of Win2PDF is for evaluation or non-commercial use only.