

TD-0099/05-01

005.4
Pei
2
TD-0099-05-01

INSTITUTO SUPERIOR POLITÉCNICO "JOSÉ ANTONIO ECHEVERRÍA"
FACULTAD DE INGENIERÍA INDUSTRIAL
INGENIERÍA INFORMÁTICA



*Registro de la Clasificación Internacional de Enfermedades
y Problemas Relacionados con la Salud para el Registro
Informatizado de Salud*

Trabajo para optar por el título de Ingeniería en Informática

Autor

Dayner Rogelio Peinado Patterson

Tutor

Ing. Lourdes Escalona Peral

**Ciudad de La Habana
Julio de 2005**

RESUMEN

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

A propuesta de la dirección de nuestro país se lleva a cabo a partir del año 2000 el Proceso General de Informatización del Sistema Nacional de Salud de Cuba, tomando la Atención Primaria como eje fundamental de dicho proceso. La Empresa SOFTEL en conjunto con la UCI tienen la misión de ejecutar en el marco del Proyecto APS el análisis, diseño y desarrollo de un producto de software, siguiendo las buenas prácticas y estrategias trazadas por el Ministerio de Salud Pública, facilitando la gestión de la información en los niveles de atención primaria y el flujo de información hacia los niveles de toma de decisiones.

El Registro de la Clasificación Internacional de Enfermedades y Problemas relacionados con la Salud para el Registro Informatizado de Salud formará parte de la solución que se propone para lograr la informatización del sector de la salud, el cual se presenta en este trabajo, que tiene como objetivo diseñar una aplicación Web que permita adoptar un lenguaje unificado y estandarizado, propuesto por la Organización Mundial de la Salud en la Décima Revisión de la Clasificación Internacional de Enfermedades (CIE10) y que sea utilizado por los otros registros del Sistema Integral de Salud.

Para esto investigamos sobre el proceso de desarrollo unificado, RUP y usamos el lenguaje de modelado UML.

ÍNDICE

INTRODUCCIÓN.....	1
CAPÍTULO 1 FUNDAMENTACIÓN DEL TEMA	5
1.1 INTRODUCCIÓN	5
1.2 SISTEMA NACIONAL DE SALUD	5
1.2.1 Informatización del Sistema Nacional de Salud	7
1.2.2 Informatización de la Atención Primaria de Salud	9
1.2.3 Registro Informatizado de Salud (RIS)	10
1.2.4 Solución integral propuesta para la Informatización del SNS	11
1.3 OBJETO DE ESTUDIO	13
1.3.1 Descripción General	13
1.3.2 Descripción actual de los procesos de negocio.	16
1.3.3 Situación Problemática.	18
1.4 SISTEMAS AUTOMATIZADOS EXISTENTES VINCULADOS AL CAMPO DE ACCIÓN	18
1.5 ANÁLISIS COMPARATIVO DE OTRAS SOLUCIONES EXISTENTES CON LA PROPUESTA	18
1.6 OBJETIVOS	19
1.6.1 Objetivo general.....	19
1.6.2 Objetivos específicos	19
1.7 CONCLUSIONES	20
CAPÍTULO 2 TENDENCIAS Y TECNOLOGÍAS ACTUALES A CONSIDERAR	21
2.1 INTRODUCCIÓN	21
2.2 ¿QUÉ ES INTERNET?	21
2.2.1 ¿Cómo funciona Internet?	23
2.2.2 La información a través de Internet. El World Wide Web.....	24
2.2.3 Aplicaciones Web. Aplicaciones Web vs Sitios Web	24
2.3 TECNOLOGÍA XML/WEB SERVICES (SOA).....	25
2.4 ENTORNOS DISTRIBUIDOS. MODELO CLIENTE SERVIDOR.....	26
2.4.1 Modelo Cliente Servidor de Dos Capas (Two Tier).....	27
2.4.2 Modelo Cliente Servidor de tres Capas (Three Tier)	27
2.4.3 Consideraciones sobre el hardware de la red y el software en los entornos cliente/servidor.....	28
2.4.4 Servidor web Apache.....	28
2.4.5 Arquitectura Basada en Componentes (CBA)	29
2.5 LENGUAJES DE PROGRAMACIÓN WEB	30
PERL.....	30
ASP	31
PHP	31
JSP	33
JAVA SCRIPT	33
XSLT	34
¿POR QUÉ PHP Y XSLT?	34
2.6 SISTEMAS DE GESTIÓN DE BASES DE DATOS (SGBD)	34
SQL SERVER.....	35
MYSQL	36

POSTGRESQL.....	36
2.7 DESARROLLO BASADO EN RUP BAJO LA HERRAMIENTA RATIONAL ROSE	37
2.7.1 UML (Unified Modeling Language)	38
2.7.2 Rational Rose	39
2.8 PLATAFORMA DE SERVICIO (PLASER).....	40
2.9 HERRAMIENTAS A UTILIZAR	41
2.10 REGISTRO INFORMATIZADO DE SALUD (RIS)	41
2.11 CONCLUSIONES	42
CAPÍTULO 3 DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA	43
3.1 INTRODUCCIÓN	43
3.2 MODELO DE DOMINIO	43
3.2.1 ¿Por que Modelo de dominio?.....	44
3.2.2 Glosario de términos.....	44
3.3 DIAGRAMA DEL MODELO DEL DOMINIO	45
3.4 REQUERIMIENTOS FUNCIONALES	45
3.5 REQUERIMIENTOS NO FUNCIONALES	47
3.5.1 Requisitos de Apariencia o Interfaz Externa	47
3.5.2 Requisitos de usabilidad	47
3.5.3 Requisitos de Portabilidad	47
3.5.4 Requisitos de Seguridad.....	47
3.5.5 Requisitos de Confiabilidad	48
3.5.6 Requisitos de Interfaz Interna	49
3.5.7 Requisitos de Software	49
3.5.8 Requisitos de Hardware.....	50
3.5.9 Requisitos de Diseño y la Implementación	50
3.6 DESCRIPCIÓN DEL SISTEMA PROPUESTO	50
3.6.1 Concepción general del sistema.....	50
3.6.2 Actores del sistema.....	51
3.6.3 Modelo de casos de uso del sistema	52
3.6.3.1 Gestionar Capítulo.....	52
3.6.3.2 Gestionar Grupos.....	53
3.6.3.3 Gestionar Categorías.....	54
3.6.3.4 Gestionar Subcategorías	55
3.6.3.5 Gestionar Enfermedades	56
3.6.3.6 Gestionar Subclasificaciones código opcional	57
3.6.3.7 Gestionar Definiciones.....	58
3.6.3.8 Buscar Información.....	59
3.6.3.9 Mostrar Información.....	60
3.6.4 Diagrama de Casos de Uso del Sistema	61
3.7 CONCLUSIÓN	61
CAPÍTULO 4 CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA.....	62
4.1 INTRODUCCIÓN	62
4.2 DIAGRAMA DE CLASES	62
4.2.1 Gestionar Capítulos	64
4.2.2 Gestionar Grupos	65

4.2.3 Gestionar Categorías.....	66
4.2.4 Gestionar Subcategorías.....	67
4.2.5 Gestionar Enfermedades.....	68
4.2.6 Gestionar Subclasificaciones.....	69
4.2.7 Gestionar Definiciones.....	70
4.2.8 Mostrar Información.....	71
4.3 DISEÑO DE LA BASE DE DATOS.....	72
4.3.1 Diagrama de clases persistente.....	72
4.3.2 Modelo de datos.....	73
4.4 PRINCIPIO DE DISEÑO.....	74
4.4.1 Concepción general de la ayuda.....	74
4.4.2 Tratamiento de excepciones.....	75
4.4.3 Estándares de la interfaz de la aplicación.....	76
4.4.4 Estándares de codificación.....	79
4.5 MODELO DE DESPLIEGUE.....	83
4.5.1 Modelo de Componente.....	83
4.5.2 Modelo de Despliegue.....	84
4.6 CONCLUSIONES.....	84
CAPÍTULO 5 ESTUDIO DE FACTIBILIDAD.....	85
5.1 PLANIFICACIÓN.....	85
5.2 CÁLCULO DEL ESFUERZO, TIEMPO DE DESARROLLO, CANTIDAD DE HOMBRES Y COSTO.....	88
5.3 BENEFICIOS TANGIBLES E INTANGIBLES.....	91
5.4 ANÁLISIS DE COSTOS Y BENEFICIOS.....	91
CONCLUSIONES.....	92
RECOMENDACIONES.....	93
REFERENCIA BIBLIOGRAFICA.....	94
BIBLIOGRAFÍA.....	97
GLOSARIO DE TÉRMINOS.....	100

INTRODUCCIÓN

Como parte de la Batalla de Ideas que libra actualmente nuestro pueblo, la Informatización de la Atención Primaria de Salud (APS), usando la aplicación gradual e integral de las Nuevas Tecnologías de la Información y las Comunicaciones permitirá mejorar la gerencia de los procesos relacionados con las funciones del Sistema Nacional de Salud a ese nivel de atención.

Desde hace años se han estado produciendo y poniendo en práctica soluciones informáticas para la automatización de algunos procesos administrativos, docentes, asistenciales e investigativos. El uso de la Informática en la Medicina es una de las aplicaciones más comunes e importantes desde hace varias décadas, la misma ha permitido al sector de la salud, no sólo contar con métodos novedosos, sencillos y eficaces de gestión administrativa en consultas, hospitales y centros de investigación biomédica; sino también disponer de complejos software que reducen la posibilidad de error en el diagnóstico de las enfermedades y que aceleran su formulación. A su vez, ofrece una gran ayuda en el campo de la investigación y disciplinas como Radiología y Cardiología.

En la actualidad se cuenta con una propuesta de fortalecimiento de la Informática para los policlínicos cuyos propósitos son fortalecer la conexión entre las instituciones territoriales, lo que favorecería la comunicación y la adopción de decisiones a nivel local y niveles superiores, y fortalecer el Subsistema del Médico y Enfermera de la Familia, entre otros. Se debe tener en cuenta que la tendencia en el mundo de la información apunta a que ésta y el conocimiento determinan la base de la mayoría de los procesos.

Los médicos y enfermeras de la familia tienen como misión la prevención, atención y educación para la salud de su comunidad, superación profesional e investigación, pero además tienen el deber de brindar una información fiable y actualizada al llenar correctamente los modelos de trabajo, entre los que se hallan: Hoja de Actividades Diarias, Historia de Salud Familiar, Notificación de Enfermedades de Declaración Obligatoria, Certificados Médicos de Defunción, entre otros, de modo que satisfagan las necesidades de los trabajadores de la salud que están en función del procesamiento informativo para la producción de estadísticas sanitarias que comienzan en el Policlínico

hasta la Dirección Nacional de Estadísticas del MINSAP, los cuales desempeñan una labor muy importante, gracias a la cual se obtienen las bases de conocimiento para estudios epidemiológicos y de todo tipo, que permiten apoyar las etapas de la administración, así como las actividades docentes e investigativas propias y extra sectoriales; establecer líneas de conducta para la reducción de la morbilidad y mortalidad, realización de acciones para diagnósticos precoces, trabajar en grupos de riesgo en prevención primaria y hacer comparaciones con períodos anteriores y con situaciones de otras regiones dentro o fuera del país.

El actual proyecto tiene como misión implementar un Programa General de Informatización del Sistema Nacional de Salud, que apoye las Estrategias y Políticas trazadas por la dirección del país y el Ministerio de Salud Pública (MINSAP), teniendo como centro del mismo al Policlínico.

SOFTEL, empresa productora de software en Cuba, fundamentada sobre la base de una eficiente gestión de requerimientos donde participan, desde un inicio, médicos y trabajadores de la salud, vinculados directamente a la APS en calidad de expertos funcionales en estrecho vínculo con los especialistas de Informática, tiene la misión del desarrollo de soluciones informáticas para la salud, en conjunto con la Universidad de las Ciencias Informáticas (UCI).

Se propone que el Sistema Integral de Salud esté formado por 3 grandes grupos: el Sistema Informatizado de Gestión Hospitalaria, ejemplo el Registro de Autopsia; el Sistema Informatizado de Atención Primaria, ejemplo el Registro de Población y el Sistema de Salud, integrado a su vez por el Registro Médico Informatizado de Salud y el Registro No Médico Informatizado de Salud. En este último grupo se ubicará el Registro de la Clasificación Internacional de Enfermedades y Problemas relacionados con la Salud (RCIE).

La clasificación internacional de enfermedades (CIE) y problemas relacionados con la salud puede definirse como un sistema de categorías a las cuales se asignan entidades morbosas de acuerdo con criterios establecidos. El propósito de la CIE es permitir el registro sistemático, el análisis, la interpretación y la comparación de los datos de

mortalidad y morbilidad recolectados en diferentes países o áreas, y en diferentes épocas. La CIE se utiliza para convertir los términos diagnósticos y de otros problemas de salud, de palabras a códigos alfanuméricos que permiten su fácil almacenamiento y posterior recuperación para el análisis de la información. 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 apoya al análisis de la situación general de salud de los grupos de población, el seguimiento de la incidencia y prevalencia de enfermedades y otros problemas de salud, entre otras variables, tales como las características y circunstancias de los individuos afectados.

La historia del CIE se remonta al pasado hacia, la época de Sir George H. Knibbs, el eminente estadístico australiano, el cual posee el mérito de ser el primer ensayo de clasificación sistemática de enfermedades a François Bossier de Lacroix (1706–1777), mejor conocido por el nombre de Sauvages (10). El estudio estadístico de las enfermedades se había iniciado un siglo en el siglo XVIII con el tratado de John Graunt sobre las tablas de mortalidad de Londres. En un intento por calcular la proporción de niños nacidos vivos que fallecieron antes de llegar a la edad de seis años entonces no se registraba la edad de los fallecidos Graunt tomó todas las muertes que habían sido clasificadas bajo las rúbricas muguet, convulsiones, raquitismo, trastornos de la dentición y gusanos, prematuros, recién nacidos sin bautizar (chryosomes), lactantes, hígado grande y niños asfixiados (sofocados), y les sumó la mitad de las defunciones atribuidas a viruela, varicela, sarampión y parasitados por gusanos sin convulsiones. A pesar de que su clasificación era elemental e imperfecta, su cálculo de que un 36% de la mortalidad ocurría antes de los seis años parece haber sido correcto a la luz de conocimientos ulteriores. Si bien tres siglos ha contribuido algo a la exactitud científica de la clasificación de enfermedades, muchos todavía dudan de la utilidad de las tentativas y los ensayos hechos para recopilar estadísticas de enfermedades o aun de causas de defunción, debido a las dificultades inherentes a la clasificación.

En la actualidad la OMS dicta cada cierto tiempo una nueva versión de la Clasificación Internacional de Enfermedades y problemas relacionados con la salud (CIE) con el fin de mantener actualizada la información, pues estos códigos pueden variar con el transcurso

del tiempo debido a que surgen nuevos padecimientos y otros simplemente se diversifican o mutan a otros males.

Teniendo en cuenta las directivas trazadas por nuestra máxima dirección del país con el fin de automatizar todos los procesos de atención a la salud para mejorar los servicios que en la misma se prestan se plantea el siguiente problema.

Problema:

Como facilitar, mediante la informatización, la gestión de la información estandarizada y homogénea de la Clasificación Internacional de Enfermedades y Problemas relacionados con la Salud en todos los niveles de atención de salud.

Objeto de estudio:

Proceso de gestión de la información estandarizada de la Clasificación Internacional de Enfermedades y Problemas relacionados con la Salud en los niveles de atención para el Sistema Nacional de Salud, teniendo en cuenta las herramientas de diseño y desarrollo estudiadas.

Campo de acción:

Proceso del registro informatizado que controla la información estandarizada de la Clasificación Internacional de Enfermedades y Problemas relacionados con la Salud en los niveles de atención de salud.

Objetivo:

Realizar el diseño de una aplicación Web que facilite para todos los niveles de atención de salud la información estandarizada y homogénea de la Clasificación Internacional de Enfermedades y Problemas relacionados con la Salud.

CAPÍTULO 1 FUNDAMENTACIÓN DEL TEMA

1.1 Introducción

El objetivo fundamental de este capítulo es abordar distintos aspectos que se utilizan como soporte teórico del sistema diseñado. Se exponen a través de una descripción de los conceptos asociados al problema, de la estructura organizativa del Sistema Nacional de Salud (SNS), así como de los conceptos principales para comprender la gestión de la información de los pacientes y la familia que son atendidos por el Equipo Básico de Salud (EBS). Se define además el objeto de estudio, la situación problémica y los objetivos generales y específicos de la presente investigación.

1.2 Sistema Nacional de Salud

La garantía de atención médica gratuita a toda la población cubana se convirtió desde los primeros momentos del triunfo de la Revolución en uno de los paradigmas sociales fundamentales, en correspondencia con la esencia humanista y de justicia social que caracteriza a nuestro proceso revolucionario. [1]

Desde el propio triunfo revolucionario se adoptaron medidas para transformar la salud pública en Cuba, una de las principales y más novedosa fue la creación del Sistema Nacional de Salud (SNS), designándose al Ministerio de Salud Pública como su organismo rector. [2]

Esta estructura organizativa comenzó a realizar importantes reformas a partir de los años 60, como parte fundamental de las transformaciones del período revolucionario y en respuesta al respeto más absoluto de uno de los derechos humanos fundamentales de todo ciudadano. Surge el servicio de hospitales rurales llevando la atención médica a zonas apartadas de la geografía nacional, se dan los primeros pasos para el fortalecimiento de la atención primaria; surgen los policlínicos integrales como una unidad asistencial creada para brindar servicios y resolver los principales problemas existentes en los primeros años de la revolución.

En la década del 70, por los cambios en el cuadro de morbilidad - mortalidad, los servicios prestados en los policlínicos integrales cobran nuevas funciones, cambiando la estructura de los mismos, pasando a una atención médica general, surgiendo así el policlínico comunitario donde prestaban atención los médicos generales. [3]

En la década del 80 surge el Programa del Médico y la Enfermera de la Familia, sentando precedentes en la salud pública internacional por su carácter novedoso y futurista, especialmente con la implantación y desarrollo del modelo de atención de Medicina Familiar a partir de 1984. [4]

En 1996, el SNS adoptó desde el punto de vista organizativo, estrategias fundamentales y priorizó cuatro programas básicos para continuar perfeccionándose: el Programa de Atención Materno Infantil (PAMI), de Control de Enfermedades Transmisibles y Crónicas no Transmisibles, y el de Atención al Adulto Mayor, todos los que han sido monitorizados, controlados y evaluados de acuerdo a la metodología establecida.

El Programa del Médico y la Enfermera de la Familia, se ratifica como el eje del actual desarrollo estratégico, orientándose el resto de las estrategias en función del mismo. Este modelo de atención es la mayor fortaleza y potencialidad que tiene el SNS. Por su existencia, filosofía, bases teóricas y lo que ha podido proporcionarle al sistema de salud se ha logrado mantener los indicadores de salud y satisfacer las necesidades de la población, constituyendo un pilar básico de la Salud Pública Cubana. [5]

Con más de 20 años de experiencia en este programa se comienzan a experimentar cambios para la atención primaria, de esta forma, servicios que antes eran exclusivos de hospitales son abiertos en instituciones de la atención primaria; surgiendo así hace aproximadamente 2 años el novedoso modelo de policlínico con nuevas funciones, acercando los servicios a la población, para hacer realidad las palabras de nuestro Comandante en Jefe: “... **una profunda revolución en los servicios de salud tendrá lugar en nuestra Patria...**” [6]

1.2.1 Informatización del Sistema Nacional de Salud

La informatización del SNS tiene como objetivo acercar eficientemente y con calidad la prestación de los servicios de salud a la población, por lo que se pretende implementar un Programa General de Informatización del SNS, que apoye las Estrategias y Políticas trazadas por la dirección del país y del MINSAP; de manera que se logre la incorporación progresiva y sistemática de las Nuevas Tecnologías de la Información y las Comunicaciones (NTIC) en función de la adquisición y gestión del conocimiento y los servicios de salud.

Se quiere que las instituciones del país alcancen un elevado nivel de informatización de las actividades que brindan, partiendo del Sistema de Atención Primaria y tomando como eje al policlínico, de manera que se logre 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 usuarios del Sistema Nacional de Salud.

Como solución integral significa la articulación de un nuevo paradigma en la prestación de servicios de salud, regido por el principio básico de lograr acercar cada vez más los servicios de salud a la población. Entre los principales impactos esperados con la Informatización del SNS podemos mencionar:

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.

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.

En las líneas generales del Desarrollo Informático en la Salud se encuentran: la Atención Primaria, Secundaria y Terciaria, el Sistema Integrado de Urgencia Médica, Vigilancia de Salud, Telemedicina, Medicamentos y Fármacos, Epidemiología, Biblioteca y Universidad Virtual, Docencia Médica, entre otros.

SOFTEL, Empresa del Ministerio de la Informática y las Comunicaciones (MIC), tiene la misión de generar las 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, 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.

A través del proceso de desarrollo unificado (RUP), y haciendo uso del Lenguaje Unificado de Modelado (UML) se describen los procesos que se proponen para automatizar.

SOFTEL ejecuta estos objetivos en colaboración con la Universidad de las Ciencias Informáticas (UCI), para lograr la vinculación a la producción desde los primeros años de estudio de los estudiantes y los profesores líderes de proyectos y la formación en un segundo perfil en temas relacionados con la salud.

La experiencia de este trabajo en SOFTEL debe constituir el inicio de buenas prácticas en la producción de software con alta calidad y un ejemplo de normativa para los proyectos que deben irse abriendo de ahora en adelante en la Informatización del Sector de la Salud. [7].

El Sistema de Salud Cubano, posee en el nivel de Atención Primaria una plataforma ideal para articular los avances de las nuevas tecnologías de la información en función de hacer más eficiente todo el aparato estratégico y administrativo que rodea al propio sistema.

1.2.2 Informatización de la Atención Primaria de Salud

El Proceso de Informatización de la Atención Primaria de Salud (APS), es un proyecto priorizado para el SNS, cuyo objetivo fundamental consiste en la creación del Sistema Informatizado de Atención Primaria que permita la gestión médica, interacción con los consultorios del Médico de la Familia, obtención de estadísticas y apoyo en la logística de los nuevos servicios.

En el marco del Programa de Informatización de la Sociedad Cubana, el Proyecto APS en su concepción general se propone abordar el análisis, diseño y desarrollo de un producto de software, siguiendo las buenas prácticas internacionales y las normativas del MINSAP, logrando que facilite la gestión de la información en la Atención Primaria, acorde a los cambios y necesidades de este sector, permitiendo el flujo de información hacia los diferentes niveles de toma de decisiones.

En esta nueva etapa de fortalecimiento del Sistema Nacional de Salud, la Atención Primaria de Salud (APS) es el eje fundamental de estas transformaciones, teniendo como objetivo fundamental convertir a los Policlínicos en centros de atención primaria de salud de la más alta calidad, cada vez más accesibles a la población, consolidando el Sistema

Municipal de Salud, para dar cumplimiento al principio de la descentralización de las soluciones según los problemas de salud de la comunidad.

La automatización de la gestión de la APS debe comenzar por utilizar las tecnologías que permitan modelar la gestión de la información en este nivel para almacenar, procesar, recuperar y comunicar información clínica y administrativa, relativa a todas las actividades de los policlínicos y unidades de la atención primaria. Debe tener la capacidad de comunicación y de integración de toda la información, independientemente de donde se haya generado y que sirva para el aprendizaje basado en experiencias compartidas entre los profesionales en el país y fuera de nuestras fronteras, así como para lograr la integración con los procesos de los otros niveles de atención.

1.2.3 Registro Informatizado de Salud (RIS)

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, lo cual trae consigo la duplicación de información y la consiguiente falta de integridad de la misma.

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.

El Proyecto APS vinculado a la informatización de la salud en Cuba encamina su tarea a analizar, diseñar y desarrollar un producto de software, único en su tipo, que hereda las características del RIS, pero que 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, para dar respuesta a los procedimientos establecidos por el SNS para este nivel de atención.

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

El RIS se basa en una arquitectura orientada a servicios, desarrollado con tecnología XML Web Services e implementado con PHP y MySQL. Desde el año 2003 forman parte del RIS los siguientes componentes: Registro de Unidades de Salud, Registro de Profesionales de la Salud, Registro de Ubicación, Registro del Ciudadano y Registro de Equipos de Salud.

1.2.4 Solución integral propuesta para la Informatización del SNS

Para lograr la Informatización en este sector se pretende que todos los módulos estén incluidos en un conjunto de aplicaciones que formarán parte del **Sistema Integral de Salud (SISalud)**, compuesto a su vez por el **Registro Informatizado de Salud (RIS)**, el **Sistema Informatizado de Atención Primaria (SIAP)** y el **Sistema Informatizado de Gestión Hospitalaria (SIGH)**.

Registro Informatizado de Salud (RIS): Está formado por los registros que son administrados o gestionados a nivel nacional o central y que integran el **Registro No Médico Informatizado de Salud (RNMIS)** y por los registros que pueden ser accedidos desde cualquier nivel de atención o institución de salud para lograr la continuidad en el seguimiento del paciente, agrupándose éstos en el **Registro Médico Informatizado de Salud (RMIS)**.

Registro No Médico Informatizado de Salud (RNMIS): En esta nueva etapa de análisis, diseño y desarrollo se incorporarán al RNMIS: el Registro de Áreas de Salud, Registro de Medios de Diagnóstico, Registro de la Clasificación Internacional de Enfermedades y Problemas de Salud (CIE) y los codificadores propios de APS que se gestionan también a nivel central y definen diferentes aspectos que son utilizados localmente: Registro de Conductas e Indicadores y Registro de Dispensarización. De igual forma se ubicarán en el RNMIS todos los registros que en la actualidad pertenecen al RIS, mencionados en el epígrafe anterior.

Registro Médico Informatizado de Salud (RMIS): Estará formado por todos los módulos o componentes que no son del dominio de Atención Primaria propiamente, pero procesan y generan información que se obtiene de este nivel comunitario y además lo retroalimenta.

En esta primera etapa se desarrollan: Registro de Enfermedades de Declaración Obligatoria (EDO), Registro de Fallecidos y Registro de Partos y Nacimientos.

Sistema Informatizado de Atención Primaria (SIAP): Se incluirán en la etapa actual los módulos propios de este nivel de atención: Registro de Actividades Diarias EBS y Registro de Población. Estos módulos constituirán una nueva herramienta para la transformación de los servicios que se brinda en este nivel, ya que integrarán diversos subsistemas como las actividades diarias del EBS, la dispensarización y la planificación de las acciones de salud, tanto individual como familiar.

En una segunda etapa continuarán incorporándose al SIAP los próximos módulos que se definan, según las prioridades del usuario.

Sistema Informatizado de Gestión Hospitalaria (SIGH): Se agruparán aquí los módulos que pertenecen al nivel de atención secundario u hospitalario y que serán definidos para próximos desarrollos. En esta etapa comenzará a formar parte del mismo el Registro de Autopsias, diseñado en la etapa actual bajo el Proyecto APS por la integración que tiene con el resto de los módulos que se desarrollan.

A continuación se muestra un diagrama que nos permitirá comprender con claridad la definición del Proyecto del Sistema Integral de Salud, en correspondencia con la explicación realizada anteriormente.

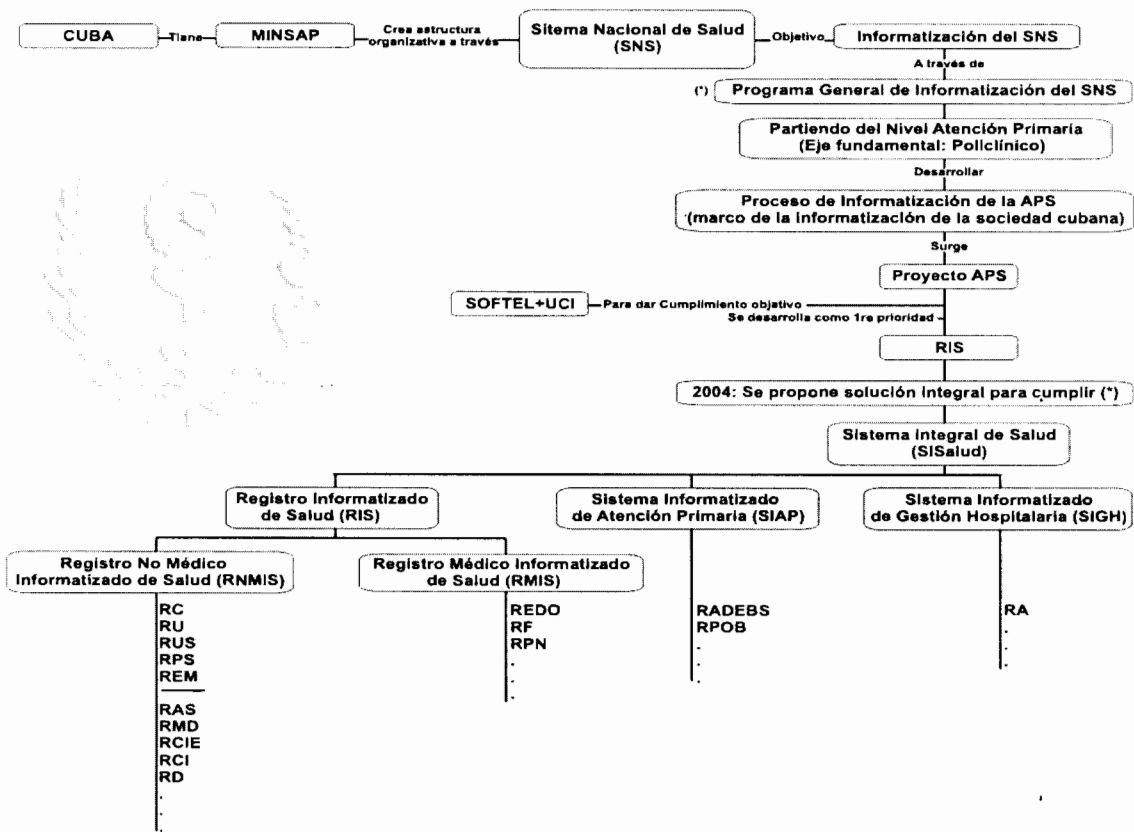


Fig.1 Proyecto del Sistema Integral de Salud (SISalud).

1.3 Objeto de estudio

1.3.1 Descripción General

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

El SNS se estructura en tres niveles que se corresponden con la estructura político-administrativa del país. El nivel nacional está representado por el Ministerio de Salud Pública que es el órgano rector con funciones metodológicas, normativas y de coordinación y control, al cual se le subordinan directamente los centros universitarios, institutos de investigaciones, centros hospitalarios de asistencia médica altamente especializados, centros de distribución y comercializadoras de suministros y tecnologías

médicas, así como otros centros y entidades nacionales destinados a actividades técnicas y de apoyo. [8]

Los otros dos niveles están representados por las direcciones provinciales y municipales de salud que agrupan a las instituciones de salud en su respectivo nivel y que, al igual que en el nivel central, se subordinan desde el punto de vista administrativo a las estructuras de Gobierno en los distintos niveles organizativos, representando sus intereses ante ellos y dando respuesta a las demandas y necesidades de la población.

Las áreas de salud y las unidades que prestan servicios en la atención primaria se subordinan a los Consejos Populares, quienes se encargan de resolver los problemas más específicos de la población de su radio de acción. Además la labor de los EBS se subordina a los Delegados de Circunscripción pertenecientes a los Consejos Populares. [9]

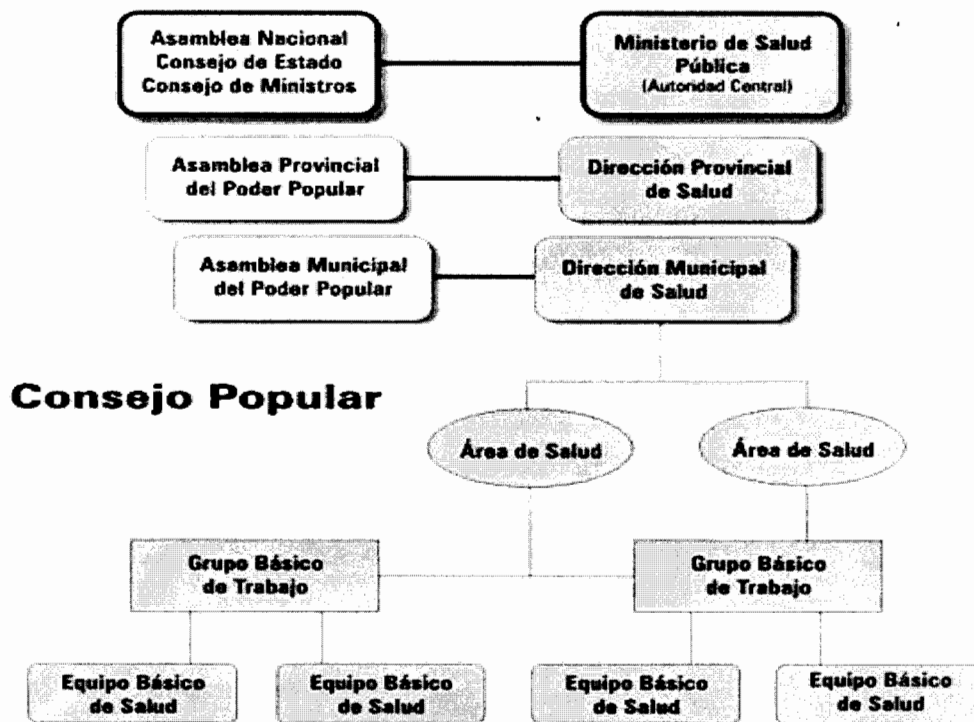


Fig.2 Esquema de la Estructura Organizativa del MINSAP

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

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

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

La función principal de la atención primaria es la promoción-prevención de salud en las diferentes comunidades, además se realizan procedimientos diagnósticos y terapéuticos que no requieren técnicas complejas, que aplicadas con calidad pueden resolver la mayor parte de los padecimientos que afectan a las poblaciones. Se diagnostican enfermedades graves que pueden ser derivadas a niveles de atención superiores, realizan seguimiento a personas con padecimientos crónicos y pueden otorgar bienestar a pacientes con patologías incurables. En general tiene carácter ambulatorio y comprende tanto a personas aparentemente sanas como a enfermas y/o discapacitadas. [10]

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

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

Puede tener carácter ambulatorio (policlínicos especializados, servicios externos hospitalarios) o de hospitalización. En el mismo se brindan procedimientos diagnósticos y

terapéuticos de elevada complejidad, que dan respuesta a los problemas moderados y graves de salud.

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

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

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

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

1.3.2 Descripción actual de los procesos de negocio.

La CIE actualmente se ha convertido en una clasificación diagnóstica estándar internacional para todos los propósitos epidemiológicos generales y muchos otros de

administración de salud. Esto apoya 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.

Toda la información registrada mediante este estándar de clasificación aparece registrada en una amplia variedad de registros, a partir de los cuales se derivan muchas estadísticas y otras informaciones sobre la situación de salud.

El CIE esta estructurado de la siguiente forma:

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 orientación 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.

En el primer volumen se encuentra los grupos de enfermedades las cuales se agrupan en capítulos, grupos, categorías y subcategorías en ese orden donde la siguiente es un subconjunto de la anterior. La OMS es la entidad encargada de organizar dicha estructura, así como la informar sobre las posibles actualizaciones que se presenten en el transcurso del tiempo.

Partiendo de todo esto, podemos decir que al personal médico le resulta complejo la utilización de cada uno de los volúmenes para diagnosticar las enfermedades o

problemas relacionados con la salud producto del gran volumen de información que poseen dichos tomos, además de su compleja forma de organizarse; [✓] Pues dicha información se encuentra distribuida indistintamente en todos los tomos y para realizar un diagnóstico [✓] completo necesita consultar los 3 tomos a la vez.

1.3.3 Situación Problémica.

El problema principal radica en el proceso de informatización que se lleva a cabo en nuestro país y en especial el modulo del RCIE para apoyar los servicios del RIS en los diferentes módulos. Además el uso que actualmente tiene la Clasificación Internacional de Enfermedades y problemas relacionados con la salud (CIE) dictada por la OMS para realizar los diagnósticos médicos, los cuales se dificultan por varias razones, poca disponibilidad de los tomos referentes ^(?) los códigos de la CIE debido a la escasez de dichos tomos en nuestro país, la complejidad para diagnosticar a los pacientes a partir de dichos tomos por el gran volumen de información que contiene cada uno de ellos, el poco conocimiento del mismo por parte de personal calificado para poder hacer uso adecuado de la información que en su interior contiene cada volumen.

1.4 Sistemas automatizados existentes vinculados al campo de acción [✓]

Actualmente existe una aplicación multimedia (CIE-10) para facilitar la información de la décima revisión de la Clasificación Internacional de Enfermedades Problema relacionado con la Salud, en su interior contiene toda la información relacionada con los códigos alfanuméricos de morbilidad y mortalidad al igual que toda la información adicional referente a la codificación.

1.5 Análisis comparativo de otras soluciones existentes con la propuesta [✓]

Para realizar un análisis comparativo de la aplicación antes expuesta y de la solución que se propone se tuvieron en cuenta los siguientes parámetros: Usabilidad, gestión para la actualización de la información y plataformas que permite para su uso.

Analizando la propuesta de la Organización Mundial de la Salud (OMS) podemos decir que la misma tiene organizada toda la información y que permite realizar las búsquedas por distintos criterios devolviendo la respuesta deseada en el caso de acertada la búsqueda. Pero dicha propuesta presenta algunos inconvenientes como la poca compatibilidad con otros sistemas que puedan hacer uso de sus servicios, la negativa de no poder realizar las actualizaciones de su información a partir de los nuevos cambios dictados por la OMS.

En la actualidad la propuesta que se plantea supera en esos aspectos la anterior versión partiendo que la misma presenta gran compatibilidad con otros sistemas externos y es muy flexible a la hora de realizar las actualizaciones dictadas por la OMS, permitiendo realizar numerosas búsquedas de información por distintos parámetros manteniendo el principio de la aplicación anterior.

La solución existente funciona como un libro de consulta y no como una solución informática que permita la integración con otros módulos de salud que necesitan la CIE para gestionar sus procesos informáticos.

1.6 Objetivos

1.6.1 Objetivo general

Este trabajo tiene como objetivo realizar el diseño de una aplicación Web que facilite a todos los niveles de salud la información estandarizada y homogénea de la Clasificación Internacional de Enfermedades y Problemas Relacionados con la Salud.

1.6.2 Objetivos específicos

Para dar solución al objetivo propuesto se plantean los siguientes objetivos específicos.

- Analizar los aspectos teóricos conceptuales del Ministerio de Salud Pública.
- Realizar la gestión de la información de las Enfermedades y Problemas de Salud.

- Realizar el análisis y diseño del Registro de la Clasificación Internacional de Enfermedades y problemas relacionados con la Salud, utilizando el proceso de desarrollo RUP y el lenguaje de modelado UML.
- Modelar los procesos de interacción del módulo que conformarán el funcionamiento del mismo. |←
- Realizar un estudio de las herramientas informáticas a utilizar para la futura implementación del módulo.

1.7 Conclusiones

Descritos los principales conceptos teóricos que sustentan el presente trabajo están creadas las condiciones para escoger las herramientas informáticas que permitan construir una solución factible basada en la propuesta realizada.

CAPÍTULO 2 TENDENCIAS Y TECNOLOGÍAS ACTUALES A CONSIDERAR

2.1 Introducción

En este capítulo se realiza un análisis detallado de los principales conceptos y tecnologías que pueden ser adecuados para el desarrollo del sistema. Se describen los sistemas integrados y distribuidos, el uso de las arquitecturas existentes y la metodología a utilizar para el análisis y diseño del sistema teniendo en cuenta las facilidades que puede aportar al trabajo. Se hace un estudio de algunos de los diferentes lenguajes de programación y de los sistemas de Gestión de Bases de datos (SGBD) más usados; definiéndose los más adecuados para el sistema.

2.2 ¿Qué es Internet?

En 1957 ocurre un suceso que influiría posteriormente en el desarrollo de las redes de computadoras. El Departamento de Defensa de los EE.UU. crea la Agencia de Proyectos de Investigaciones Avanzadas, ARPA (Advanced Research Projects Agency), como respuesta al lanzamiento por la URSS del primer satélite artificial de la Tierra. Esta agencia se encargaría de desarrollar proyectos técnicos e investigativos aplicables en el campo militar.

Es en este contexto que ARPA decide realizar investigaciones con el objetivo de interconectar computadoras utilizando como medio de transmisión la red telefónica existente en el país y experimentar si computadoras ubicadas en lugares geográficos distantes podían comunicarse empleando una nueva tecnología que había surgido recientemente en el viejo continente con el nombre de Conmutación de Paquetes.

Es conveniente recalcar que la meta de ARPA no era la creación de la red internacional de computadoras existente hoy día, sino como dijimos antes, la conformación y desarrollo de una red de computadoras que pudiera sobrevivir a un ataque nuclear soviético. Estados Unidos estaba envuelto en la escalada militar y su principal preocupación era poseer un sistema de comunicaciones que sobreviviera a dicho ataque. La solución era crear una red de computadoras sin centro, en la que los mensajes no tuvieran una única ruta para

transitar de un punto a otro del territorio norteamericano.

Para ello en 1966 un grupo de investigadores de ARPA inició el primer esfuerzo serio por “enseñarle” a las computadoras a “hablar” unas con otras. Como resultado de todo ese esfuerzo aparece en diciembre de 1969 la red bautizada con el nombre de ARPANet, germen de la actual Internet.

La red ARPANET conectaba las Universidades de California en Santa Bárbara y en los Angeles, la de UTAH en Salt Lake City y la Universidad de Standford. Esta tecnología posibilitaba que los científicos de esas universidades compartieran la información y los recursos de la red independientemente de la distancia que los separara.

Coincidentemente, en el propio año 1969 Ken Thompson creaba la primera versión de UNIX, sistema operativo que luego representaría un papel principal en la comunicación entre computadoras.

Los científicos descubrieron enseguida que podían usar esa red para comunicarse. A la red primitiva se comenzaron a enlazar universidades que empezaron a usarla entre otras cosas para transmitir conferencias en línea. Pero hasta esos momentos sólo podían interconectarse computadoras con similares sistemas operativos.

En 1972 ya había 37 sitios diferentes conectados a ARPANet. Estos sitios intercambiaban mensajes entre usuarios individuales, lo que se llamó correo electrónico (e-mail), tenían la posibilidad de controlar una computadora de manera remota, proceso conocido como login remoto (r login o telnet) y permitían el envío de largos ficheros de textos o de datos entre computadoras utilizando el FTP (File Transfer Protocol).

Durante 1973 ARPANet desborda las fronteras de los EEUU al establecer conexiones internacionales con la “University College of London” de Inglaterra y el “Royal Radar Establishment” de Noruega. En este propio año se presentan las ideas básicas de Internet en el Grupo de Trabajo de Interconexión de Redes (INWG, InterNet Working Group).

Ya más avanzada la década del 70, DARPA (nuevo nombre tomado por ARPA a partir de 1972) le encarga a la Universidad de Standford la elaboración de protocolos que

permitieran la transferencia de datos entre diferentes tipos de redes de computadoras. Es el momento en el que Bob Kahn, Vinton Cerf y un grupo de sus estudiantes desarrollan los protocolos TCP/IP que hicieron posible el surgimiento de la red universal de computadoras que existe en la actualidad bajo el nombre de Internet.

En la década de 1980 esta red de redes conocida como la Internet fue creciendo y desarrollándose debido a que con el paso del tiempo cientos y miles de escuelas, universidades, centros de investigación y agencias del gobierno fueron conectando sus computadoras a la red. Algunas de estas instituciones comenzaron a permitir acceso al público, es decir, que a partir de esos momentos toda persona con una computadora y un módem podía colocarse en la red.

Como es natural, en la actualidad la Red Internet está soportada por una sofisticada y continuamente cambiante plataforma tecnológica, la cual permite a esta “Red de Redes” crecer exponencialmente.

2.2.1 ¿Cómo funciona Internet?

El uso de la tecnología de Conmutación de Paquetes propicia que todas las computadoras estén al mismo nivel en la red, lo que descentraliza su control, cumpliéndose así el objetivo de evitar la existencia de una computadora central.

Para que todas estas computadoras puedan comunicarse entre sí, debe existir un medio físico que las una (pares de cobre, enlaces satelitales, infrarrojos, microondas, fibra óptica, etc.). Además deben “ponerse de acuerdo” en cuanto al lenguaje que utilizarán para comunicarse, lo que técnicamente se conoce como Protocolo de Comunicación.

Los protocolos de comunicación son conjuntos de reglas que rigen la forma de comunicación entre máquinas. Al igual que los humanos usamos ciertas reglas para comunicarnos, el mundo de los unos y los ceros de las computadoras también las tienen. Cada una de esas formas es llamada protocolo de comunicación. Un conjunto de esos protocolos está encapsulado en el TCP/IP (Transmission Control Protocol/ Internet Protocol), los que fueron adoptados por decisión de DARPA en 1982 como el conjunto de protocolos estándar para todas las computadoras conectadas a ARPANet. Decisión

gracias a la cual se convirtieron en el lenguaje de comunicación de Internet. [11]

2.2.2 La información a través de Internet. El World Wide Web

World Wide Web (WWW), o simplemente Web, es el universo de información accesible a través de Internet, una recopilación universal del conocimiento humano. Es un sistema de información global, interactivo, dinámico, distribuido, gráfico, basado en Hipertexto, con plataforma de enlaces cruzados, que se ejecuta en Internet.

El componente más usado en Internet es definitivamente la Web. Su característica sobresaliente es el texto remarcado, un método para referencias cruzadas instantáneas. Usando la Web, se tiene acceso a millones de páginas de información en todo el mundo. La exploración se realiza por medio de un software especial denominado "Browser" o Navegador. La apariencia de un Sitio Web puede variar ligeramente dependiendo del navegador que se utilice. Las versiones más recientes de estos navegadores incorporan funcionalidades adicionales para permitir la reproducción de animaciones, realidad virtual, sonido y video en formato digital. El protocolo que se utiliza para la comunicación en la Web es el http (Hypertext Transfer Protocol) y el formato que se utiliza para la transferencia es el HTML (Hypertext Markup Language).

2.2.3 Aplicaciones Web. Aplicaciones Web vs Sitios Web

Las aplicaciones Web se desarrollan como una extensión de los Sistemas Web para agregar funcionalidad de negocio al proceso. En términos más simples, una Aplicación Web es un Sistema Web que permite a los usuarios ejecutar lógica de negocio a través de un Navegador (Browser), o lo que es lo mismo, modificar el estado del negocio.

Las Aplicaciones Web utilizan las tecnologías existentes para generar contenidos dinámicos y permitir a los usuarios del sistema modificar la lógica del negocio en el servidor. Si no existe lógica de negocios en el servidor, el sistema no puede ser considerado una aplicación Web, en ese caso se considera como un sitio Web. En esencia una aplicación Web usa un sitio Web como entrada (front-end) a una aplicación típica.

La arquitectura de un Sitio Web es simple. Contiene como componentes principales: el Servidor Web, una Red y uno o más Navegadores o clientes. El servidor Web distribuye páginas de información formateada a los clientes que las solicitan. Los requerimientos son hechos a través de una conexión de red, y para ello se usa el protocolo HTTP.

El cliente o un navegador (browser) es el responsable de mostrar la información al usuario y de hacer validaciones sencillas en la entrada de datos para que la información sea mostrada al usuario.

2.3 Tecnología XML/WebServices (SOA) ✓

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

Los servicios Web XML actúan de forma independiente y además permiten que las aplicaciones compartan información e invoquen funciones de otras aplicaciones, independientemente del sistema operativo o la plataforma en que se ejecutan y los dispositivos utilizados para obtener acceso a ellos, o sea, son rutinas en Internet actuando como catalizadores de transacciones vía Web. Los Servicios Web usan SOAP (Simple Object Access Protocol) como protocolo de transporte estándar por su simplicidad, se puede identificar un mensaje SOAP como un documento XML conformado por una envoltura "envelope" obligatoria, un encabezamiento "header" opcional y un cuerpo "body", también obligatorio [SOAP-W3C], y la forma de acceder a ellos es a través del WSDL (Web Services Description Languages). Estos servicios deben publicar una interfaz que funja como un contrato de servicio y donde se describan cada una de las funciones que provee además de las funciones que estos ofrecen, como realizar el intercambio de mensajes, especificar el contenido de una petición y el aspecto de la respuesta en una

0008-05

notación inequívoca. Además de describir el contenido de los mensajes, WSDL define dónde está disponible el servicio y qué protocolo de comunicaciones utilizar para hablar con el servicio. Esto significa que el archivo WSDL define todo lo necesario para escribir un programa que interactúe con un Servicio Web. [12] [13] [14] [15]

Cada vez más las empresas exigen aplicaciones más complejas, con menos tiempo y presupuesto que antes. Crear estas aplicaciones, requiere en muchos casos de funcionalidades ya antes implementadas como parte de otros sistemas. SOA (Service Oriented Architecture) nace como una estrategia de integración, expone servicios con funcionalidad bien definida a la aplicación que la requiera. De esta manera, una aplicación final simplemente orquesta la ejecución de un conjunto de estos servicios, añade su lógica particular y le presenta una interfaz al usuario final.

Exponer procesos de negocio como servicios es la clave a la flexibilidad de la arquitectura. Esto permite que otras piezas de funcionalidad (incluso también implementadas como servicios) hagan uso de otros servicios de manera natural, sin importar su ubicación física. Así un sistema evoluciona con la adición de nuevos servicios y su mejoramiento, donde cada servicio evoluciona de una manera independiente. La Arquitectura Orientada a Servicios (SOA) resultante, define los servicios de los cuales estará compuesto el sistema, sus interacciones, y con qué tecnologías serán implementados. Las interfaces que utiliza cada servicio para exponer su funcionalidad son gobernadas por contratos, que definen claramente el conjunto de mensajes soportados, su contenido y las políticas aplicables.

2.4 Entornos Distribuidos. Modelo Cliente Servidor

La arquitectura Cliente-Servidor, es una forma de dividir y especializar programas y equipos de cómputo a fin de que la tarea que cada uno de ellos realiza se efectúe con la mayor eficiencia, y permita simplificar las actualizaciones y mantenimiento del sistema, en una arquitectura monolítica no hay distribución; los tres niveles tienen lugar en el mismo equipo, en el modelo cliente-servidor, en cambio, el trabajo se reparte entre dos ordenadores.

Se puede decir que todas las aplicaciones tienen la misma arquitectura básica y se pueden subdividir en tres partes:

1. Interfaz de Usuario: La presentación al usuario, con las entradas de datos y las consultas.
2. Reglas de Negocio: El procesamiento de la información.
3. Accesos a Datos: El control del almacén de datos.

Ventajas del modelo Cliente Servidor:

- El servidor no necesita una elevada capacidad de procesamiento, parte del proceso se reparte con los clientes.
- Se reduce considerablemente el tráfico de red. El cliente se conecta al servidor cuando es estrictamente necesario, obtiene los datos que necesita y cierra la conexión dejando la red libre.
- El sistema es fácilmente escalable.

2.4.1 Modelo Cliente Servidor de Dos Capas (Two Tier)

En este modelo el sistema se separa en dos partes fijas: cliente y servidor estableciendo un middleware que controla las comunicaciones entre ambas, la lógica de las aplicaciones debe estar en el cliente o en el servidor; la comunicación es transparente para el usuario.

Como limitante:

- El sistema no es escalable: No se adapta fácilmente al número de usuarios.
- El sistema no es manejable ✓
- Bajo rendimiento ✓

2.4.2 Modelo Cliente Servidor de tres Capas (Three Tier)

Cada uno de los componentes de la aplicación en una arquitectura de tres capas se separa en una sola entidad. Esto te 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.

En esta Arquitectura todas las peticiones de los clientes se controlan en la capa correspondiente a la lógica del negocio. Cuando el cliente necesita hacer una petición se la hace a la capa en la que se encuentra la lógica del negocio. Esto es bastante importante pues eso quiere decir que [16]

- El cliente no tiene que tener drivers ODBC ni la problemática consiguiente de instalación de los drivers por tanto se reduce el costo de mantener las aplicaciones cliente.
- El Cliente y el Gestor de Reglas de negocio tienen que hablar el mismo lenguaje (COM, CORBA, SOAP).
- El Gestor de Reglas de Negocio y el Servidor de Datos tienen que hablar el mismo lenguaje (ODBC).

2.4.3 Consideraciones sobre el hardware de la red y el software en los entornos cliente/servidor

La elección entre arquitecturas de dos o tres capas es independiente del hardware con que se cuente. Afortunadamente, en la actualidad el hardware, las redes y los sistemas operativos permiten a los desarrolladores implementar aplicaciones de una forma que antes era absolutamente imposible. Una aplicación desarrollada usando un modelo en tres capas se puede desarrollar en un PC, incluso en un NetPc o en entornos tan diversos como Mac, Microsoft Windows NT™ o servidores Unix dando servicios a aplicaciones de datos en redes Microsoft Windows® o Mac. Una aplicación de tres capas no serviría de nada si no es escalable.

2.4.4 Servidor web Apache

Es el servidor Web más utilizado en el mundo. Su coste gratuito, gran fiabilidad y extensibilidad le convierten en una herramienta potente y muy fiable.

Dentro de sus puntos fuertes se encuentran:

- Tiene interfaz con todos los sistemas de autenticación.

- Facilita la integración como "plug-ins" de los lenguajes de programación de páginas Web dinámicas más comunes.
- Tiene integración en estándar del protocolo de seguridad SSL.
- Provee interfaz a todas las bases de datos.
- Posee Virtual Host.

Apache fue diseñado para proveer un alto grado de calidad y fortaleza para las implementaciones que utilizan el protocolo HTTP. Está ligado a la plataforma (Linux, Windows, UNIX) sobre la cual los individuos o instituciones pueden construir sistemas confiables con fines experimentales o para resolver un problema específico de la organización. [17]

Apache es un software libre, porque sus desarrolladores defienden la teoría de que las transmisiones usando la red deben estar en las manos de todos, y que las compañías de software deben ganar dinero solamente ofertando servicios de valor añadido tales como módulos especializados, soportes y otros, y no adueñándose de un protocolo. Así, el proyecto de crear una implementación robusta con referencia absolutamente libre para quien lo quiera usar es un buen paso para evitar la propiedad sobre los protocolos.

¿Por qué tres capas y Apache? ✓

La aplicación necesitará ser flexible, portable y fiable, es decir, estará en servidores Windows o en la familia de los Unix; esto permitirá resolver complejos problemas inmersos en cambios constantes. Las arquitecturas basadas en tres capas permiten a los componentes de negocio correr en una LAN, WAN o Internet. Esto significa que cualquiera con un ordenador y conexión a la Red posee toda la funcionalidad que tendría si se encontrase delante de su sistema de escritorio.

2.4.5 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.

2.5 Lenguajes de Programación Web

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

Entre los lenguajes que trabajan del lado del servidor podemos citar algunos, que se destacan por ser los más sobresalientes como son PERL, ASP, PHP, Java, JSP, los módulos CGIs e ISAPIs etc., etc. Estos lenguajes desarrollan la lógica de negocio dentro del servidor, además se encargan de los accesos a los distintos Sistemas de Gestión de Bases de Datos. Dentro de los lenguajes que trabajan del lado del cliente se encuentran el JavaScript, XSLT y el Visual Basic Script, estos dos últimos al combinarse con el HTML forman lo que se conoce como DHTML, es decir, salida estándar dinámica o HTML dinámico.

Esta distinción entre los lenguajes ha sido necesaria debido a que el protocolo http es un protocolo sin estado (state less), no guarda información sobre conexiones anteriores y al finalizar la transacción los datos se pierden, cada petición/respuesta es una operación distinta, por lo que la Web trabaja en modo desconectado; o sea, un usuario a través de un navegador hace una petición de una página Web a un Servidor Web (Request), el Servidor obtiene la petición, la procesa y le envía la respuesta al Cliente (Response), este la recibe y se desconecta.

PERL

Es un lenguaje de programación muy utilizado para construir aplicaciones CGI para el Web. PERL es un acrónimo de Practical Extracting and Reporting Language, que viene a indicar que se trata de un lenguaje de programación muy práctico para extraer información de archivos de texto y generar informes a partir del contenido de los ficheros.

Antes estaba muy asociado a la plataforma Unix, pero en la actualidad está disponible en otros sistemas operativos como Windows. PERL es un lenguaje de programación interpretado, al igual que muchos otros lenguajes de Internet como JavaScript o ASP. [21]

ASP

ASP (Active Server Pages) es la tecnología desarrollada por Microsoft para la creación de páginas dinámicas del servidor. ASP se escribe en la misma página Web, utilizando el lenguaje Visual Basic Script o Jscript (JavaScript de Microsoft). La mayor desventaja de este lenguaje es que solo se puede implementar sobre los Servidores Web de su desarrollador: Microsoft. Actualmente se ha presentado ya la segunda versión: el ASP.NET, que comprende algunas mejoras en cuanto a las posibilidades del lenguaje y rapidez con la que funciona. ASP.NET tiene algunas diferencias en cuanto a la sintaxis con el ASP, de modo que tienen formas distintas de utilizarse. Para implementarlo es necesario montar en el Servidor la Plataforma.NET [22]

PHP

PHP (Personal Home Page) es el acrónimo de Hypertext Preprocessor. Es un lenguaje de programación del lado del servidor, gratuito e independiente de plataforma, rápido, con una gran librería de funciones y mucha documentación. Es también un lenguaje interpretado y embebido en el HTML. [23]

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.

Fue creado originalmente en 1994 por Rasmus Lerdorf, pero como PHP está desarrollado en política de código abierto, a lo largo de su historia ha tenido muchas contribuciones de otros desarrolladores. Actualmente PHP se encuentra en su versión 5, que utiliza el motor Zend-2, desarrollado con mayor robustez para cubrir las necesidades de las aplicaciones Web actuales.

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

PHP es la gran tendencia en el mundo de Internet. Últimamente se puede observar un ascenso imparable, ya que cada día son muchísimas más las páginas Web que lo utilizan para su funcionamiento.

Resumiendo, el PHP 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:

- Simplicidad. 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.
- Si bien es cierto que hay ciertas características avanzadas que presentan las plataformas J2EE o .NET y que PHP no las tiene, no todas las aplicaciones Internet ameritan tal grado de complejidad. PHP fácilmente puede cubrir más del 75% de las necesidades del mercado.
- 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 software libre, lo que implica menos costos y servidores más baratos, por lo que podemos usarlo en proyectos comerciales si queremos, sin tener que pagar por su licencia. El tiempo, es uno de los costos más altos que hay que tener en cuenta antes de empezar un proyecto. Para empezar, el tiempo de aprendizaje de PHP es muy corto gracias a su simplicidad. Luego, el tiempo de desarrollo es también corto. Podríamos hacer más de un proyecto Web con PHP en el mismo tiempo que tomaría hacer un solo proyecto con Java o .NET. Otro aspecto que hay que tener en cuenta es el del hardware. Para desarrollar en PHP no se requiere tener grandes capacidades de hardware, como sí lo requieren los pesados IDEs para programar en Java o .Net.

Luego, en el caso de los servidores, una aplicación en PHP no requiere tanta memoria de máquina como podría requerir una aplicación en Java con sus servidores de aplicaciones que podrían requerir hasta varios procesadores y varios Gigas de memoria RAM.

- 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 sintaxis está inspirada en C, ligeramente modificada para adaptarlo al entorno en el que trabaja, de modo que si se está familiarizado con esta sintaxis, le resultará muy fácil aprender PHP.
- 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 ((con lo que no es complicado)) encontrar ayuda, documentación, artículos, noticias, y demás recursos.

Desventaja:

La legibilidad de código puede verse afectada al mezclar sentencias HTML y PHP.

JSP

JSP es un acrónimo de Java Server Pages, que en castellano vendría a ser algo como Páginas de Servidor Java. Es pues, una tecnología orientada a crear páginas Web con programación en Java.

Con JSP podemos crear aplicaciones Web que se ejecuten en variados Servidores Web, de múltiples plataformas, ya que Java es en esencia un lenguaje multiplataforma. Las páginas JSP están compuestas de código HTML/XML mezclando con etiquetas especiales para programar scripts de servidor en sintaxis Java. Por tanto, las JSP podremos escribirlas con nuestro editor HTML/XML habitual.

JAVA SCRIPT

Es un lenguaje de programación interpretado, con capacidades elementales orientadas a objeto. El código Javascript es embebido directamente en el código HTML, haciendo fácil la creación de páginas Web con contenido dinámico. Está diseñado para controlar la apariencia y manipular los eventos dentro de la ventana del navegador Web y es soportado por la gran mayoría de los navegadores. [25]

XSLT

XSLT es el acrónimo de Extensible Stylesheet Language Transformation, es un lenguaje que se usa para convertir documentos XML en otros documentos XML; puede convertir un documento XML que obedezca a un DTD a otro que obedezca otro diferente, un documento XML bien formado a otro que siga un DTD, o, lo más habitual, convertirlo a "formatos finales", tales como WML (usado en los móviles WAP) o XHTML.

Los programas XSLT están escritos en XML, y generalmente, se necesita un procesador de hojas de estilo, o *stylesheet processor* para procesarlas, aplicándolas a un fichero XML. Los programas XSLT están escritos en XML, y generalmente, se necesita un procesador de hojas de estilo, o *stylesheet processor* para procesarlas, aplicándolas a un fichero XML. [26]

¿Por qué PHP y XSLT?

De acuerdo con las características antes expuestas de los múltiples lenguajes existentes, se llega a la conclusión de que los lenguajes a utilizar serán PHP y XSLT. Por la velocidad de PHP a la hora de procesar los datos, ser uno de los lenguajes más universales de la actualidad, por tener una comunidad tan grande y ser soportado en varias plataformas. XSLT es el lenguaje de transformación de documentos, que servirá para definir la presentación del cliente.

2.6 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.
- Minimización 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.

SQL SERVER

Microsoft SQL Server 2000 es uno de los mejores SGDB base de datos para Windows, es el RDBMS de elección para una amplia gama de clientes corporativos y Proveedores Independientes de Software (ISVs) que construyen aplicaciones de negocios. Las necesidades y requerimientos de los clientes han llevado a la creación de innovaciones de producto significativas para facilitar la utilización, escalabilidad, confiabilidad y almacenamiento de datos. [27]

Ventajas:

- Soporta la configuración automática y la auto-optimización.
- Administración multiservidor para un gran número de servidores.
- Gran variedad de opciones de duplicación de cualquier base de datos.
- Acceso universal a los datos (Universal Data Access).
- Fácil de usar.
- Escalabilidad: Se adapta a las necesidades de la empresa, soportando desde unos pocos usuarios a varios miles.
- Potencia: Microsoft SQL Server es la mejor base de datos para Windows NT Server.

- Posee los mejores registros de los benchmarks independientes (TCP) tanto en transacciones totales como en coste por transacción.
- Gestión: Con una completa interfaz gráfica que reduce la complejidad innecesaria de las tareas de administración y gestión de la base de datos.

Desventajas:

- Licencias con costos altos.
- Plataformas Windows.

MySQL

Es un SGBD basado en Open Source (Código abierto) diseñado para los sistemas Unix formando parte de la tecnología LAMP (Linux, Apache, MySQL y PHP), aunque existen versiones para Windows. Actualmente está en su versión 5.0.6-beta incluyendo procedimientos almacenados (stored procedures), disparadores (triggers), vistas (views) y muchas otras características. [28]

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.

Inconvenientes:

- No soporta subconsultas.
- Es gratis para aplicaciones de código abierto, de lo contrario hay que pagar licencia comercial.

POSTGRESQL

PostgreSQL posee una amplia licencia BSD (esta licencia básicamente consiste en que el código puede ser redistribuido y modificado. La FSF lo considera, junto con la licencia GPL, Software libre) y ampliamente utilizado. Posee una estabilidad y confiabilidad legendaria nunca ha presentado caídas en varios años de operación de alta actividad. Fue diseñado para ambientes de alto volumen intentando estar a la altura de Oracle, Sybase o Interbase. Escala muy bien al aumentar el número de CPUs y la cantidad de RAM. Soporta transacciones y desde la versión 7.0, claves ajenas con comprobaciones de integridad referencial. Tiene mejor soporte para subselects, triggers, vistas y procedimientos almacenados en el servidor, además tiene ciertas características orientadas a objetos. Sin embargo consume muchos recursos y no escala bien en la plataforma Windows. [29]

Ventajas:

- Soporta transacciones y desde la versión 7.0, llaves foráneas (integridad referencial).
- Soporta un subconjunto de SQL92 MAYOR que el que soporta MySQL.

Inconvenientes

- Consume bastantes recursos y carga más el sistema.

2.7 Desarrollo basado en RUP bajo la herramienta Rational Rose

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.

El Proceso Unificado es un proceso de desarrollo de software (conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema de software). Es un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas de software, para diferentes áreas de aplicación, diferentes tipos de organizaciones, diferentes niveles de aptitud y diferentes tamaños de proyectos.

El Proceso Unificado está basado en componentes. Utiliza el lenguaje unificado de modelado (UML) para preparar todos los esquemas de un sistema de software. De hecho, UML es una parte esencial de RUP, sus desarrollos fueron paralelos. No obstante los verdaderos aspectos definitorios del proceso unificado se resumen en tres fases claves: dirigido por casos de uso, centrado en la arquitectura, e iterativo e incremental. [30]

2.7.1 UML (Unified Modeling Language)

UML (Unified Modeling Language) o Lenguaje de Modelación Unificado es un lenguaje gráfico para especificar, construir, visualizar y documentar las partes o artefactos (información que se utiliza o produce mediante un proceso de software). Pueden ser artefactos: un modelo, una descripción que comprende el desarrollo de software que se basen en el enfoque Orientado a Objetos, utilizándose también en el diseño Web. UML usa procesos de otras metodologías, aprovechando la experiencia de sus creadores, eliminó los componentes que resultaban de poca utilidad práctica y añadió nuevos elementos. [31]

UML es un lenguaje más expresivo, claro y uniforme que los anteriores definidos para el diseño Orientado a Objetos, que no garantiza el éxito de los proyectos pero si mejora sustancialmente el desarrollo de los mismos, al permitir una nueva y fuerte integración entre las herramientas, los procesos y los dominios.

De forma general 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.

UML es desde finales de 1997, un lenguaje de modelado orientado a objetos estándar, de acuerdo con el Object Management Group, siendo utilizado diariamente por grandes organizaciones como: Microsoft, Oracle, Rational.

2.7.2 Rational Rose

Existen herramientas Case de trabajo visuales como el Analise, el Designe y el Rational Rose, que permiten realizar el modelado del desarrollo de los proyectos, en la actualidad la mejor y más utilizada en el mercado mundial es Rational Rose y es la que se utiliza en la modelación de este proyecto.

La Corporación Rational ofrece un Proceso Unificado (RUP) para el desarrollo de los proyectos de software, desde la etapa de Ingeniería de Requerimientos hasta la de pruebas. Para cada una de estas etapas existe una herramienta de ayuda en la administración de los proyectos, Rose es la herramienta del Rational para la etapa de análisis y diseño de sistemas.

Rose es una herramienta con plataforma independiente que ayuda a la comunicación entre los miembros de equipo, a monitorear el tiempo de desarrollo y a entender el entorno de los sistemas. Una de las grandes ventajas de Rose es que utiliza la notación estándar en la arquitectura de software (UML), la cual permite a los arquitectos de software y desarrolladores visualizar el sistema completo utilizando un lenguaje común, además los diseñadores pueden modelar sus componentes e interfaces en forma individual y luego unirlos con otros componentes del proyecto.

La metodología usada para desarrollar el proyecto fue RUP. Este es un proceso que garantiza la elaboración de todas las fases de un producto de software orientado a objeto. RUP utiliza UML, que es un lenguaje que permite la modelación de sistemas con tecnología orientada a objetos.

Rational Rose permite completar una gran parte de las disciplinas (flujos fundamentales) del proceso unificado de Rational (RUP), en concreto [32]:

- Modelado del negocio.
- Captura de requisitos (parcial).
- Análisis y diseño (completo).
- Implementación (como ayuda).
- Control de cambios y gestión de configuración (parte).

Características principales

- Admite como notaciones: UML, COM, OMT y Booch.
- Realiza Chequeo semántico de los modelos.
- Ingeniería “de ida y vuelta”: Rose permite generar código a partir de modelos y viceversa.
- Desarrollo multiusuario.
- Integración con modelado de datos.
- Generación de documentación.
- Tiene un lenguaje de script para poder ampliar su funcionalidad.

2.8 Plataforma de Servicio (PLASER)

La plataforma de Servicios (PLASER) está conformada fundamentalmente por varias clases en PHP, una librería, que puede usada opcionalmente para que un componente se integre al Registro Informatizado de la Salud (RIS), pero 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.

Este sistema esta 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; 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

2.9 Herramientas a utilizar

Se decidió que se utilizaría el Rational Rose Enterprise Edition 2003, para sustentar la documentación, como modelador visual de la notación UML (Unified Modeling Language) para la confección de los diagramas que se ilustran en este documento. Esta herramienta es muy completa y ofrece amplias potencialidades.

Para el diseño de las páginas Web será utilizado en una primera etapa el Dreamweaver MX de la familia Macromedia. También el Stylus Studio 5.1 que se utilizará por los desarrolladores para crear los ficheros XSL a través de escenarios.

Para la edición del código PHP se utilizará el NuSphere y PHPEd 3.3.3, este último es un editor para programadores con soporte para múltiples formatos, similar a otros editores como el Zend Studio, incluye un cliente de FTP y un servidor Web integrado, como servidor Web Apache 2.0 con PHP 4.3.4; el servidor de bases de datos se escogerá MySQL 4.x.x y PHPmyadmin como su front.

2.10 Registro Informatizado de Salud (RIS)

El Registro Informatizado de la Salud (RIS), es un sistema disponible en la Red Telemática de Salud Infomed, concebido completamente sobre Arquitectura Basada en Componentes y Orientada a Servicios usando XML WebServices, específicamente SOAP; utiliza PLASER (Plataforma de Servicios) para la programación y homogeneidad de los módulos. En el mismo se recogen algunos servicios tales como Registro de Localidad, Registro de Unidades de Salud, Registro del Ciudadano, Registro de Áreas de Salud, además del Registro de la Clasificación internacional de enfermedades y problemas relacionado con la salud.

2.11 Conclusiones

En este capítulo se profundizó en el conocimiento de algunos conceptos necesarios para la comprensión de este trabajo. Además se realizó un análisis completo de las tecnologías que serán utilizadas a lo largo del desarrollo del sistema propuesto, y se fundamentaron las elecciones del lenguaje, el sistema gestor de bases de datos, y la metodología a utilizar.

CAPÍTULO 3 DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA

3.1 Introducción

Para tener una idea clara de la solución propuesta, primero es necesario entender el contexto donde se desarrolla el sistema, para lo cual existen dos métodos fundamentales, el primero, Modelo del Negocio, está dirigido a entender el funcionamiento del negocio dentro de las organizaciones y el segundo Modelo del Dominio, se utiliza para capturar los tipos de objetos más importantes en el contexto del sistema. En el siguiente epígrafe se podrá apreciar cual fue el método utilizado y porque se seleccionó.

Mediante el diagrama de casos de uso y la descripción de cada uno de ellos, se definen los límites del sistema y las relaciones entre el sistema y el entorno y a su vez cada caso de uso responde a uno o varios requerimientos funcionales del sistema, los cuales aparecen enumerados en el epígrafe correspondiente.

Todos los aspectos anteriores, unidos a la descripción general del sistema y a los requerimientos no funcionales aportan una visión clara de cómo está concebido este sistema.

3.2 Modelo de Dominio

El modelo del dominio o modelo conceptual, nos 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 del dominio e información del dominio. Aprovechando las bondades de los diagramas UML para representar cosas y conceptos el diagrama del modelo del dominio se presenta en forma de diagrama de clases donde figuran los principales conceptos y roles del sistema analizado.

3.2.1 ¿Por que Modelo de dominio?

Nos permite de manera visual mostrar al usuario los principales conceptos que se manejan en el dominio del sistema en desarrollo. Esto ayuda a los usuarios, clientes y desarrolladores e interesados, a utilizar un vocabulario común para poder entender el contexto en que se emplaza el sistema. Para capturar correctamente los requisitos y poder construir un sistema correcto se necesita tener un firme conocimiento del funcionamiento del objeto de estudio. Este modelo va a contribuir posteriormente a identificar algunas clases que se utilizarán en el sistema.

Primeramente vamos a identificar todos los conceptos que se utilizarán en el diagrama, mediante un glosario de términos sobre los nombres

3.2.2 Glosario de términos

RCIE: abreviatura utilizada para hacer referencia al ^Rregistro de Codificador Internacional de _Eenfermedades y problemas de salud

Capítulos: Entidad reservada para almacenar toda la información referente a los Capítulos dentro del RCIE.

Grupos: Entidad reservada para almacenar toda la información referente a los Grupos.

Subcategorías: Entidad reservada para almacenar toda la información referente a las Subcategorías dentro del RCIE.

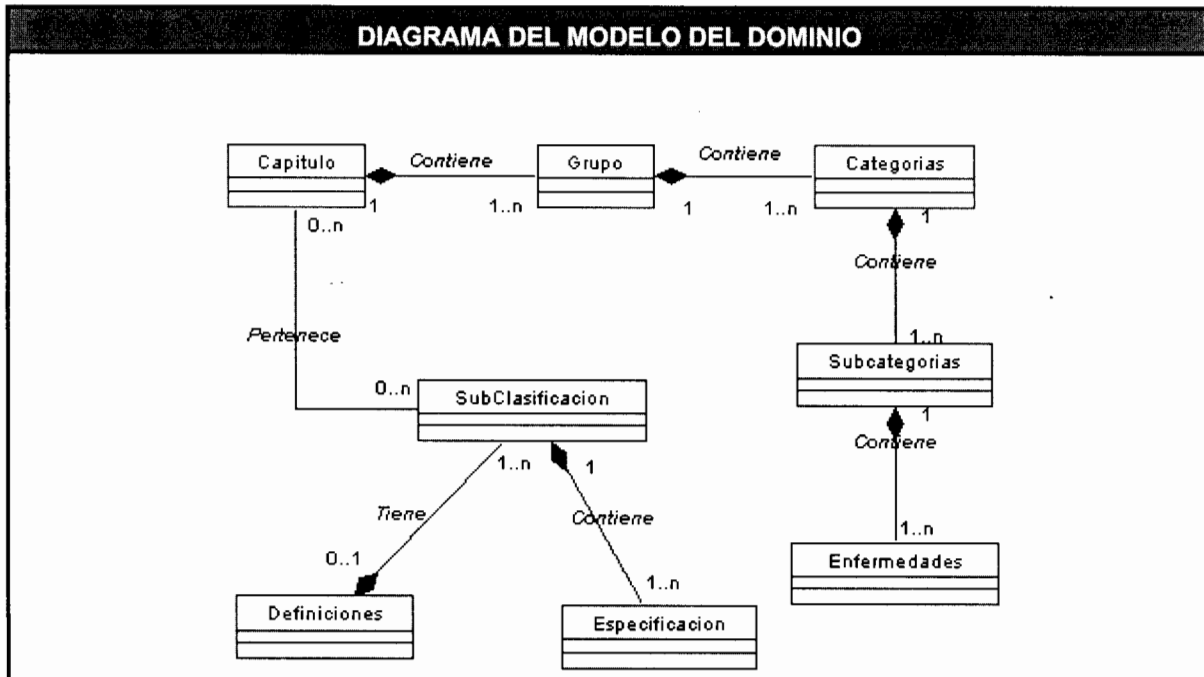
Enfermedades: Entidad reservada para almacenar toda la información referente a las Enfermedades dentro del RCIE.

SubClasificación: Entidad reservada para almacenar toda la información referente a los códigos opcionales de clasificación dentro del RCIE.

Definiciones: Entidad reservada para almacenar toda la información referente a las definiciones adicionales que deben tenerse en cuenta a la hora de codificar una enfermedad o problema de salud con un código opcional dentro del RCIE.

Especificación: Entidad reservada para almacenar toda la información referente a cada una de las especificaciones dentro de una misma subclasificación dentro del RCIE.

3.3 Diagrama del Modelo del Dominio



3.4 Requerimientos funcionales

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**:

- Código Capitulo ✓
- Descripción
- Incluye
- Excluye
- Observaciones

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

- Código Grupo
- Descripción

- Incluye
- Excluye
- Observaciones

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

- Código Categoría
- Descripción
- Incluye
- Excluye
- Observaciones

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

- Código Subcategoría
- Descripción
- Incluye
- Excluye
- Observaciones

RF1.5 Gestionar (adicionar, modificar y eliminar) la información de las **Enfermedades**:

- Código Enfermedad
- Descripción

RF1.6 Gestionar (adicionar, modificar, eliminar) la información de los códigos adicionales o opcionales dentro del CIE.

- Código de la subclasificación.
- Descripción subclasificación.
- Definiciones para cada subclasificación.

RF2. Permitir búsquedas en el CIE según los distintos criterios de selección que pueda necesitar el cliente.

RF2.1 Permitir búsquedas por código en todas las entidades (Capítulos, Grupos, Categorías, Subcategorías, Enfermedades, códigos opcionales o subclasificaciones y definiciones) que se gestionan en el CIE.

RF2.2 Permitir búsquedas por Descripción subconjuntos de descripciones en todas las entidades (Capítulos, Grupos, Categorías, Subcategorías, Enfermedades, códigos opcionales o subclasificaciones y definiciones) que se gestionan en el CIE.

RF2.3 Permitir búsquedas avanzadas por distintos criterios de selección en las entidades seleccionada según la opción escogida por el usuario dentro de las entidades (Capítulos, Grupos, Categorías, Subcategorías, Enfermedades) que se gestionan en el CIE.

RF3. Autenticar a todo usuario que necesite permiso para la gestión de la información antes de entrar al sistema.

3.5 Requerimientos no funcionales

3.5.1 Requisitos de Apariencia o Interfaz Externa

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

3.5.2 Requisitos de usabilidad

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

3.5.3 Requisitos de Portabilidad

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

3.5.4 Requisitos de Seguridad

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

DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA.

- ✓ La autenticación será la primera acción del administrador 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.
- ✓ Si el **usuario autenticado** no se encuentra registrado se debe reportar un error de acceso.
- ✓ Si el **usuario autenticado** se encuentra registrado se autoriza su acceso y se crea un certificado digital.
- ✓ 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.
 - Un listado de los módulos que el usuario tiene acceso y el tipo de acceso en cada uno de ellos (Editor o Visualizador).
- ✓ Deberá registrarse el token, el identificador del usuario presentes en el certificado digital y el día, mes, año, hora, minuto, segundo de la creación del certificado.
- ✓ 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.
- ✓ Si el token se encuentra registrado y la petición se corresponde con los derechos del usuario se procederá a ejecutar la petición solicitada.
- ✓ 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.
- ✓ 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.

3.5.5 Requisitos de Confiabilidad

- ✓ Prevenir posibles fallos y recuperarse ante ellos.

DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA.

- ✓ La información manejada por el sistema será objeto de cuidadosa protección contra la corrupción y estados inconsistentes. Deberán existir mecanismos de chequeo de integridad.
- ✓ 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.
- ✓ Se permitirá la creación de copias de respaldo que puedan restaurar el sistema en caso de fallo crítico o pérdida total de la información.

3.5.6 Requisitos de Interfaz Interna

- ✓ Todos los componentes del sistema deben desarrollarse siguiendo el principio de máxima **cohesión** y mínimo **acoplamiento**.
- ✓ Los componentes reutilizables entre los módulos que componen el Registro Informatizado de Salud deberán ser desarrollados como **servicios Web XML** que interactúan a través de SOAP con otros componentes.

3.5.7 Requisitos de Software

- ✓ La lógica de presentación constituirá una capa independiente de la lógica de negocio, centrandó su función en la interfaz de usuario y validaciones simples de los datos de entrada.
- ✓ RNF9 Los clientes tendrán acceso al Registro Informatizado de Salud (RIS) a través de cualquier navegador Web, recomendado Mozilla 1.5, Internet Explorer 5.0 o superior.
- ✓ RNF10 Las aplicaciones legacy deberán integrar los componentes que estén desarrollados según los requerimientos tecnológicos que está evaluando SOFTEL.
- ✓ RNF11 Diseñar una interfase donde el personal que trabaja con la aplicación, seleccione y organice la información que desee mostrar con sólo hacer clic.
- ✓ RNF12 Disponer de instrucciones en una opción de ayuda.

- ✓ RNF13 Incluir las enfermedades dentro de las Subcategorías, las Subcategorías dentro de las Categorías, las Categorías dentro de los Grupos, y los Grupos dentro de los Capítulos. Acorde con la taxonomía de la Clasificación Internacional de Enfermedades y Problemas Relacionados con la Salud vigente.
- ✓ RNF14 Validar el proceso de la captación de datos para evitar entradas inadecuadas.

3.5.8 Requisitos de Hardware

- ✓ 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.
- ✓ Impresora local o de red para imprimir los reporte solicitados.

3.5.9 Requisitos de Diseño y la Implementación

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

3.6 Descripción del sistema propuesto

3.6.1 Concepción general del sistema

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

Una clasificación de enfermedades puede definirse como un sistema de categorías a las que se asignan entidades morbosas, de conformidad con criterios establecidos. La clasificación puede girar en torno a varios posibles ejes, y la elección de uno en particular estará determinada por el uso que se hará de las estadísticas recopiladas. Una clasificación estadística de enfermedades debe abarcar toda la gama de estados morbosos dentro de un número manejable de categorías. Las clasificaciones de las enfermedades se agrupan en Capítulos, Grupos, Categorías y Subcategorías.

Por lo antes expuesto se decidió que el RCIE se conciba lo mas semejante posible al formato del libro para permitir su familiaridad a los distintos clientes que puedan hacer uso

DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA.

del mismo. Estos se agrupan 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. Se debe tener en cuenta que una enfermedad o problema de salud puede ser codificado de disímiles maneras y es por ello que algunas de las enfermedades pueden tener implícita en su código, por convenio de la OMS un código opcional.

A raíz de esto uno de los principales motivos de la modelación e implementación de este software es servir de complemento en la automatización de la atención primaria salud, para permitir que se estandaricen los diagnósticos médicos, con vista a futuros estudios estadísticos a diversos niveles organizativos.

3.6.2 Actores del sistema

Actores del sistema	Justificación
Codificador de la Dirección Nacional de Estadística del MINSAP.	Es el encargado de la gestión de la información (crear, modificar, eliminar) dentro de las distintas entidades dentro del RCIE.
Usuario	Usuario (todos los que soliciten CIE desde cualquier lugar, no necesitan ser autenticados) es el encargado de hacer búsquedas y recibir información.

3.6.3 Modelo de casos de uso del sistema

3.6.3.1 Gestionar Capítulo

Nombre del caso de uso	Gestionar Capítulos.
Actores	Codificador de la Dirección Nacional de Estadística del MINSAP
Propósito	Mantener actualizada la información referente a los capítulos dentro del RCIE
<p>El Caso de Uso comienza cuando el actor solicita actualizar o adicionar informaciones dentro de la entidad capítulos, si se selecciono la opción de modificar la información el sistema le muestra la interfaz de modificación donde podrá realizar distintas operaciones dentro de la misma como eliminar, buscar o modificar la información o si escoge la opción de adicionar un nuevo elemento el sistema le mostrara la interfaz adecuada para realizar su inserción dentro de la entidad capitulo, luego se oprime el botón respectivo para cada acción y el sistema se encarga de actualizar sus datos.</p>	
Referencias	RF1.1, RF3.
Precondiciones	El usuario debe haberse autenticado y tener permiso de edición dentro del RCIE.
Poscondiciones	El sistema deja actualizada la información de la entidad capítulos dentro del RCIE.
Prototipo.	

3.6.3.2 Gestionar Grupos

Nombre del caso de uso	Gestionar Grupos.
Actores	Codificador de la Dirección Nacional de Estadística del MINSAP
Propósito	Mantener actualizada la información referente a los grupos dentro de el RCIE
<p>El Caso de Uso comienza cuando el actor solicita actualizar o adicionar informaciones dentro de la entidad grupos, si se selecciono la opción de modificar la información el sistema le muestra la interfaz de modificación donde podrá realizar distintas operaciones dentro de la misma como eliminar, buscar o modificar la información o si escoge la opción de adicionar un nuevo elemento el sistema le mostrara la interfaz adecuada para realizar su inserción dentro de la entidad grupos, luego se oprime el botón respectivo para cada acción y el sistema se encarga de actualizar sus datos.</p>	
Referencias	RF1.2, RF3.
Precondiciones	El usuario debe haberse autenticado y tener permiso de edición dentro del RCIE.
Poscondiciones	El sistema deja actualizada la información de la entidad grupos dentro del RCIE.
Prototipo	

3.6.3.3 Gestionar Categorías

Nombre del caso de uso	Gestionar Categorías.
Actores	Codificador de la Dirección Nacional de Estadística del MINSAP
Propósito	Mantener actualizada la información referente a las categorías dentro de el RCIE
<p>El Caso de Uso comienza cuando el actor solicita actualizar o adicionar informaciones dentro de la entidad categorías, si se selecciono la opción de modificar la información el sistema le muestra la interfaz de modificación donde podrá realizar distintas operaciones dentro de la misma como eliminar, buscar o modificar la información o si escoge la opción de adicionar un nuevo elemento el sistema le mostrara la interfaz adecuada para realizar su inserción dentro de la entidad categorías , luego se oprime el botón respectivo para cada acción y el sistema se encarga de actualizar sus datos.</p>	
Referencias	RF1.3, RF3.
Precondiciones	El usuario debe haberse autenticado y tener permiso de edición dentro del RCIE.
Poscondiciones	El sistema deja actualizada la información de la entidad categorías dentro del RCIE.
Prototipo	

3.6.3.4 Gestionar Subcategorías

Nombre del caso de uso	Gestionar Subcategorías.
Actores	Codificador de la Dirección Nacional de Estadística del MINSAP
Propósito	Mantener actualizada la información referente a las subcategorías dentro de el RCIE
<p>El Caso de Uso comienza cuando el actor solicita actualizar o adicionar informaciones dentro de la entidad subcategorías, si se selecciono la opción de modificar la información el sistema le muestra la interfaz de modificación donde podrá realizar distintas operaciones dentro de la misma como eliminar, buscar o modificar la información o si escoge la opción de adicionar un nuevo elemento el sistema le mostrara la interfaz adecuada para realizar su inserción dentro de la entidad subcategorías , luego se oprime el botón respectivo para cada acción y el sistema se encarga de actualizar sus datos.</p>	
Referencias	RF1.4, RF3.
Precondiciones	El usuario debe haberse autenticado y tener permiso de edición dentro del RCIE.
Poscondiciones	El sistema deja actualizada la información de la entidad subcategorías dentro del RCIE.
Prototipo	

3.6.3.5 Gestionar Enfermedades

Nombre del caso de uso	Gestionar Enfermedades.
Actores	Codificador de la Dirección Nacional de Estadística del MINSAP
Propósito	Mantener actualizada la información referente a las enfermedades dentro de el RCIE
<p>El Caso de Uso comienza cuando el actor solicita actualizar o adicionar informaciones dentro de la entidad enfermedades, si se selecciono la opción de modificar la información el sistema le muestra la interfaz de modificar donde podrá realizar distintas operaciones dentro de la misma como eliminar, buscar o modificar la información o si escoge la opción de adicionar un nuevo elemento el sistema le mostrara la interfaz adecuada para realizar su inserción dentro de la entidad enfermedades, luego se oprime el botón respectivo para cada acción y el sistema se encarga de actualizar sus datos.</p>	
Referencias	RF1.5, RF3.
Precondiciones	El usuario debe haberse autenticado y tener permiso de edición dentro del RCIE.
Poscondiciones	El sistema deja actualizada la información de la entidad enfermedades dentro del RCIE.
Prototipo	

3.6.3.6 Gestionar Subclasificaciones código opcional

Nombre del caso de uso	Gestionar Subclasificaciones código opcional
Actores	Codificador de la Dirección Nacional de Estadística del MINSAP
Propósito	Mantener actualizada la información referente a los subclasificaciones dentro de el RCIE
<p>El Caso de Uso comienza cuando el actor solicita actualizar o adicionar informaciones dentro de la entidad subclasificaciones, si se selecciono la opción de modificar la información el sistema le muestra la interfaz de modificar donde podrá realizar distintas operaciones dentro de la misma como eliminar, buscar o modificar la información o si escoge la opción de adicionar un nuevo elemento el sistema le mostrara la interfaz adecuada para realizar su inserción dentro de la entidad subclasificaciones , luego se oprime el botón respectivo para cada acción y el sistema se encarga de actualizar sus datos.</p>	
Referencias	RF1.6, RF3.
Precondiciones	El usuario debe haberse autenticado y tener permiso de edición dentro del RCIE.
Poscondiciones	El sistema deja actualizada la información de la entidad subclasificaciones dentro del RCIE.
Prototipo	

3.6.3.7 Gestionar Definiciones

Nombre del caso de uso	Gestionar Definiciones
Actores	Codificador de la Dirección Nacional de Estadística del MINSAP
Propósito	Mantener actualizada la información referente a las definiciones dentro de el RCIE
<p>El Caso de Uso comienza cuando el actor solicita actualizar o adicionar informaciones dentro de la entidad definiciones, si se selecciono la opción de modificar la información el sistema le muestra la interfaz de modificar donde podrá realizar distintas operaciones dentro de la misma como eliminar, buscar o modificar la información o si escoge la opción de adicionar un nuevo elemento el sistema le mostrara la interfaz adecuada para realizar su inserción dentro de la entidad definiciones , luego se oprime el botón respectivo para cada acción y el sistema se encarga de actualizar sus datos.</p>	
Referencias	RF2.1, RF2.2, RF2.3.
Precondiciones	El usuario debe haberse autenticado y tener permiso de edición dentro del RCIE.
Poscondiciones	El sistema deja actualizada la información de la entidad definiciones dentro del RCIE.
Prototipo	

3.6.3.8 Buscar Información

Nombre del caso de uso	Buscar Información
Actores	Codificador de la Dirección Nacional de Estadística del MINSAP, usuarios
Propósito	Permitir realizar búsquedas dentro del RCIE por los disímiles parámetros de entrada según el criterio de selección del usuario.
<p>El Caso de Uso comienza cuando el actor solicita hacer alguna búsqueda por los distintos parámetros dentro del RCIE. Si selecciono <u>✓</u> buscar por códigos, el sistema hará una búsqueda minuciosa por todas las entidades de la aplicación buscando el código deseado enviando una respuesta del registro deseado; si selecciono buscar por descripción el sistema realiza una búsqueda minuciosa en toda las entidades de mi aplicación buscando donde aparece al menos el subconjunto de la cadena tecleada enviando como respuesta un subconjunto de los datos existente en la base de datos; si selecciono <u>✓</u> la opción de búsqueda avanzada el sistema le da la opción de realizar búsquedas en determinados lugares en específico y bajo ciertos criterios.</p> <p>Si la búsqueda es satisfactoria el sistema devuelve las respectivas respuestas en virtud de la solicitud realizada; sino devuelve un mensaje indicando que no se encuentran elementos con las condiciones dada por el usuario.</p>	
Referencias	RF2.1, RF2.2, RF2.3.
Precondiciones	El usuario no necesita haberse autenticado dentro del RCIE para poder realizar las búsquedas bajos los distintos criterios dentro del RCIE.
Poscondiciones	El sistema devuelve las posibles respuestas del resultado de la búsqueda.
Prototipo	

3.6.3.9 Mostrar Información.

Nombre del caso de uso	Mostrar Información
Actores	Codificador de la Dirección Nacional de Estadística del MINSAP, usuarios
Propósito	Permitir brindar el resultado de las diferentes búsquedas realizadas dentro del RCIE
<p>El caso de uso comienza cuando el actor necesita el resultado de alguna búsqueda por algún parámetro en específico y lo visualiza de una forma amena para el usuario. Si el resultado de la búsqueda no devuelve ningún resultado, envía un mensaje de sugerencia para cambiar el criterio de búsqueda o lugar donde se está realizando.</p>	
Referencias	RF2.1, RF2.2, RF2.3, buscar información [incluye].
Precondiciones	El usuario no necesita haberse autenticado dentro del RCIE para poder y haber realizado el caso de uso Buscar Información.
Poscondiciones	El sistema devuelve las posibles respuestas del resultado de la búsqueda y las visualiza de forma amena para el usuario.
Prototipo	

3.6.4 Diagrama de Casos de Uso del Sistema

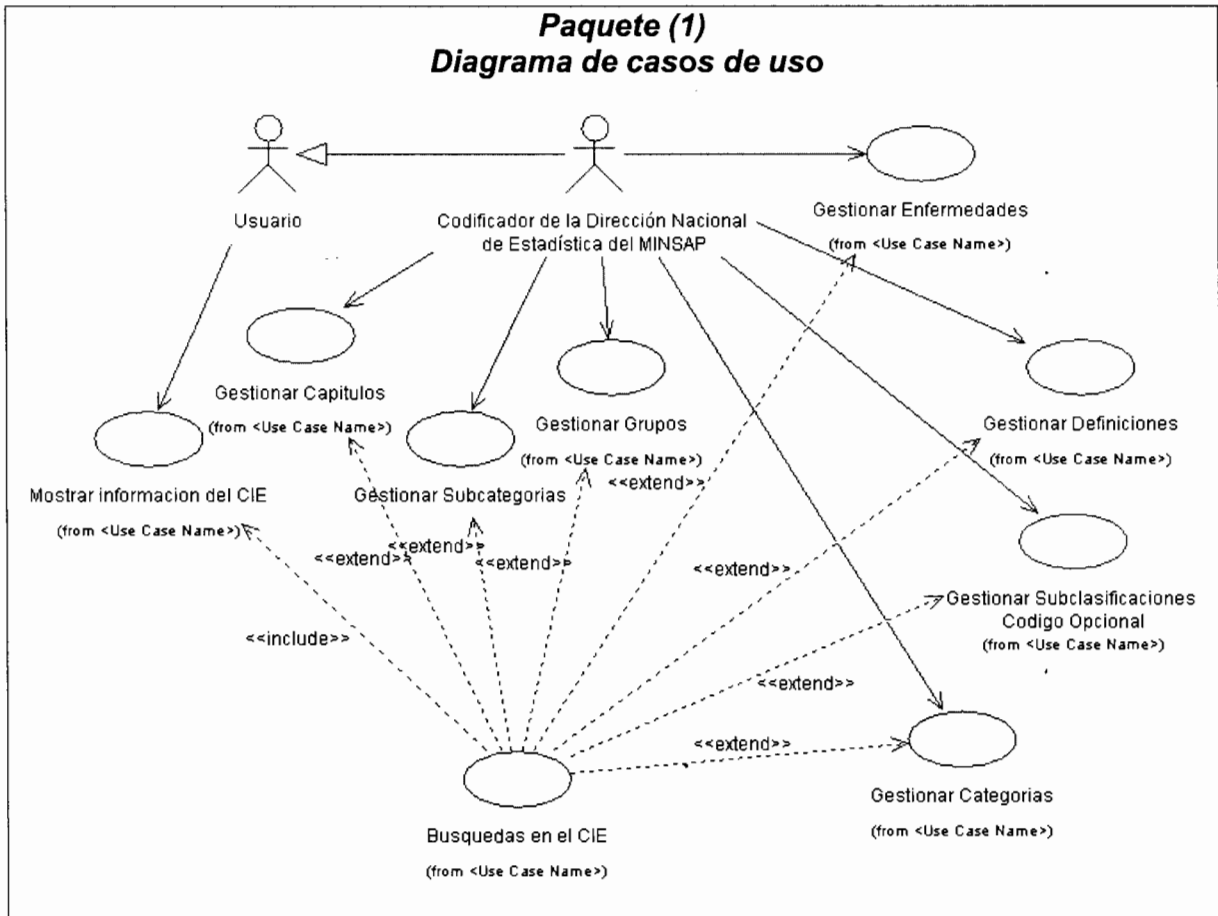


Fig.3 diagrama de casos de uso del sistema.

3.7 Conclusión

Los artefactos resultantes en los flujos de trabajo de requisitos y análisis crean una base sólida en el sistema en construcción, generando varios prototipos de confrontación con el cliente para evitar malos entendidos en las siguientes iteraciones.

CAPÍTULO 4 CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

4.1 Introducción

El diseño es la parte del proceso de desarrollo de software que tiene como objetivo principal definir la estructura del sistema en estudio, siempre basándose en los requisitos funcionales que fueron seleccionados en las etapas anteriores.

En este capítulo se muestran los principales artefactos UML obtenidos en los flujos de trabajo de diseño e implementación según RUP. Para su estudio más detallado el diagrama de clases se estructuró por paquetes, obteniendo como resultado el diagrama de clases persistentes y su modelo lógico de datos. Finalizando se presentarán el modelo de implementación mediante el diagrama de componentes y de despliegue que resultaron del diseño realizado de cada uno de los casos de uso del sistema.

4.2 Diagrama de clases

Estereotipos Web

Una de las características más relevantes de la notación UML es su capacidad para absorber nueva semántica sin romper su lógica interna. La necesidad de implementar Servicios Web a través de complejas arquitecturas con múltiples capas de componentes y una gran dispersión geográfica de nodos, ha supuesto todo un reto al abordar su modelado y especificación. Por lo que Jim Conallen ha desarrollado desde 1998 una extensión de la notación UML denominada WAE "Web Application Extensión" que permite rentabilizar toda la gramática interna de UML para modelar aplicaciones con elementos específicos de la arquitectura de un entorno Web.

Alguien dijo alguna vez:

"UML es una caja de herramientas, como un cajón de sastre, no tienes por qué usarlas todas y ni siquiera tienes por qué usarlas para lo que está especificado que se usen".

El proceso de modelado que se ha ejecutado de los módulos que formarán parte del Sistema Integral de Salud en su primera prioridad, se realizó hasta la etapa de diseño, utilizando los estereotipos que ofrece el lenguaje UML, resultando suficientes para dicho modelado.

A pesar de estar creadas todas las condiciones para usar la notación UML denominada WAE por las características propias del proyecto y la arquitectura de los módulos que

están en desarrollo, no se utilizaron estos estereotipos web por la forma en que se ha organizado y ejecutado el Proyecto APS.

La capacitación y preparación acelerada en el uso de las herramientas de modelado con Rational Rose y del lenguaje UML, por parte de todos los integrantes del equipo de trabajo, para modelar todos los procesos de los módulos de esta primera etapa, no dejó espacio de tiempo para continuar la modelación incorporando también el estudio de los temas relacionados con los estereotipos web y a la vez aplicarlos.

Existen además plazos ambiciosos de entrega de los módulos actuales y compromisos con la máxima dirección del MINSAP, de comenzar la implantación piloto de los módulos que se comiencen a liberar en la fase de implementación, en 9 de los 445 policlínicos a partir de septiembre del presente año.

Se considera que para el proceso de modelado de las próximas etapas del proyecto se deben usar estos estereotipos web, aplicando toda la experiencia acumulada en la concepción de la etapa actual.

Propósito de algunas clases Utilizadas.

CLASE	PROPÓSITO
Dbz_class	Clase para la conexión con bases de datos MySql, usa el modulo dbx de PHP para su funcionalidad. Además crea un objeto conexión que permite hacer consultas, y recuperar los resultados; insertar, eliminar y actualizar datos. Esta clase estará en la capa de Acceso a Datos. Esta clase está dentro de un framework llamado PLASER utilizado por la empresa para la reutilización de los distintos componentes y lograr una calidad adecuada, que implementa el patrón de diseño Modelo Vista Controlador (MVC).
Fachada	Clase general que se encuentra dentro del framework PLASER, que permite aplicar un encapsulamiento al mismo, creándose de esta forma un nivel de abstracción, logrando una mejor arquitectura del sistema.
Fachada CIE	Clase que hereda de Fachada, que aplica una fachada en la capa de presentación, disminuyendo así la carga de negocio en gran medida. Esta clase implementa el patrón Fachada, de esta forma la aplicación solo le hará las peticiones a esta clase.

4.2.1 Gestionar Capítulos

Descripción: En este diagrama se visualiza las posibles operaciones y sus clases cuando se desea gestionar los Capítulos, el cual tiene como principal objetivo mantener totalmente actualizada la información relacionada con la entidad. El usuario puede seleccionar algunas opciones como adicionar, eliminar o modificar y el sistema será el encargado de visualizar las distintas pantallas para su realización.

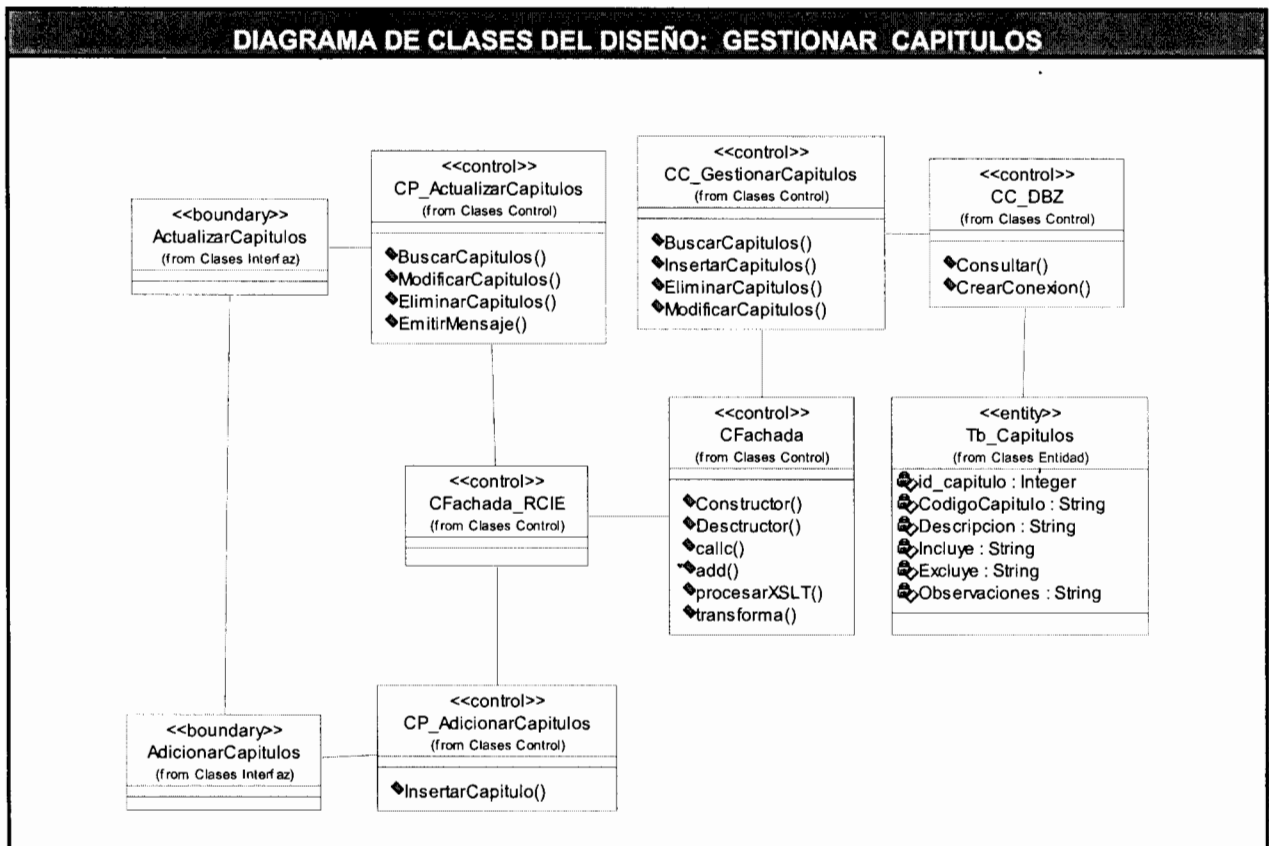


Fig. 4 Diagrama de clases Gestionar Capítulos.

4.2.2 Gestionar Grupos

Descripción: En este diagrama se visualiza las posibles operaciones y sus clases cuando se desea gestionar los Grupos, el cual tiene como principal objetivo mantener totalmente actualizada la información relacionada con la entidad. El usuario puede seleccionar algunas opciones como adicionar, eliminar o modificar y el sistema será el encargado de visualizar las distintas pantallas para su realización.

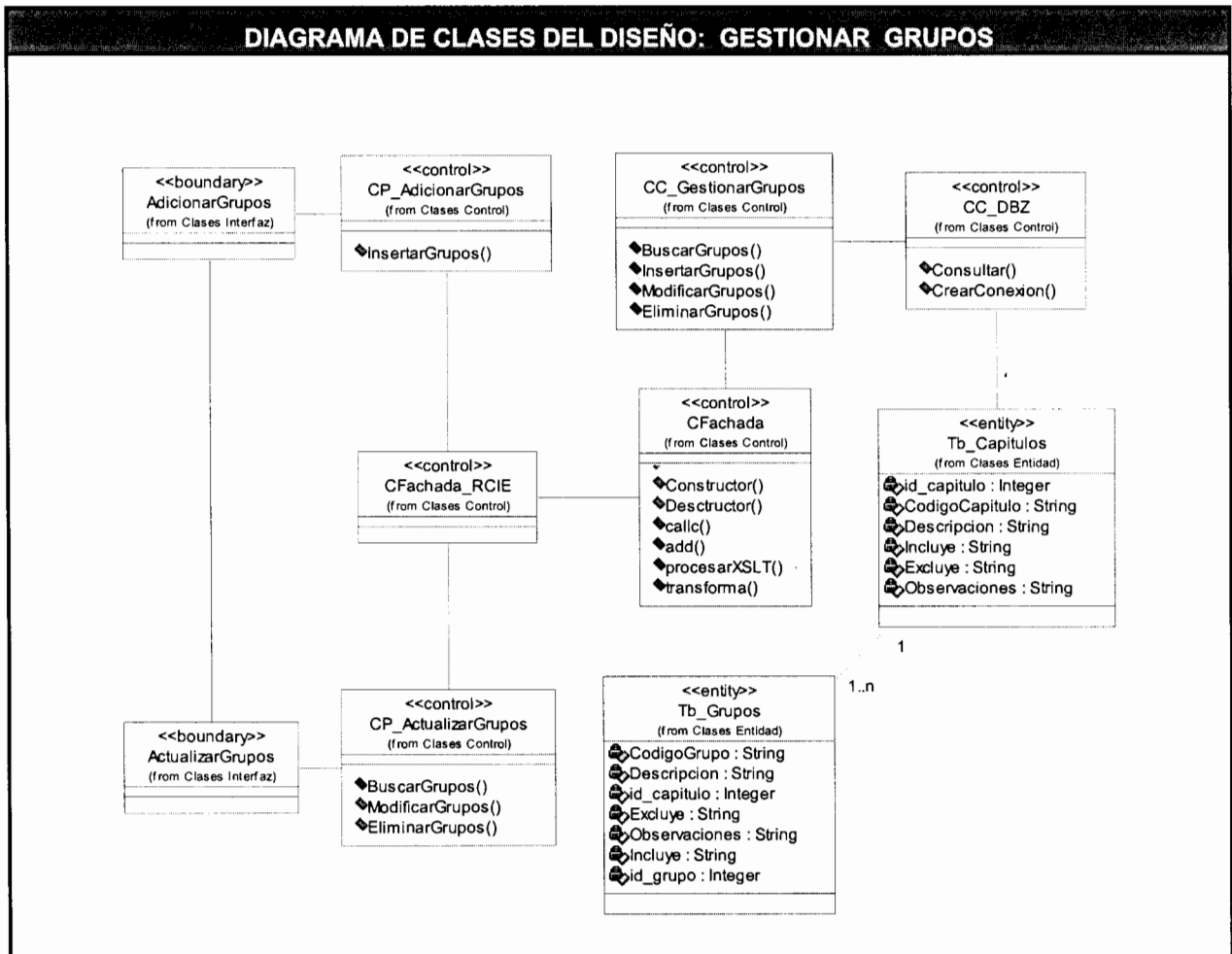


Fig. 5 Diagrama de clases Gestionar Grupos.

4.2.3 Gestionar Categorías

Descripción: En este diagrama se visualiza las posibles operaciones y sus clases cuando se desea gestionar las Categorías, el cual tiene como principal objetivo mantener totalmente actualizada la información relacionada con la entidad. El usuario puede seleccionar algunas opciones como adicionar, eliminar o modificar y el sistema será el encargado de visualizar las distintas pantallas para su realización.

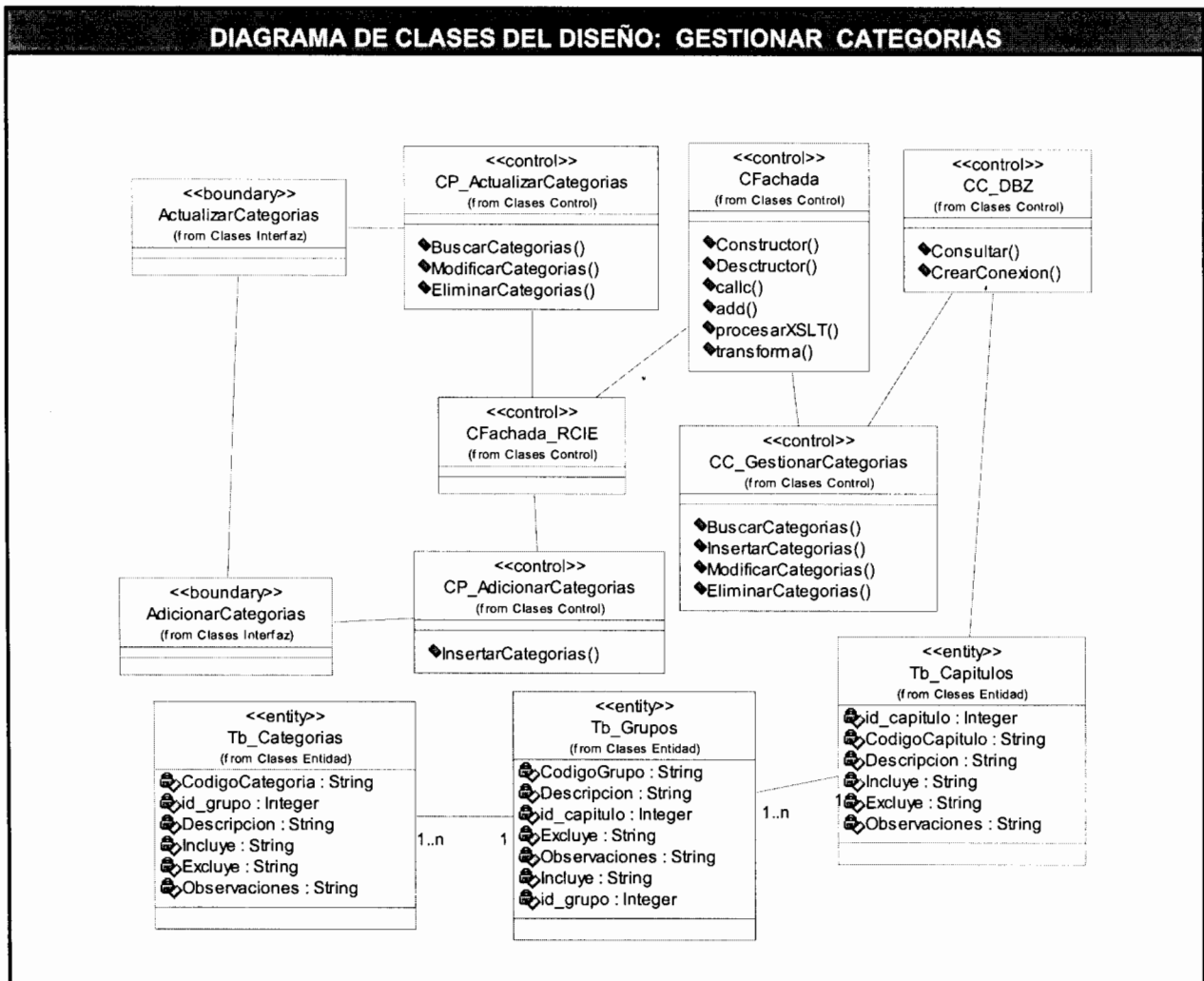


Fig.6 Diagrama de clases Gestionar Categorías.

4.2.5 Gestionar Enfermedades

Descripción: En este diagrama se visualiza las posibles operaciones y sus clases cuando se desea gestionar las Enfermedades, el cual tiene como principal objetivo mantener totalmente actualizada la información relacionada con la entidad. El usuario puede seleccionar algunas opciones como adicionar, eliminar o modificar y el sistema será el encargado de visualizar las distintas pantallas para su realización.

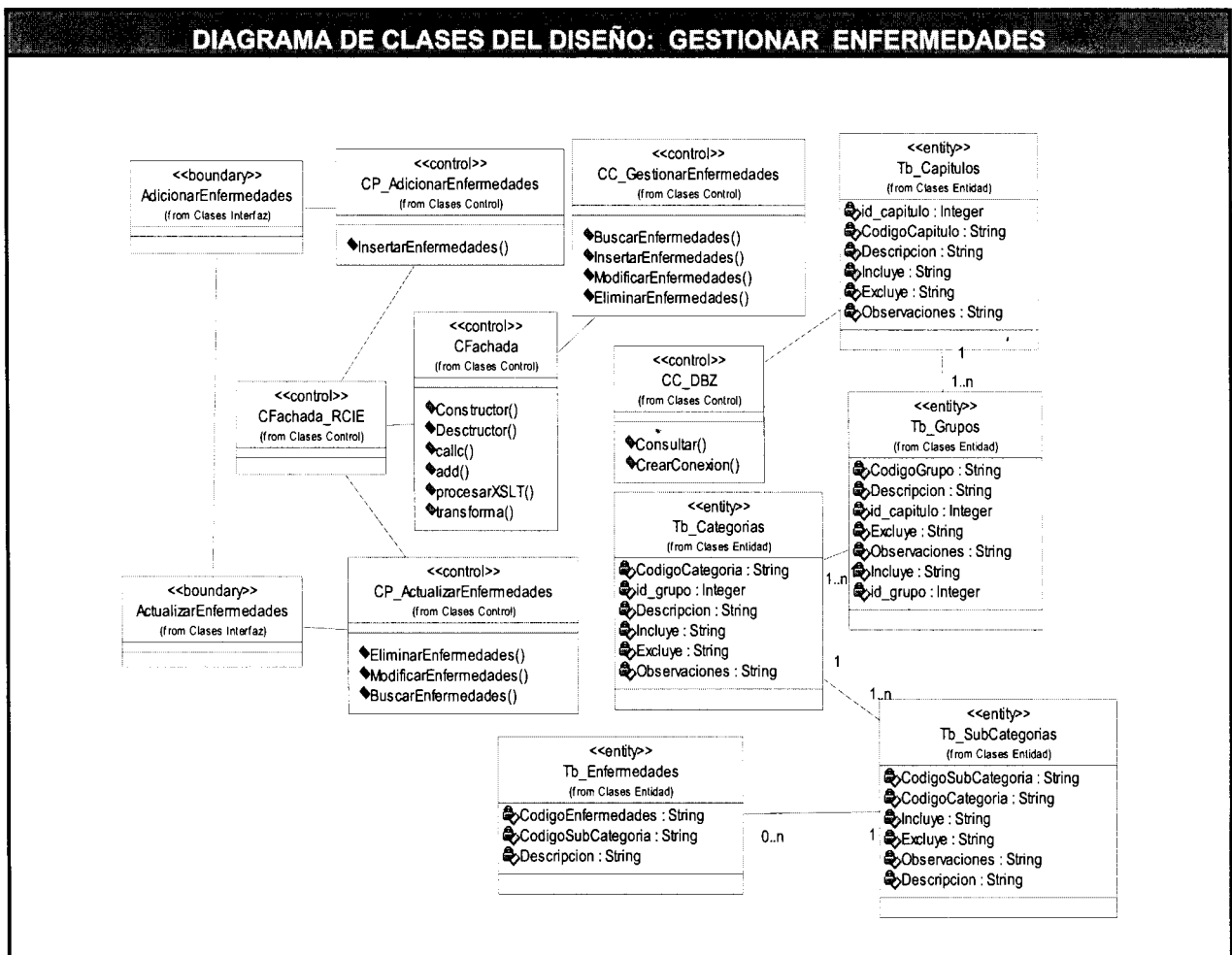


Fig.8 Diagrama de clases Gestionar Enfermedades.

4.2.6 Gestionar Subclasificaciones

Descripción: En este diagrama se visualiza las posibles operaciones y sus clases cuando se desea gestionar los códigos opcionales, el cual tiene como principal objetivo mantener totalmente actualizada la información relacionada con la entidad. El usuario puede seleccionar algunas opciones como adicionar, eliminar o modificar y el sistema será el encargado de visualizar las distintas pantallas para su realización.

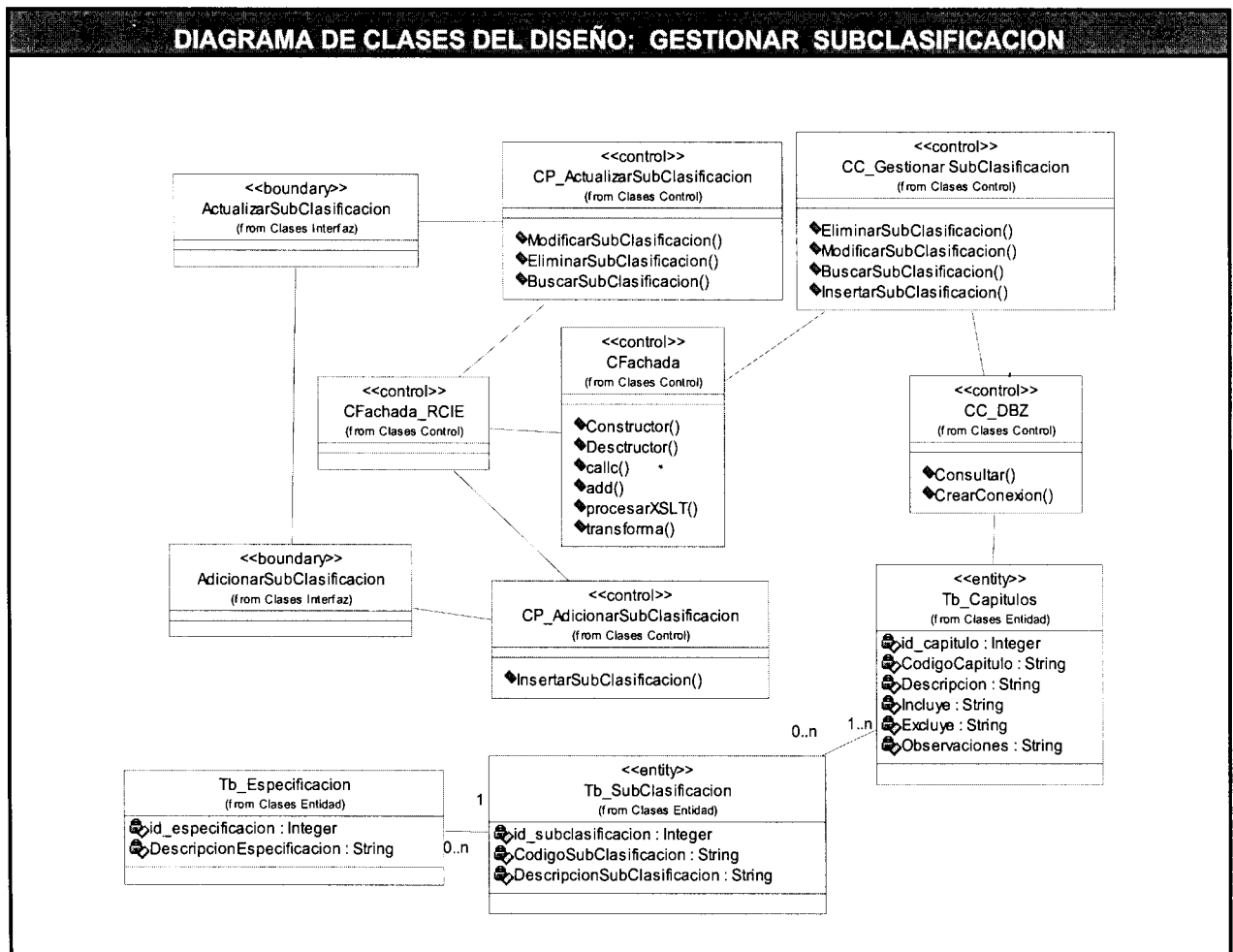


Fig.9 Diagrama de clases Gestionar Subclasificaciones.

4.2.7 Gestionar Definiciones

Descripción: En este diagrama se visualiza las posibles operaciones y sus clases cuando se desea gestionar las Definiciones, el cual tiene como principal objetivo mantener totalmente actualizada la información relacionada con la entidad. El usuario puede seleccionar algunas opciones como adicionar, eliminar o modificar y el sistema será el encargado de visualizar las distintas pantallas para su realización.

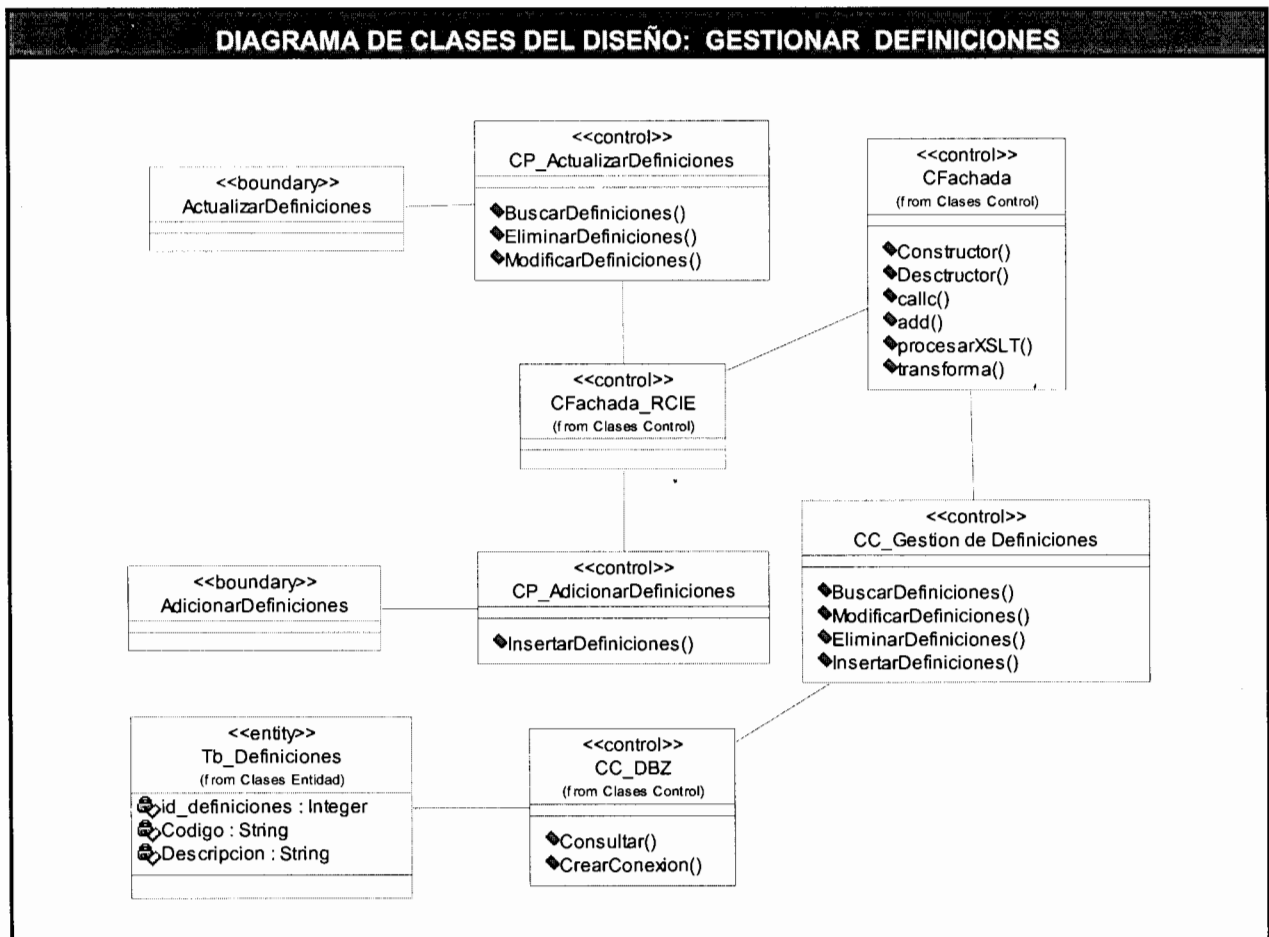


Fig.10 Diagrama de clases Gestionar Definiciones.

4.2.8 Mostrar Información

Descripción: En este diagrama se visualiza las posibles operaciones y sus clases cuando se desea hacer distintas búsquedas dentro de la aplicación, el cual tiene como principal objetivo visualizar toda la información referente a la petición hecha por el usuario. Se podrá seleccionar algunas opciones para realizar las búsquedas como búsqueda estándar o búsqueda avanzada y el sistema será el encargado de visualizar las distintas pantallas para su realización.

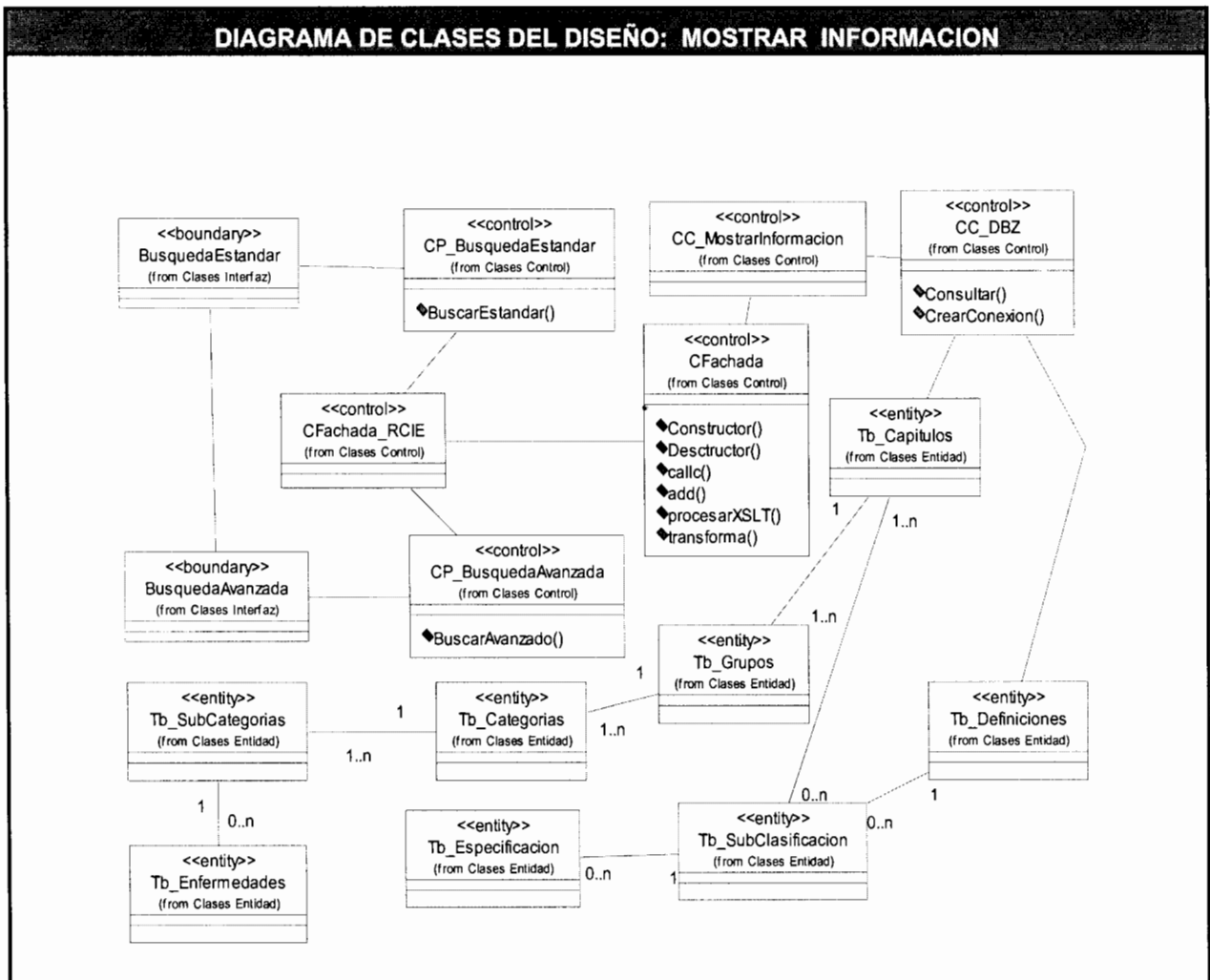


Fig.11 Diagrama de clases Mostrar Información.

4.3 Diseño de la base de datos

4.3.1 Diagrama de clases persistente

Descripción: En este diagrama represento el modelo de las clases persistentes, las cuales formaran parte de mi modelo de datos. El mismo se explica por si solo en la figura que viene a continuación.

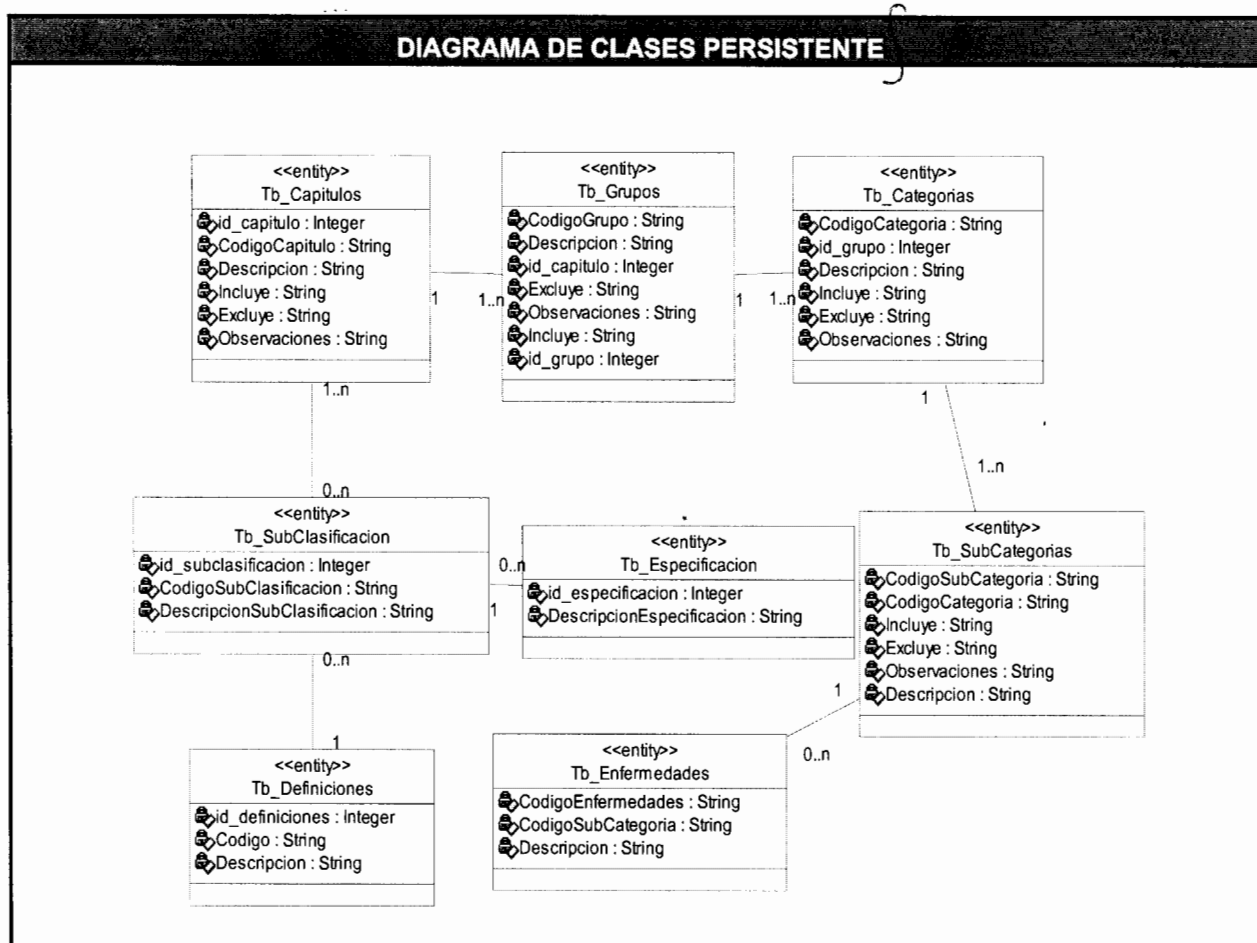


Fig.12 Diagrama de las clases persistentes.

4.3.2 Modelo de datos.

Descripción: El modelo de datos no es más que las clases entidades unido a las nuevas tablas que salen producto de las relaciones, para poder llevar a cabo el modelo físico de la Bases de Datos. A partir del modelo de las clases persistentes de mi sistema.

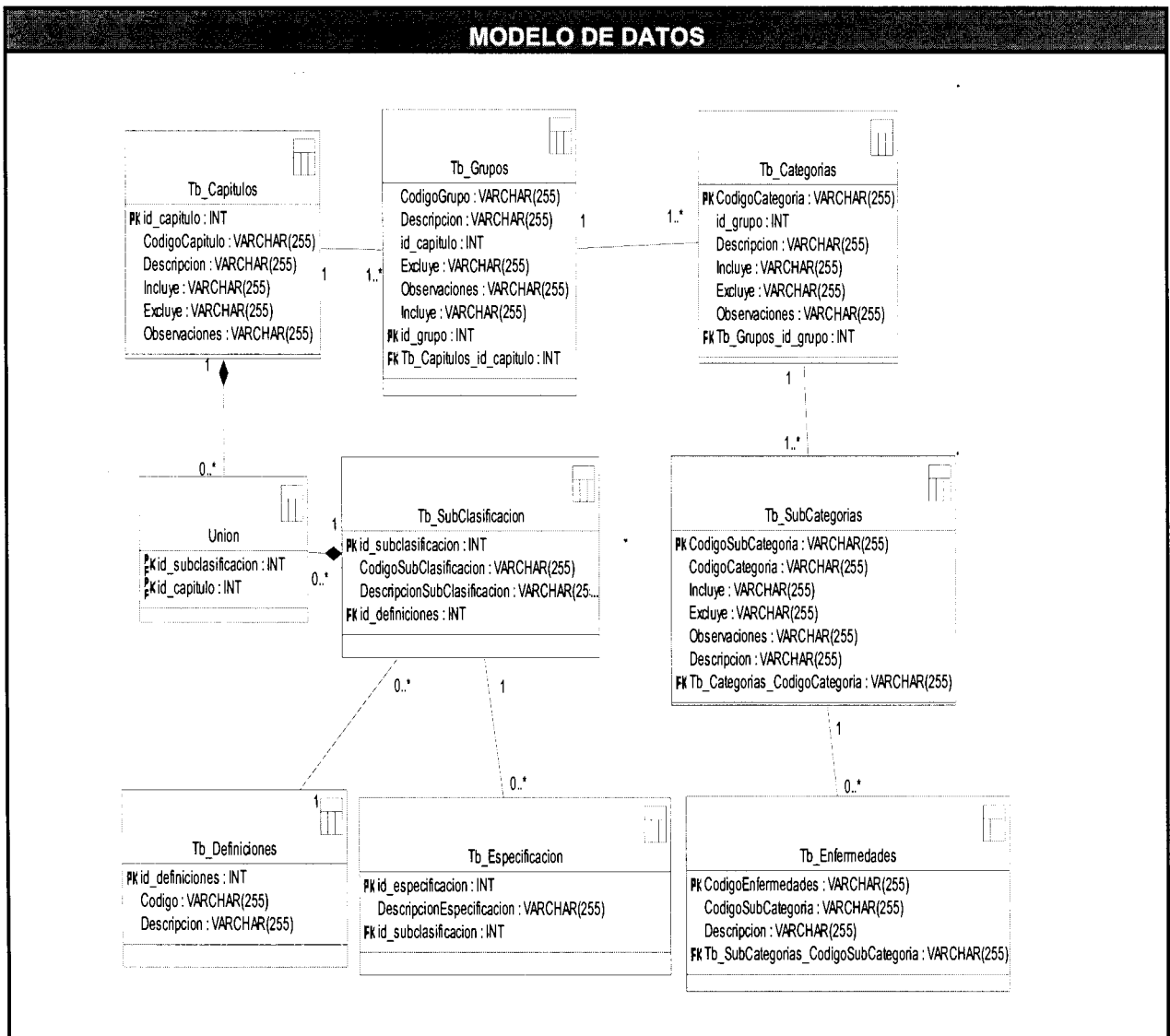


Fig.13 Modelo de datos

4.4 Principio de diseño

4.4.1 Concepción general de la ayuda

Teniendo en cuenta que las ayudas en las aplicaciones Web en general no suelen ser explicaciones detallistas del sistema informático al cual representan, sino que son generalmente simples aclaraciones, informaciones generales de la aplicación o datos de la empresa que le da soporte o realizó el producto se está proponiendo que la ayuda estará accesible como parte del menú en todas las páginas de la aplicación, con el objetivo de que el usuario vea la información que necesita en ese momento. Cada página mostrará como realizar aquellas operaciones que estén relacionadas con la posición donde se encuentre el usuario en dicho momento, además se aportan los conceptos que se manejan en la aplicación, para que el usuario se familiarice con algunas entradas, el entorno de la elaboración de los informes o reportes y otras funcionalidades que se le brindan en el sistema.

La ayuda para todas las aplicaciones del Sistema Integral de Salud estará concebida bajo los principios del soporte técnico en línea, que es una práctica muy utilizada en las aplicaciones Web dinámicas, como la que se está desarrollando y generalmente se implementará con una explicación general de las opciones y con vínculos a sistemas de correo o a otros sitios Web.

Se podrá contar también con un soporte técnico en línea fuera de la aplicación principal para que los usuarios puedan informar acerca de errores que suceden en la aplicación, emitir sugerencias de su funcionalidad o recibir soluciones a las preguntas que de forma “directa” pueden realizar a los administradores y creadores del producto.

Esta forma de ayuda resulta de gran ventaja, ya que contribuye a la resolución de problemas en el software, la gestión de cambios y configuraciones y la actualización y el mantenimiento del producto.

Además se tendrá en cuenta la confección de manuales de usuario y será entregado a los usuarios de cada módulo un manual de usuario en formato digital o en papel, que explicará de forma detallada las principales funcionalidades y opciones que brinda el software.

También está concebida una capacitación técnica directa durante la etapa de implantación del producto, ya que por parte de la empresa productora y bajo acuerdo con los clientes se coordinará la puesta en marcha de cursos de capacitación o entrenamiento en el uso de la aplicación, dirigida a todos los usuarios potenciales antes y durante la implantación oficial de este producto en el Sistema Nacional de Salud.

4.4.2 Tratamiento de excepciones

Para depurar los errores se hará utilizando 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 en caso contrario.

Otros errores en la capa de negocio serán tratados devolviendo un SOAP_FAULT, cuyos elementos FaultCode, FaultString, FaultAutor describiremos 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-RF: 1.5 que indica error 5 en el método 1 del módulo Registro de Fallecidos 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: Formato de entrada no válido para la fecha de cierre estadístico.

- **FaultActor:**

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

- **Detail:**

Este elemento se usa para llevar mensajes de error específicos de aplicaciones, se empleará únicamente en errores cuya resolución depende del Centro de Control, en cualquier otro caso este elemento debe estar vacío.

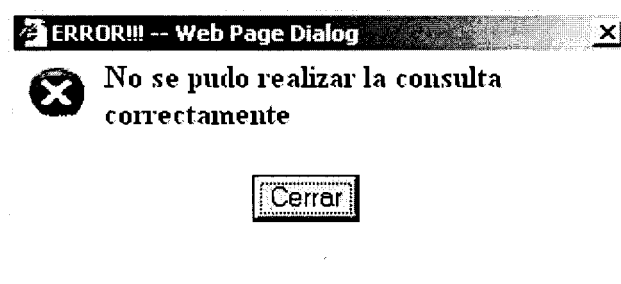


Fig.14 Pagina de Diseño Error

4.4.3 Estándares de la interfaz de la aplicación

Para lograr una mayor eficiencia en el proceso de trabajo, y sobre todo para lograr una coherencia formal entre todos los módulos del sistema, y que sean identificados así como parte de un todo, se han pautado una serie de elementos comunes que facilitarán su reconocimiento y el uso que se haga de ellos.

Se diseñará una Pantalla Inicial global del Sistema Integral de Salud, desde la cual se accederá a los diferentes módulos del RIS, del SIAP y del SIGH. Esta pantalla contará con accesos a los diferentes módulos, informaciones generales, guías de ayuda, sistema de avisos que genera cada registro y enlaces definidos.

Así mismo será diseñada una Pantalla Inicial para cada una de las aplicaciones, que contará con accesos a todas las utilidades, avisos, ayuda y un enlace para regresar a la Pantalla Inicial del Sistema Integral de Salud.

La estructura base de las aplicaciones es la misma para todos los módulos: las pantallas más usadas, los modelos establecidos, las rutas de navegación, las utilidades básicas, la organización de los elementos en pantalla y el diseño de identificadores serán comunes para todos.

Para particularizar el diseño de cada módulo se ha definido entonces una pauta de dos colores básicos para cada uno, con sus degradaciones hacia blanco y negro, así como la diferenciación por logotipo e imagen principal del cabezal, que identificará a cada módulo.

Su diseño está determinado fundamentalmente por el principio de la usabilidad, teniendo en cuenta que no se trata de un sitio web, sino de una aplicación de trabajo donde el diseño tiene como principal propósito facilitar su uso, comprensión y navegación, por encima de ornamentos inútiles, aunque manteniendo pautas estéticas, orgánicas y agradables.

Formalmente, usabilidad se define como la medida en que un producto puede ser usado por determinados usuarios para conseguir objetivos específicos con efectividad, eficiencia y satisfacción, en un contexto de uso especificado.

La resolución óptima para la cual están diseñadas las aplicaciones es de 800 x 600 px. El fondo siempre será blanco y los elementos de pantalla de los colores definidos para cada módulo.

Se ha definido un cabezal pequeño de 65 px de altura, más pequeño que el utilizado en las páginas web, que recomiendan cabezales de hasta 80 px de altura.

El menú principal siempre estará situado en una barra superior horizontal de solo 15 px de altura. No existirá barra vertical de menú situada a la izquierda de la página (como usualmente se hace) para ampliar el espacio de trabajo, pues estará reservado lo más amplio posible para la inserción de grandes tablas y formularios que constituyen la base fundamental de estas aplicaciones.

El logo siempre estará ubicado en el extremo superior izquierdo de la página, es una imagen que cuenta con un ancho de 270 px y se corresponde con el nombre de cada módulo. Estará constituido por un juego tipográfico en *Frankling Gothic Medium*, y en el caso de las aplicaciones propias del Proyecto APS, estando especificado dentro del logo como una especie de genérico.

Bajo el logo existirá una barra de ubicación dentro del sitio, funcionando como hipervínculo, que servirá como referencia para saber donde se encuentra el usuario o para acceder rápidamente a cualquiera de los niveles superiores de navegación dentro de los que se encuentra. Además se encontrará destacado dentro del menú principal (con un

destaque en el color secundario) en cual de los elementos del menú se encuentra el usuario en ese momento.

La tipografía será siempre Tahoma, por su amplia legibilidad y por las facilidades conocidas que brinda para la lectura digital. El menú principal será a 7 ptos y los submenús a 6 ptos. Los demás puntajes se definirían en dependencia de las necesidades puntuales de cada pantalla.

El espacio de trabajo comienza 33 px por debajo del menú. El espacio intermedio que queda es también con fondo blanco y está reservado para el texto de ubicación dentro del sitio (justificado a la izquierda) y para ubicar los botones propios de la pantalla (justificados a la derecha). Estos se organizarán en una o dos filas, de hasta cuatro botones (13 x 72 px) cada una. Los botones se corresponden también con los colores pautados.

Entre los elementos comunes del menú principal se encuentran *Inicio* para regresar a la página inicial del módulo, *Salir* para desconectarse del sistema, y *Otros Módulos* para facilitar los enlaces a otros módulos necesarios. Son también comunes a casi todos los botones del menú principal *Configurar* para la configuración de codificadores, *Cierre* para la realización de cierre estadístico y *Reportes* para generar reportes de actividades u operaciones.

Es común para todos los módulos el diseño de una serie de ventanas, en las que solo cambiarían los colores, en dependencia de cada uno. Son estas las ventanas de precaución, error, validación de datos, etc.

En cuanto a los elementos de diseño del interior de las pantallas, es decir, de las tablas, formularios, etc., se definen los edit que se utilicen con una altura de 16 px y la separación entre estos y entre ellos y los bordes de tablas será de 8 px. Será de 8 px la separación entre el texto y el edit. Los textos de estos campos serán justificados siempre a la derecha, es decir, justificados a 8 ptos de cada edit.

En el caso de tablas generadas por búsquedas, que ordenan una serie de elementos, y necesiten selección, se harán a través de checkboxes justificados a la izquierda de la tabla.

Siempre habrá un checkbox en la fila de título, también a la izquierda, que facilite seleccionar todos. Es necesario destacar que estas tablas pueden tener una cantidad grande de líneas generadas por la búsqueda, por lo que debe quedar pautado que hasta 25 resultados la tabla funcione con scroll, pero más de esta cantidad será entonces por paginado, al estilo de Google, con 25 resultados por página.

Existen detalles que serán definidos particularmente en cada uno de los módulos, ya que satisfacen a necesidades específicas de los mismos.

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

A continuación se muestra una representación gráfica de las pautas generales de la organización de elementos en pantalla.

4.4.4 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`.

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.

Los arreglos empezarán con el identificador array y las palabras no se separarán con el carácter “_”. Ejemplo: Arrayidtipoenfermedad.

Las estructuras se identificarán poniendo al final del nombre struct. Ejemplo: paginadostruct.

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: ListarTotalPersonalSalud.

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 “/* */”.

Se usará una indentación en el código de cuatro espacios para facilitar la lectura de éste. Las llaves se usarán poniendo la llave inicial en una línea para ella sola, y en su respectiva columna la llave final también en una línea.

El 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 interfase de administración se usará el español para esclarecer los objetivos de cada método o script a utilizar.

Para lograr que las comparaciones sean seguras, se colocarán siempre los valores constantes a la izquierda de la comparación “if (6 == \$variable)”, con esto garantizará la generación de un error cuando por error escriba '=' y no '=='. 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.

Los controles seguirán el siguiente tratamiento:

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

Tabla 7 – 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 tienen que nombrar la base de datos poniendo el identificador del proyecto “APS” seguido del carácter “_” y del nombre del módulo. Ejemplo: APS_RF, APS_PN.

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

4.5 Modelo de despliegue

4.5.1 Modelo de Componente

Descripción: aquí se muestra la organización del componente RCIE, el cual se encuentra dividido en tres capas y a su vez este se relaciona con los componentes externos a mi aplicación que se encuentran situados en la misma plataforma mediante PlaSer.

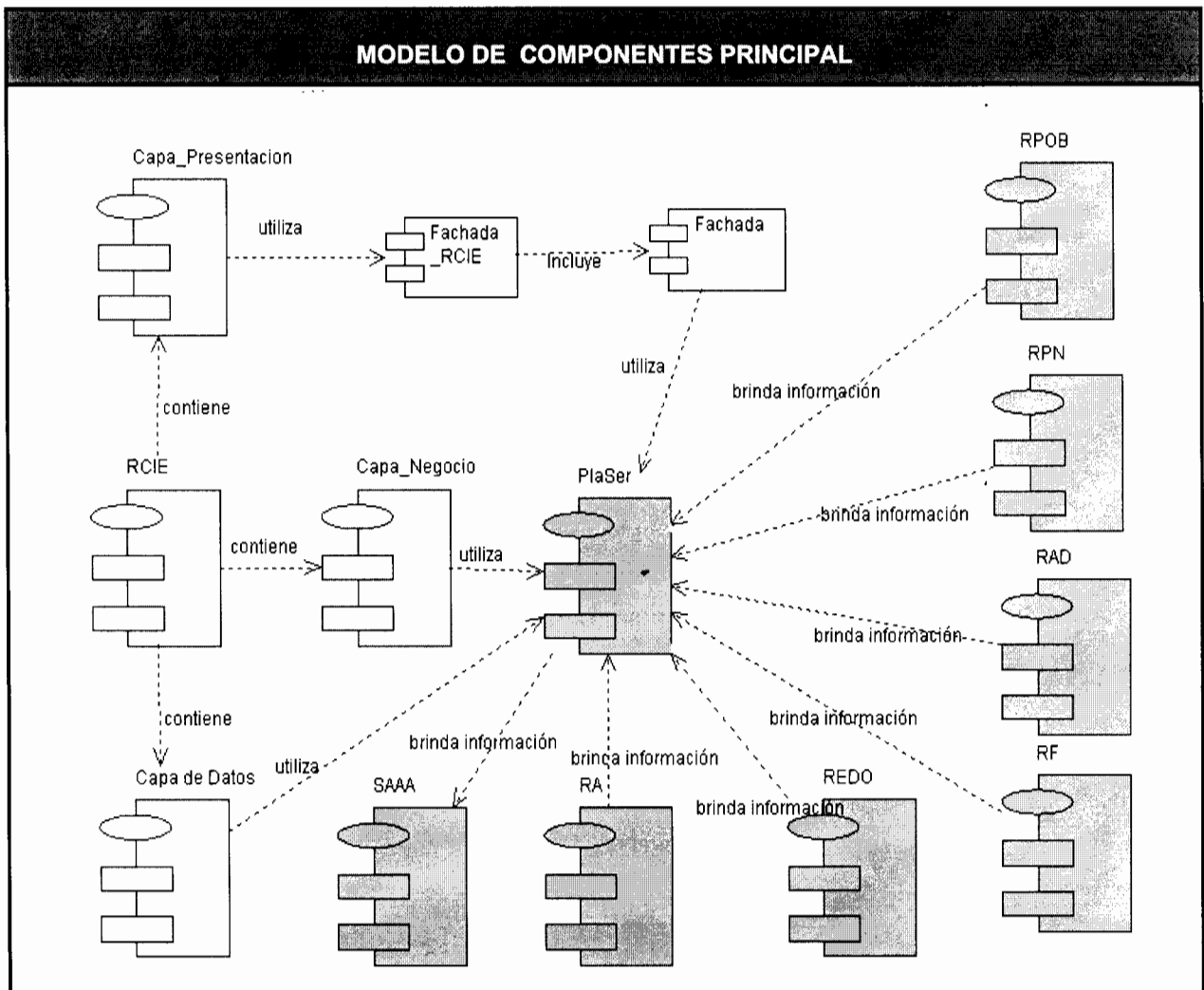


Fig.15 Diagrama de componentes

4.5.2 Modelo de Despliegue

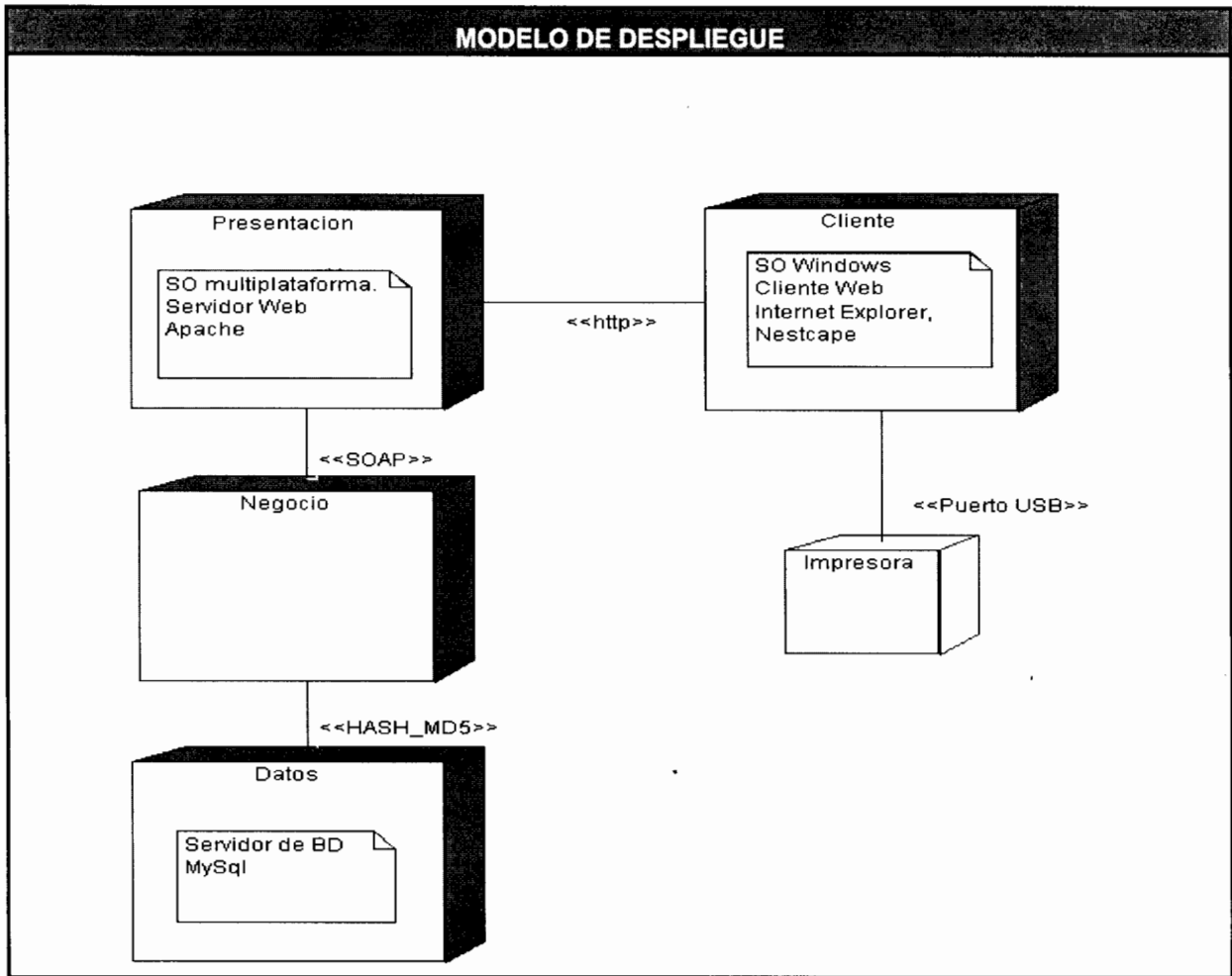


Fig.16 Diagrama de despliegue

4.6 Conclusiones

Después de realizadas cinco iteraciones a través de los flujos de trabajo que propone RUP, se ha logrado modelar el presente sistema, cualquier programador o estudiante con los mínimos conocimientos de ingeniería de software puede utilizar los diagramas antes expuestos como referencia para realizar una operación de mantenimiento, mejora o simplemente como material de estudio, ahora solo queda implementar todo lo expuesto en el transcurso del documento para obtener el producto final.

CAPÍTULO 5 ESTUDIO DE FACTIBILIDAD

Hoy en día el Software es el elemento más caro de la mayoría de los sistemas informáticos, un gran error en la estimación del costo puede ser lo que marque la diferencia entre beneficios y pérdidas.

Existen diferentes modelos para calcular la estimación del costo de un sistema, actualmente el más difundido es el Modelo Constructivo de Costes (COCOMO). Es uno de los modelos más documentados en la actualidad y es muy fácil de utilizar, por todo lo anterior lo utilizaremos en nuestro trabajo.

Fue desarrollado por B. W. Boehm a finales de los 70 y comienzos de los 80, pertenece a la categoría de modelos de estimación basados en estimaciones matemáticas. Está orientado a la magnitud del producto final, midiendo el tamaño del proyecto en líneas de código principalmente. [25]

Entre sus ventajas se encuentran:

- ✓ La utilización de unidades físicas, tales como las líneas de código para medir la magnitud del producto.
- ✓ La fácil realización, ya que las unidades de medida son tangibles.
- ✓ La flexibilidad y la utilización en cualquier tipo de proyecto.
- ✓ Los resultados obtenidos son proporcionales a las tareas técnicas.

5.1 Planificación

Entradas Externas	Ficheros	Elementos de datos	Clasificación
Insertar datos de los Capítulos	1	6	Simple
Modificar datos de los Capítulos	1	6	Simple

Eliminar datos de los Capítulos	1	6	Simple
Insertar datos de los Grupos	1	7	Simple
Modificar datos de los Grupos	1	7	Simple
Eliminar datos de los Grupos.	1	7	Simple
Insertar datos de las Categorías	1	8	Simple
Modificar datos de las Categorías	1	8	Simple
Eliminar datos de las Categorías	1	8	Simple
Insertar datos de las subcategorías	1	9	Simple
Modificar datos de las subcategorías	1	9	Simple
Eliminar datos de las subcategorías	1	9	Simple
Insertar datos de las enfermedades	1	8	Simple
Modificar datos de las enfermedades	1	8	Simple
Eliminar datos de las enfermedades	1	8	Simple
Insertar datos Código Opcional	3	8	Media
Modificar datos Código Opcional	3	8	Media
Eliminar datos Código Opcional	3	8	Media
Insertar datos de las Definiciones	1	3	Simple
Eliminar datos de las Definiciones	1	3	Simple
Modificar datos de las Definiciones	1	3	Simple
TOTAL	Simple: 18, Media: 1, Compleja: 0.		

Tabla 5.1 Entradas Externas.

Salidas Externas	Ficheros	Elementos de datos	Clasificación
TOTAL	Simple: 0, Media: 0, Compleja: 0.		

Tabla 5.2 Salidas Externas.

Peticiones	Ficheros	Elementos de datos	Clasificación
Buscar Capítulos.	1	8	Simple
Buscar Grupos	1	9	Simple
Buscar Categorías	1	10	Simple
Buscar Subcategorías	1	11	Simple

Buscar Enfermedades	1	12	Simple
Buscar Subclasificaciones	3	11	Medio
Buscar Definiciones	1	5	Simple
Búsqueda estándar	9	5	Complejo
Búsqueda Ampliada.	9	10	Complejo
Búsqueda por código	9	28	Complejo
TOTAL	Simple: 6, Media: 1, Compleja: 3.		

Tabla 5.3 Peticiones.

Ficheros Lógicos Internos	Records	Elementos de datos	Clasificación
Capítulos	1	6	Simple
Grupos	1	7	Simple
Categorías	1	6	Simple
Subcategorías	1	6	Simple
Enfermedades	1	3	Simple
Subclasificación	3	10	Simple
Definiciones	1	3	Simple
TOTAL	Simple: 21, Media: 0, Compleja: 0.		

Tabla 5.4 Ficheros Lógicos Internos.

Interfaz Externa	Records	Elementos de datos	Clasificación
-	-	-	-
TOTAL	Simple: 0, Media: 0, Compleja: 0.		

Tabla 5.5 Ficheros de Interfaz Externa.

Elementos	Simples		Medios		Complejos		Subtotal Pts de función
	No.	X Peso	No.	X Peso	No.	X Peso	
Ficheros lógicos internos.	7	*7=49					49
Ficheros de interfaces externas							0
Entradas externas.	18	*3=54	3	*4=12			66
Salidas externas.							0
Peticiones.	6	*3=18	1	*4=4	3	*6=18	40
TOTAL							155

Tabla 5.6 Puntos de Función sin ajustar.

Cálculo de las instrucciones fuentes.

El cálculo de las instrucciones fuentes, según Cocomo II, se basa en la cantidad de instrucciones por punto de función que genera el lenguaje de programación empleado.

Características	Valor		
Puntos de función desajustados	155		
Lenguaje	PHP	JavaScript	SQL
% de utilización en la aplicación	85% (≈135)	10% (≈16)	5% (≈8)
Instrucciones fuentes por puntos de función	60	56	39
Instrucciones fuentes	8,100	0,900	0,400
Total Instrucciones fuentes	9.400		

Tabla 5.7 Instrucciones fuentes

5.2 Cálculo del esfuerzo, tiempo de desarrollo, cantidad de hombres y costo

Multiplicadores de esfuerzo:

Multip.	Descripción	Valor
RCPX	La complejidad del producto es media.	1
RUSE	Se implementa código reutilizable para su aprovechamiento en el proyecto.	1
PDIF	La plataforma es estable. Requerimientos bajos de almacenamiento y tiempo de ejecución.	0.87
PERS	La capacidad de los especialistas (analistas-programadores) es alta. La continuidad del personal es alta.	0.83
PREX	El equipo tiene poco dominio y conocimiento del lenguaje de programación, plataforma y herramientas de desarrollo utilizados. No ha desarrollado aplicaciones similares, casi ninguna experiencia.	1.33
FCIL	Se utilizan herramientas e instrumentos de programación modernos.	1
SCED	Los requerimientos de calendario de desarrollo son bajos.	1.14
		1.03

Tabla 5.8 Definición de los Multiplicadores de Esfuerzo (MEj)

7

$$EM = \prod_{i=1}^7 EM_i = RCPX * RUSE * PDIF * PERS * PREX * FCIL * SCED = 1.03$$

Factores de Escala

Factor	Descripción	Valor
PREC	Bastante parecido	2.48
FLEX	El sistema cuenta con alguna flexibilidad en relación con las especificaciones de los requerimientos preestablecidos y a las especificaciones de interfaz externa.	3.04
TEAM	Interacciones principalmente cooperativas. Mediana experiencia previa operando	2.19
RESL	La arquitectura es sólida y los riesgos generalmente se mitigan. Poca incertidumbre, riesgos no son críticos.	1.41
PMAT	Relación con el proceso de madurez del software. Nivel 3.	3.12
		5 $\sum_{i=1}^5 SF_i$
		12.24

Tabla 5.9 Definición de los valores de los Factores de Escala (SF_i)

$$SF = \sum SF_i = PREC + FLEX + RESL + TEAM + PMAT = 12.24$$

Valores calibrados:

$$A=2.94; B=0.91; C=3.67; D=0.24$$

$$E = B + 0.01 * \sum SF_i = 0.91 + 0.01 * 12.24 = 1.03$$

$$F = D + 0.2 * (E - B) = 0.24 + 0.2 * (1.03 - 0.91) = 0.26$$

Cálculo del esfuerzo (PM)

$$PM = A * (MSLOC)^E * \Pi Em_i = 2.94 * (9.400)^{1.03} * 1.09 = 33 \text{ Hombres/Mes.}$$

Cálculo del tiempo de desarrollo.

$$TDEV = C * PM^F = 3.67 * (33)^{0.26} = 9.109 \approx \mathbf{9 \text{ meses (Estimado)}}$$

Cálculo de la cantidad de hombres.

$$CH = PM / TDEV = 33 / 9 = \mathbf{4 \text{ hombres}}$$

Como el equipo de trabajo está formado realmente por 3 personas, se recalcula el tiempo de desarrollo para la cantidad real de hombres.

$$CH^* = 3 \text{ hombres.}$$

$$TEDV = PM / CH^* = 33 / 3 = \mathbf{10 \text{ meses.}}$$

Cálculo del costo.

Asumiendo como salario promedio mensual (SP) \$225.00

$$CHM = CH * SP = 3 * \$225.00 = \$675.00$$

$$\text{Costo} = CHM * PM = \$675.00 * 33 = \$ 22 275.00$$

- Cálculos

Cálculo de:	Valor
Esfuerzo (PM: Hombres - mes)	33 Hombres/Mes
Tiempo de Desarrollo (meses)	10 Meses

Cantidad de Hombres	3
Costo	\$22 275.00
Salario medio	\$ 225.00

Tabla 5.10 Calculo del esfuerzo, tiempo de desarrollo, cantidad de hombres y costo.

5.3 Beneficios tangibles e intangibles

Los beneficios que se obtendrán con el desarrollo del sistema propuesto son fundamentalmente intangibles, pues este sistema formará parte de conjunto de módulos que facilitarán considerablemente el trabajo en el sector de la salud en nuestro país.

La solución informática propuesta, estandarizará la forma de ^{clasificar} diagnosticar las distintas enfermedades o problemas de salud, con el fin de poder almacenar los diversos diagnósticos médicos que permitan realizar estudios estadísticos futuros en los diversos niveles del SNS, posibilitando la toma de decisiones. También permitirá realizar los diagnósticos médicos con una mayor exactitud, lo cual ayudaría considerablemente a todo el personal de la salud que haga uso de estos codificadores.

5.4 Análisis de costos y beneficios

El desarrollo de todo producto informático va asociado a un costo, el justificarlo depende de los beneficios tangibles e intangibles que trae consigo.

La utilización de este nuevo sistema apoyará íntegramente a otros sistemas en sus funciones. La nueva herramienta automatizada brinda numerosas facilidades para obtener reportes no sólo con fines estadísticos, sino del desarrollo del cumplimiento de las actividades según sus responsables en cada momento.

Por lo cual podemos llegar a ^{la} conclusión que el sistema es factible llevarlo a la etapa de implementación con un costo aproximado de \$22 275.00.

CONCLUSIONES

- Al concluir el presente trabajo se concluye que los objetivos general y específicos fueron cumplidos cabalmente.
- Se realizó el análisis de los aspectos generales y específicos, así como los conceptos asociados al dominio del Ministerio de Salud Pública.
- Se realizó el proceso de gestión de la información relacionada con las enfermedades y problemas de la salud.
- Se realizaron los flujos de trabajo, análisis y diseño del módulo Registro de la Clasificación Internacional Enfermedades y Problemas relacionados con la Salud para el RIS, utilizando el proceso de desarrollo RUP y el UML como herramienta de modelación.
- Se modelaron los procesos de interacción del módulo que conformarán el funcionamiento del mismo.
- Se realizó un estudio de las herramientas informáticas adecuadas para realizar una futura implementación del sistema propuesto e integrarlo al Sistema Integral de Salud que responderá a la Informatización del Sistema Nacional de Salud.

RECOMENDACIONES

Recomendamos para próximas etapas del desarrollo del Proyecto APS:

Que la Empresa SOFTEL perteneciente al Ministerio de la Informática y las Comunicaciones (MIC), en conjunto con la Universidad de las Ciencias Informáticas (UCI), realice el estudio de los aspectos generales y específicos descritos en el trabajo para entender el dominio del problema.

Que se encuentre un apoyo en el presente documento para entender el proceso de gestión de la información relacionada con las enfermedades y los problemas de salud para tomar dicho conocimiento como base para el desarrollo futuro de los flujos de trabajo implementación, prueba y mantenimiento.

Que el Equipo de Trabajo del Proyecto APS se apoye en la ingeniería de software desarrollada durante el proyecto, para realizar en un futuro un producto con la calidad requerida.

Que se estudien los procesos de interacción reflejados y descritos en el trabajo para realizar un producto eficiente funcionalmente y gestione la información de forma rápida, segura y flexible.

Que se estudien todas las herramientas informáticas propuestas en el proyecto para realizar la futura implementación del módulo.

REFERENCIA BIBLIOGRAFICA

1. Marín, ME. *Pilares y debilidades del proceso de informatización de la Atención Primaria de Salud (APS)*. Inédito. Taller de Informática en Salud. Universidad de las Ciencias Informáticas. Ciudad de La Habana: 29 y 30 de abril 2004.
2. Granma, *Discurso pronunciado por el Presidente de la República de Cuba, Fidel Castro Ruz, en el acto conmemorativo del aniversario 40 del Instituto de Ciencias Básicas y Preclínicas Victoria de Girón*, 17 de octubre de 2002 •✓
3. De la Osa, José A. *La semilla del desarrollo de la salud pública en Cuba*.
4. Escuela Nacional de Salud Pública, *El cuidado de la salud en Cuba*. Ministerio de Salud Pública, 2003.
5. Granma, *Discurso pronunciado por el Presidente de la República de Cuba, Fidel Castro Ruz, en la Tercera Graduación del Contingente del Instituto Superior de Ciencias Médicas de la Habana*. Teatro "Carlos Marx". Ciudad de la Habana. 27 de agosto de 1990.
6. Granma. *Discurso pronunciado por Fidel Castro Ruz en el acto de inauguración de obras del extraordinario programa de salud*. Teatro Astral. 7 de abril del 2003).
7. Granma, *Discurso pronunciado en la Clausura del VI Seminario Internacional de Atención Primaria, Ciudad de la Habana*, 28 de noviembre de 1997.
8. Ramírez Márquez, Abelardo *El Sistema Nacional de Salud de Cuba*. / Pastor Castell-Florit Serrate; Guillermo Mesa. ENSAP, 2003.
9. Lemus, Elia Rosa, *Atención Primaria de Salud y Medicina Familiar.*, Eugenio Radamés Borroto Cruz; Ramón Aneiros-Riba. Atención Primaria de Salud, Medicina Familiar y Educación Médica, Biblioteca de Medicina Volumen XXXIV, La Paz, 1998.
10. Martínez Calvo, Silvia, *Análisis de Situación de Salud*. / Héctor Gómez de Haz. Escuela Nacional de Salud Pública, 2003.
11. *Internet*, <http://es.wikipedia.org/wiki/Internet> (20/02/2005)
12. *What is Service-Oriented Architecture?*,
<http://webservices.xml.com/pub/a/ws/2003/09/30/soa.html> (10/3/2005)

13. *SOAP: The Simple Object Access Protocol*,
<http://www.microsoft.com/mind/0100/soap/soap.asp> (10/3/2005)
14. *Simple Object Access Protocol (SOAP)*,
<http://www.desarrolloweb.com/articulos/1557.php?manual=54> (10/3/2005)
15. *Lenguaje de descripción de servicios Web (WSDL)*,
<http://www.microsoft.com/spanish/msdn/articulos/archivo/090201/voices/wsdl.asp>
(12/3/2005)
16. *Diseño de Aplicaciones Three Tier*, <http://www.fpress.com/revista/Num9711/Nov97.htm>
(12/3/2005)
17. *Apache*. <http://es.wikipedia.org/wiki/Apache> (3/4/2005)
18. *Introducción a la Arquitectura de Software*,
http://www.microsoft.com/spanish/msdn/arquitectura/roadmap_arq/intro.asp#17
(12/3/2005)
19. *Sistemas distribuidos*, www.monografias.com/trabajos16/sistemas-distribuidos/sistemas-distribuidos.shtml (12/3/2005)
20. *Understanding Single Sign-on*,
http://www.intranetjournal.com/articles/200205/se_05_28_02a.html, (3/4/2005)
21. *Introduction to Perl*, www.cclabs.missouri.edu/things/instruction/perl/perlcourse.html,
(3/4/2005)
22. *Introduction to Perl*,
http://www.tecnociencia.es/mediawiki/index.php/Active_Server_Pages (19/04/2005)
23. *Introducción a php*, <http://www.ciberteca.net/webmaster/php> (19/04/2005)
24. *JSP Tutorial*, <http://www.jsptut.com/Index.html> (19/04/2005)
25. *Manual de JavaScript*, <http://www.redestb.es/soporte/aula/jScript> (02/03/2005)
26. *XSLT*, <http://es.wikipedia.org/wiki/XSLT>, (02/03/2005)
27. *SQL Reference*, <https://aurora.vcu.edu/db2help/db2s0/frm3toc.htm> (3/4/2005)
28. *Manual de referencia de MySQL* <http://www.mysql.com>(11/04/2005)

29. *PostgreSQL*, <http://es.wikipedia.org/wiki/PostgresSQL> (11/04/2005)
30. Jacobson, Ivar y Booch, Grady y Rumbaugh, James. *El proceso unificado de software*. Primera edición. Pearson Educación, S.A. 2000
31. Conallen, J. – “*Modeling web application architectures with UML*” – 2000, Communications of the ACM (volume 42, number 10)
32. *Rational Rose: Procedimientos básicos para desarrollar un proyecto*, <http://www.vico.org/TallerRationalRose.pdf> (11/04/2005).

BIBLIOGRAFÍA

Arman, C. "Applying UML and Patterns. An Introduction to Object-Oriented Analysis and Design". Prentice-Hall, Inc. 1998.

Cabrera Hernández, Mirna. García Morales, Juan Carlos. Sánchez Arias, Sandra. Ávila García, Mario. E. Marín, Miguel. "Especificación Preliminar de Requisitos". Empresa de producción de software SOFTEL, Ciudad Habana, 2004.

Chávez, Francisco. "Sistemas de información para ejecutivos".
<http://www.tress.com.mx/boletin/marzo2005/sistemas.htm> (12/06/05).

COCOMO II. 23 septiembre 2002.
http://sunset.usc.edu/research/COCOMOII/cocomo_main.html (5/6/2005)

De la Fuente Moya, Antonio. *Cocoma v2, "Modelo de estimación de costes para proyectos software"*. Universidad de Castilla-La Mancha, 1999.

Dirección Nacional de Estadística MINSAP. "Anuario estadístico de Salud". Ciudad Habana, 2002.

Dirección Nacional de Registros Médicos y Estadísticas de Salud. "La Clasificación Internacional de Enfermedades y Problemas relacionados con la salud (CIE)".
<http://uvirtual.sld.cu/wssc/slides/slides.php3?c=102&d=1> (8/02/05).

Doug, Tidwell. "introducción al XML". <http://www.ibm.com/developerWorks> (5/12/04).

Hernández González, Anaisa. "Asignatura Optativa Ingeniería de Software". Diseño, Modelo de Diseño.doc.

Hernández González, Anaisa. Ampuero, Andrés, Margarita. "Asignatura Optativa Ingeniería de Software". Análisis.

Hernández González, Anaisa. Ampuero, Andrés. Margarita. Lau Fernández, Rogelio. López Valdés, Irene. "Asignatura Optativa Ingeniería de Software". Modelamiento del negocio, Modelo negocio.

<http://www.proactiva-calidad.com/java/patrones/mvc.html>. (5/6/2005).

INFOMED. "Aspectos Generales". http://www.sld.cu/sistema_de_salud/estructura.html
(12/04/05).

INFOMED. "Estrategias y programas". http://www.sld.cu/sistema_de_salud/estructura.html
(12/04/05).

INFOMED. "Estructura del Ministerio". http://www.sld.cu/sistema_de_salud/estructura.html
(12/04/05).

Jacobson, I.; Booch, G. y Rumbaugh, J.. *"El Proceso Unificado de Desarrollo de software"*.
2000. Addison-Wesley.

Jacobson, Ivar y Booch, Grady y Rumbaugh, James. *El proceso unificado de software*.
Primera edición. Pearson Educación, S.A. 2000.

Lago, Ramiro. *Patrón Modelo-Vista-Controlador* ,

Larman, Craig. *UML y Patrones. Introducción al análisis y diseño orientado a objetos*.
Prentice Hall Hispanoamérica, México, 1999.

Martínez, Rafael. *"Manual de PHP"*. <http://www.php.net/manual/es/print/index.php>
(9/6/2005)

Metodología MÉTRICA Versión 3. "Técnicas y Practicas utilizadas en los procesos
principales y en el proceso de Gestión de *Proyectos*".

Paloma Cáceres, Esperanza Marcos. "Procesos Ágiles para el Desarrollo de Aplicaciones
Web". <http://www.dlsi.ua.es/Webe01/articulos/s112.pdf> (4/06/05).

Pressman, R. *"Software Engineering. A Practitioner's Approach"*. Fourth Edition. McGraw –
Hill. USA, 1999.

Pressman, R. *"Software Engineering. A Practitioner's Approach"*. Fourth Edition. McGraw –
Hill. USA, 1999.

Rodríguez, Gutiérrez. "*Arquitectura de Aplicaciones de tres capas*".

<http://dotnetjunkies.com/WebLog/desarrollonet/archive/2004/06/17/16855.aspx> (7/6/2005)

Rosalía. "Reglamento Softel Trabajo RUP". Plantilla. Softel, 2005.

RUMBAUGH J, Jacobson I., Booch G. El Lenguaje Unificado de Modelado. Manual de Referencia. 2000 Addison Wesley.

Santana, Pedro. "Implementando servicios Web con PHP".

<http://www.pecesama.net/php/ws.php> (20/03/05).

Softel."Descripción por capas de la arquitectura para APS". Sep, 2003.

Tutorial de PHP y MySQL, http://es.tldp.org/Manuales-LuCAS/manual_PHP/manual_PHP/

GLOSARIO DE TÉRMINOS.

CodigoCapitulo: Atributo de la entidad Tb_Capitulos, en el cual almacena el rango de categorías que abarca un Capitulo.

CodigoGrupo: Atributo de la entidad Tb_Grupos, en el cual almacena el rango de categorías que abarca un Grupo.

CodigoCategoria: Atributo de la entidad Tb_Categorias, en el cual almacena el código de una categoría.

CodigoSubcategorias: Atributo de la entidad Tb_SubCategorias, en el cual almacena el código de una subcategoría.

CodigoEnfermedades: Atributo de la entidad Tb_Enfermedades, en el cual almacena el código de una enfermedad o problema de salud

CodigoClasificacion: Atributo de la entidad Tb_SubClasificacion, en el cual almacena el código de una clasificación dada para una codificación adicional de una enfermedad o problema de salud

CodigoDefinicion: Atributo de la entidad Tb_Definiciones, en el cual almacena el código de una definición para ayudar en la codificación de una enfermedad o problema de salud

Descripcion: atributos de algunas entidades para dar el nombre o funcionabilidad de cada uno de los conceptos dentro del CIE.

Incluye: Atributo en el cual se Incluye las otras enfermedades o problemas de salud que se debe tener en cuenta a la hora de codificar cualquier padecimiento medico.

Excluye: Atributo en el cual se Excluye las otras enfermedades o problemas de salud que se debe tener en cuenta a la hora de codificar cualquier padecimiento medico.

Observaciones: Atributo que se debe tener en cuenta a la hora de hacer cualquier codificación de una enfermedad o problema de salud, sirve para dejar escrita ciertas informaciones aclaratorias, las cuales serán útiles cuando se este en el proceso de diagnosticar cualquier padecimiento medico.

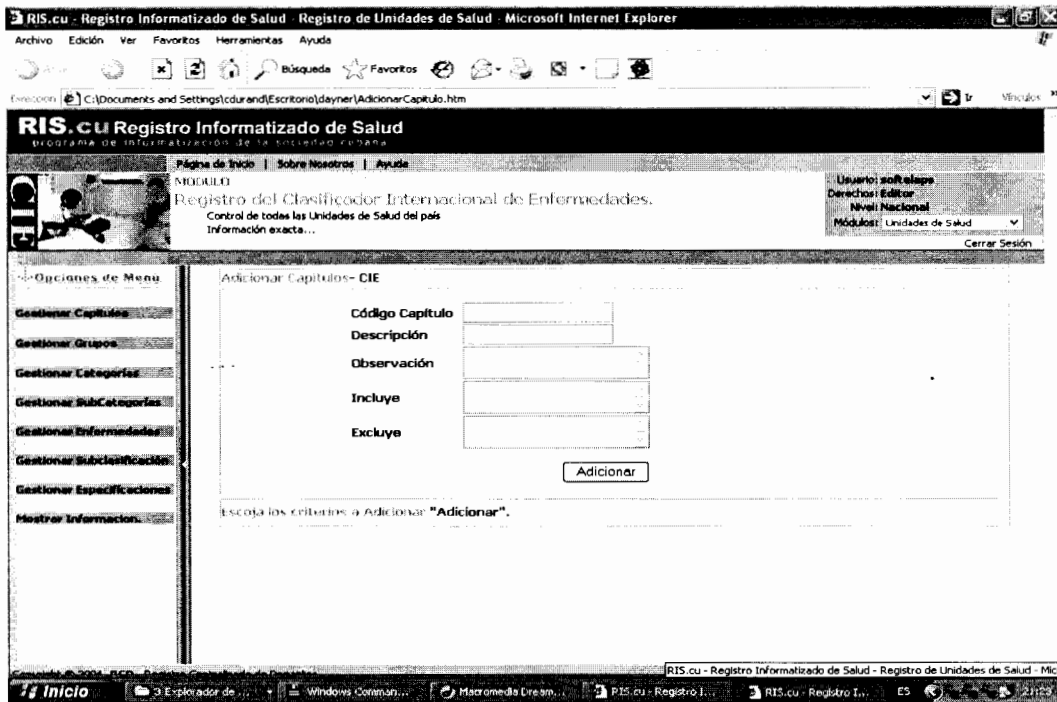


Fig.17 Gestionar Capítulos. Adicionar Capítulos

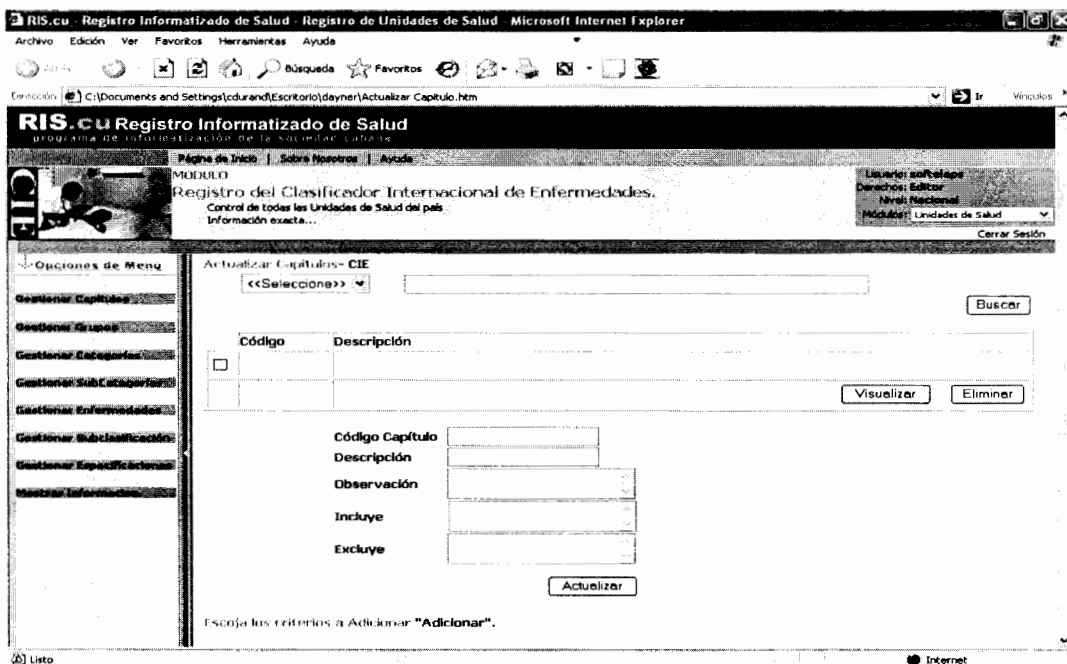


Fig.18 Gestionar Capítulos. Actualizar Capítulo.

RIS.cu - Registro Informatizado de Salud - Registro de Unidades de Salud - Microsoft Internet Explorer

Archivo Edición Ver Favoritos Herramientas Ayuda

Directorio C:\Documents and Settings\cdurand\Escritorio\dayner\Adicionar Grupo.htm

RIS.cu Registro Informatizado de Salud
programa de informatización de la sociedad cubana

Página de Inicio Sobre Nosotros Ayuda

MODULO
Registro del Clasificador Internacional de Enfermedades.
Control de Registro del Clasificador Internacional de Enfermedades y problemas de Salud.
Información exacta...

Usuario: softtelaps
Denominación: Editor
Nivel: Nacional
Módulo: Unidades de Salud
Cerrar Sesión

Opcciones de Menu

- Gestionaer Capitulo
- Gestionaer Grupos
- Gestionaer Categorías
- Gestionaer SubCategorías
- Gestionaer Enfermedades
- Gestionaer Subclasificación
- Gestionaer Especificaciones
- Mostrar Informacion

Adicionar Grupo - CIE

Código Capítulo <<Selección>>

Código Grupo

Descripción

Observación

Incluye

Excluye

Adicionar

Escoja los criterios a Adicionar "Adicionar".

Copyright © 2004. RCD - Registro Centralizado de Denonias. SOFTEL - MINSAP Ministerio de Salud Pública.

Internet

Fig.19 Gestionar Grupos. Adicionar Grupo.

RIS.cu - Registro Informatizado de Salud - Registro de Unidades de Salud - Microsoft Internet Explorer

Archivo Edición Ver Favoritos Herramientas Ayuda

Directorio C:\Documents and Settings\cdurand\Escritorio\dayner\Actualizar Grupos.htm

RIS.cu Registro Informatizado de Salud
programa de informatización de la sociedad cubana

Página de Inicio Sobre Nosotros Ayuda

MODULO
Registro del Clasificador Internacional de Enfermedades.
Control de todas las Unidades de Salud del país
Información exacta...

Usuario: softtelaps
Denominación: Editor
Nivel: Nacional
Módulo: Unidades de Salud
Cerrar Sesión

Opcciones de Menu

- Gestionaer Capitulo
- Gestionaer Grupos
- Gestionaer Categorías
- Gestionaer SubCategorías
- Gestionaer Enfermedades
- Gestionaer Subclasificación
- Gestionaer Especificaciones
- Mostrar Informacion

Actualizar grupos - CIE

<<Selección>>

Buscar

Código	Descripción
<input type="checkbox"/>	

Visualizar Eliminar

Código Capítulo

Código Grupo

Descripción

Observación

Incluye

Excluye

Actualizar

Inicio Explorador de Wi... Windows Commander... Macromedia Dreamw... RIS.cu - Registro Inf... ES 21:10

Fig.20 Gestionar Grupos. Actualizar Grupo.

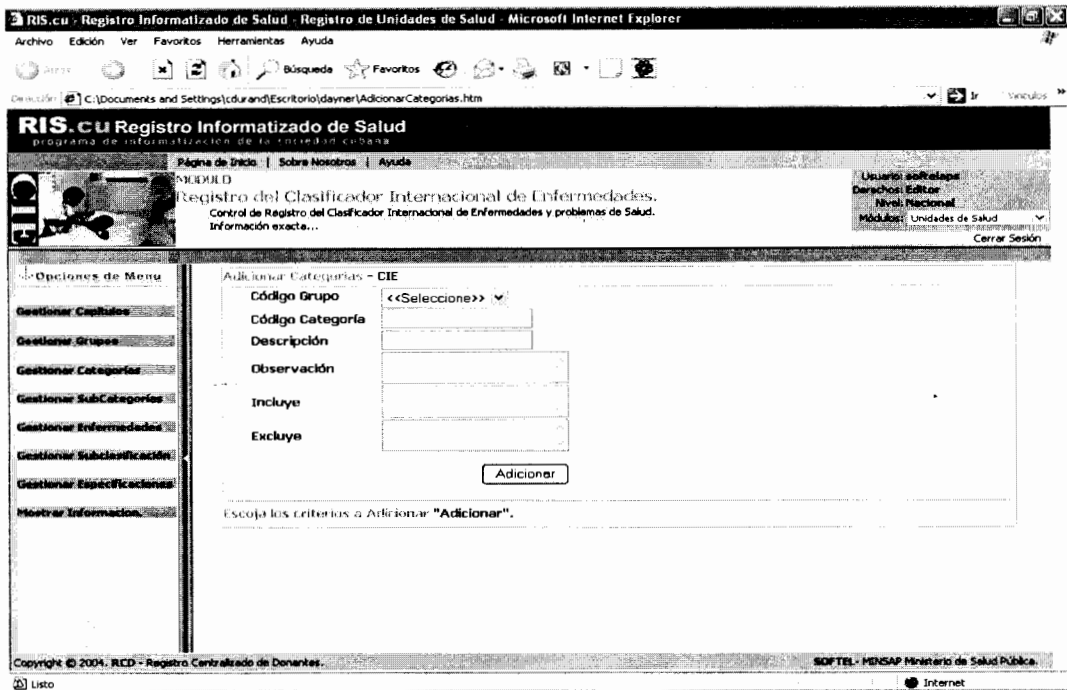


Fig.21 Gestionar Categorías. Adicionar Categorías

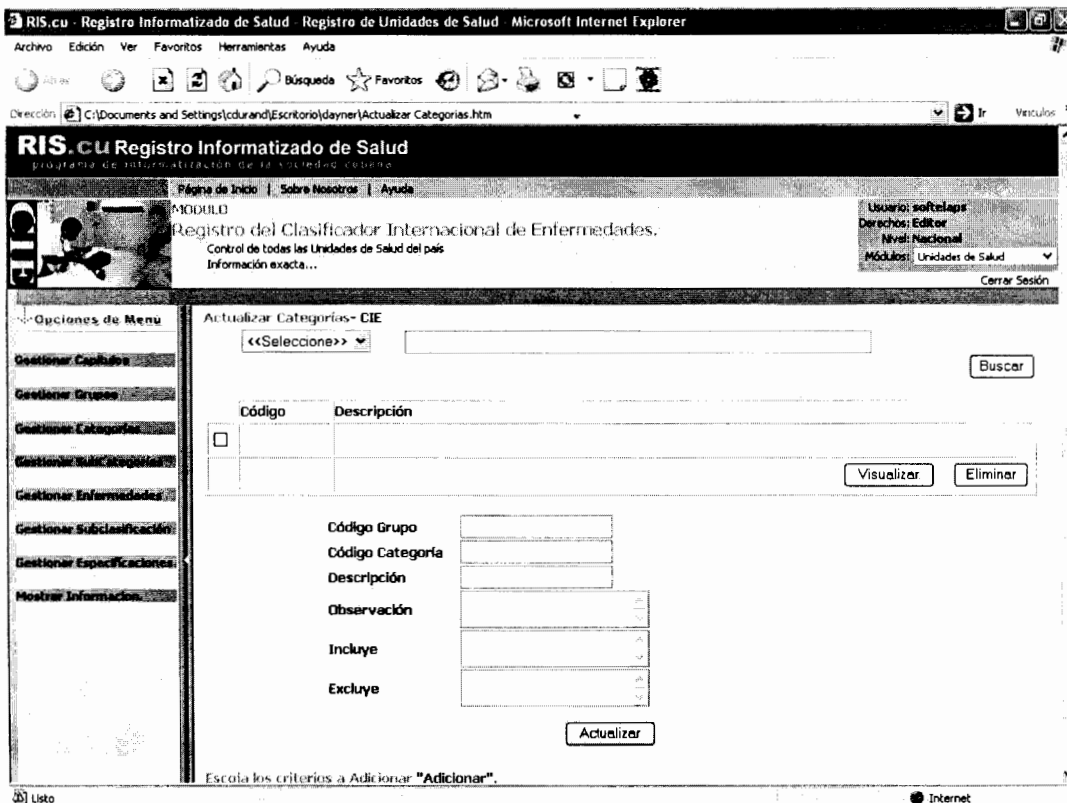


Fig.22 Gestionar Categorías. Actualizar Categorías

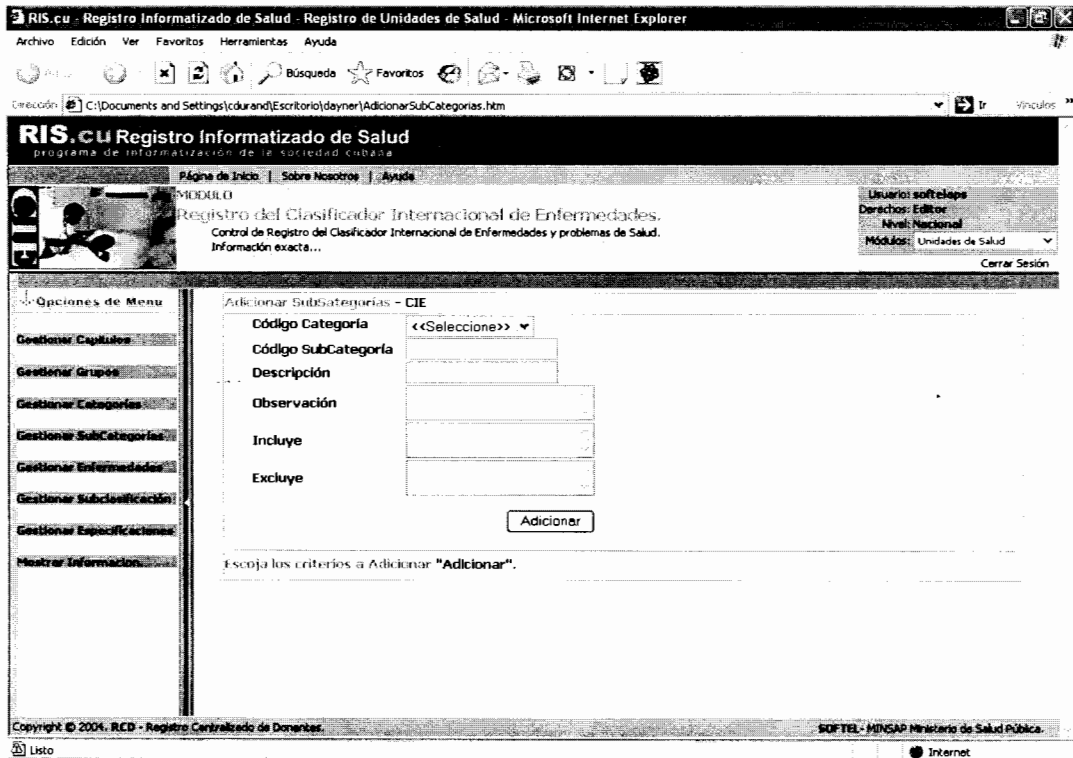


Fig.23 Gestionar Subcategorías. Adicionar Subcategorías

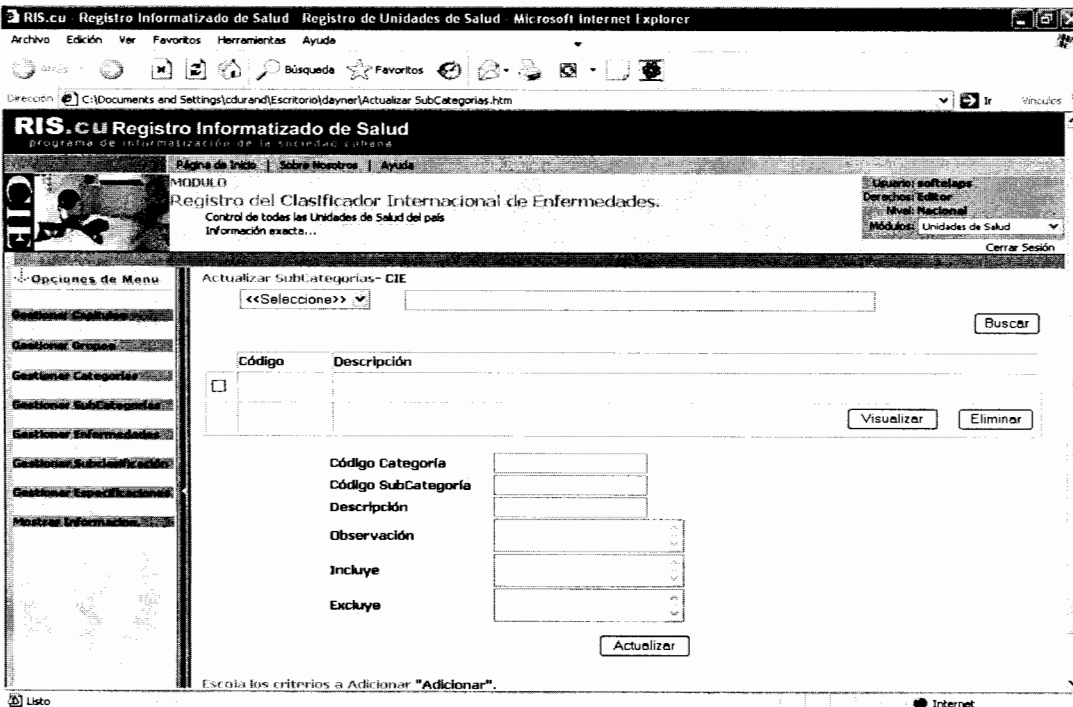


Fig.24 Gestionar Subcategorías. Actualizar Subcategorías

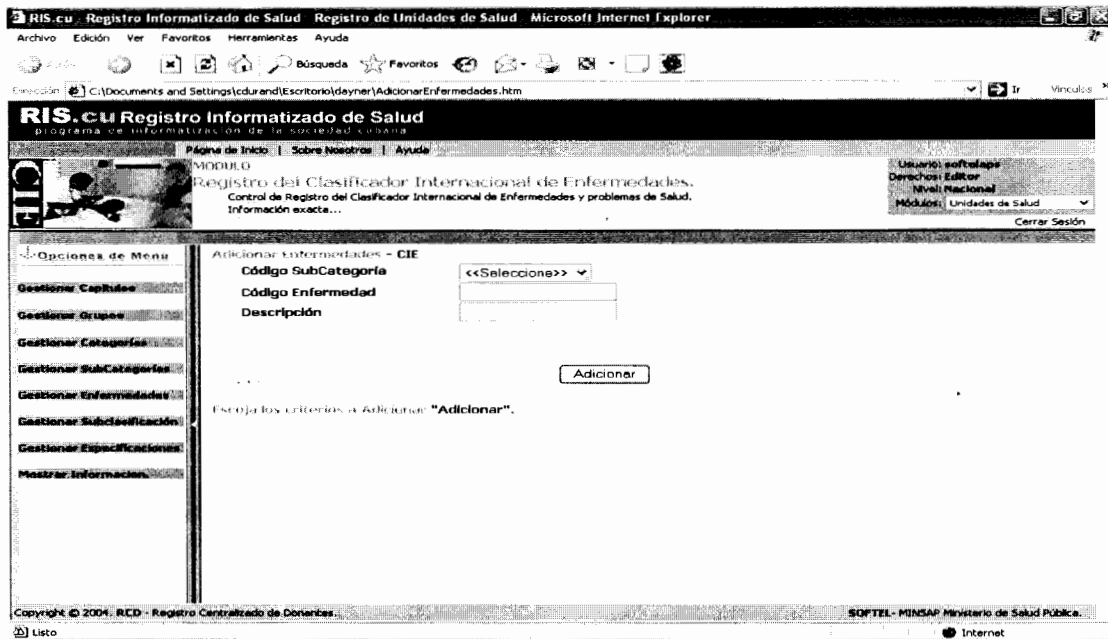


Fig.25 Gestionar Enfermedades. Adicionar Enfermedades.

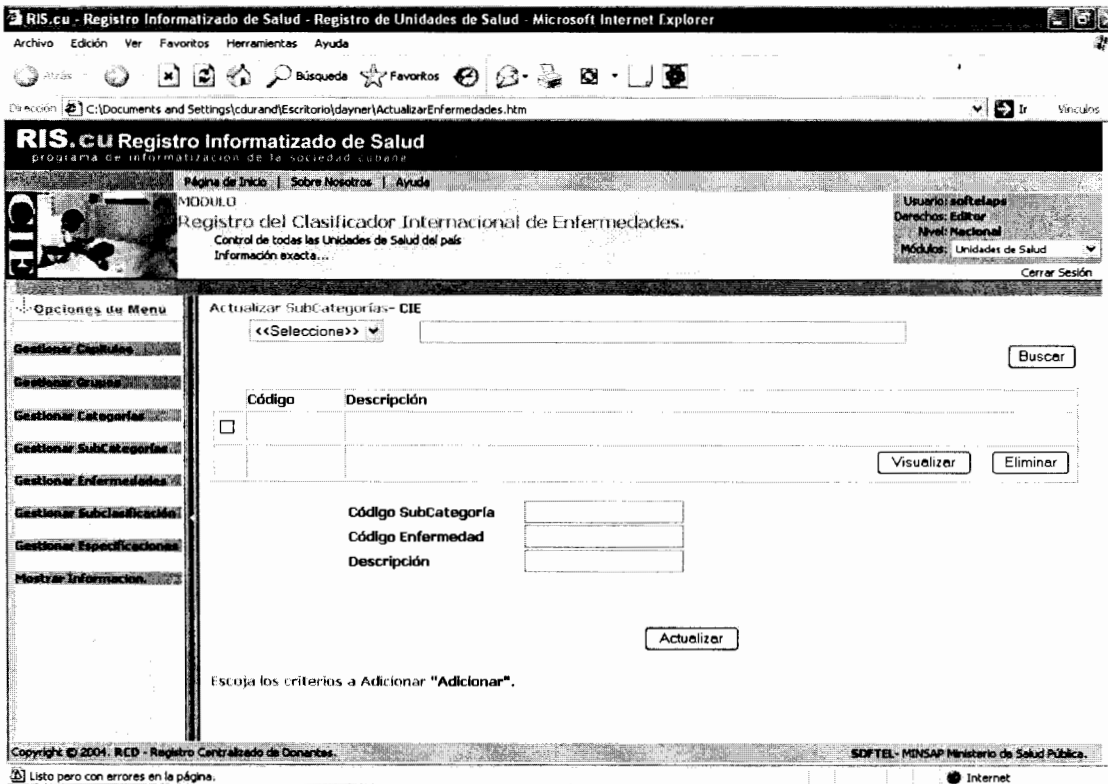


Fig.26 Gestionar Enfermedades. Actualizar Enfermedades.

RIS.cu - Registro Informatizado de Salud - Registro de Unidades de Salud - Microsoft Internet Explorer

Archivo Edición Ver Favoritos Herramientas Ayuda

De edición C:\Documents and Settings\cdurand\Escritorio\dayner\classific.htm

RIS.cu Registro Informatizado de Salud
Programa de informatización de la Sociedad Cubana

Página de Inicio | Sobre Nosotros | Ayuda

MODULO
Registro del Clasificador Internacional de Enfermedades.
Control de todas las Unidades de Salud del país
Información exacta...

Usuario: softelaps
Derechos: Editor
Nivel: Nacional
Módulos: Unidades de Salud
Cerrar Sesión

Opciones de Menu

Gestionar Capítulos
Gestionar Grupos
Gestionar Categorías
Gestionar SubCategorías
Gestionar Enfermedades
Gestionar Subclasificación
Gestionar Especificaciones
Mostrar Información

Adicionar Definiciones - CIE

Código Subclasificación Especificaciones

Descripción

Código Capítulo

Código Definición

Escoja los criterios a Adicionar "Adicionar".

Copyright © 2004. RED - Registro Centralizado de Donantes. SOFTEL - MINSAP Ministerio de Salud Pública.

Listo Internet

Fig.27 Gestionar Subclasificaciones. Adicionar Subclasificaciones.

RIS.cu - Registro Informatizado de Salud - Registro de Unidades de Salud - Microsoft Internet Explorer

Archivo Edición Ver Favoritos Herramientas Ayuda

De edición C:\Documents and Settings\cdurand\Escritorio\dayner\ActualizarSubClasificaciones2.htm

RIS.cu Registro Informatizado de Salud
Programa de informatización de la Sociedad Cubana

Página de Inicio | Sobre Nosotros | Ayuda

MODULO
Registro del Clasificador Internacional de Enfermedades.
Control de todas las Unidades de Salud del país
Información exacta...

Usuario: softelaps
Derechos: Editor
Nivel: Nacional
Módulos: Unidades de Salud
Cerrar Sesión

Opciones de Menu

Gestionar Capítulos
Gestionar Grupos
Gestionar Categorías
Gestionar SubCategorías
Gestionar Enfermedades
Gestionar Subclasificación
Gestionar Especificaciones
Mostrar Información

Adicionar Subclasificaciones - CIE

<<Selecciones>>

Código	Descripción
<input type="checkbox"/>	

Código Subclasificación Especificaciones

Descripción

Código Capítulo

Código Definición

Escoja los criterios a Adicionar "Adicionar".

Copyright © 2004. RED - Registro Centralizado de Donantes. SOFTEL - MINSAP Ministerio de Salud Pública.

Listo Internet

Fig.28 Gestionar Subclasificaciones. Actualizar Subclasificaciones.

RIS.cu - Registro Informatizado de Salud - Registro de Unidades de Salud - Microsoft Internet Explorer

Archivo Edición Ver Favoritos Herramientas Ayuda

Directorio C:\Documents and Settings\cdurand\Escritorio\dayner\AdicionarDefiniciones.htm

RIS.cu Registro Informatizado de Salud
Programa de Informatización de la Sociedad Cubana

Página de Inicio | Sobre Nosotros | Ayuda

MODULO
Registro del Clasificador Internacional de Enfermedades.
Control de todas las Unidades de Salud del país
Información exacta...

Usuario: softelaps
Derechos: Editor
Nivel: Nacional
Módulos: Unidades de Salud

Cerrar Sesión

Opciones de Menu

Gestionar Capitales
Gestionar Grupos
Gestionar Categorías
Gestionar SubCategorías
Gestionar Enfermedades
Gestionar Subclasificación
Gestionar Especificaciones
Mostrar Información

Adicionar Definiciones - CIE

Código Definición
Descripción

Adicionar

Escoja los criterios a Adicionar "Adicionar".

Copyright © 2004. RCD - Registro Centralizado de Donantes. SOFTEL - MINSAP Ministerio de Salud Pública.

Listo Internet

Fig.29 Gestionar Definiciones. Adicionar Definiciones.

RIS.cu - Registro Informatizado de Salud - Registro de Unidades de Salud - Microsoft Internet Explorer

Archivo Edición Ver Favoritos Herramientas Ayuda

Directorio C:\Documents and Settings\cdurand\Escritorio\dayner\ActualizarDefiniciones.htm

RIS.cu Registro Informatizado de Salud
Programa de Informatización de la Sociedad Cubana

Página de Inicio | Sobre Nosotros | Ayuda

MODULO
Registro del Clasificador Internacional de Enfermedades.
Control de todas las Unidades de Salud del país
Información exacta...

Usuario: softelaps
Derechos: Editor
Nivel: Nacional
Módulos: Unidades de Salud

Cerrar Sesión

Opciones de Menu

Gestionar Capitales
Gestionar Grupos
Gestionar Categorías
Gestionar SubCategorías
Gestionar Enfermedades
Gestionar Subclasificación
Gestionar Especificaciones
Mostrar Información

Actualizar Definiciones - CIE

<<Seleccione>>

Buscar

Código	Descripción
<input type="checkbox"/>	

Visualizar Eliminar

Código Definición
Descripción

Actualizar

Escoja los criterios a Adicionar "Adicionar".

Copyright © 2004. RCD - Registro Centralizado de Donantes. SOFTEL - MINSAP Ministerio de Salud Pública.

Listo Internet

Fig.30 Gestionar Definiciones. Actualizar Definiciones.

The screenshot shows the 'Búsqueda Normal - CIE' interface. At the top, there is a navigation bar with 'Página de Inicio', 'Sobre Nosotros', and 'Ayuda'. Below this, the 'MODULO' is identified as 'Registro del Clasificador Internacional de Enfermedades'. A user profile is visible in the top right corner, showing 'Usuario: softelaps', 'Derechos: Editor', 'Nivel: Nacional', and 'Módulos: Unidades de Salud'. A 'Cerrar Sesión' link is also present.

The main content area features a search form with a 'Código' dropdown menu and a 'Buscar' button. Below the form, there is a table with columns for 'Código' and 'Descripción'. A message below the table reads: 'Escoja los criterios de búsqueda y presione "Buscar".'

On the left side, there is a vertical menu titled 'Opciones de Menu' with several items: 'Gestionar Capítulos', 'Gestionar Grupos', 'Gestionar Categorías', 'Gestionar SubCategorías', 'Gestionar Enfermedades', 'Gestionar Subclasificación', 'Gestionar Especificaciones', and 'Mostrar Información'.

The footer contains copyright information: 'Copyright © 2004. RED - Registro Centralizado de Donantes.' and 'SOFTEL- MINSAP Ministerio de Salud Pública.' along with a 'Listo' status indicator and an 'Intranet local' link.

Fig. 31 Buscar Información. Búsqueda Estándar.

The screenshot shows the 'Búsqueda Avanzada - CIE' interface. It follows the same layout as the standard search page, including the navigation bar, user profile, and left-hand menu.

The search form is more complex, featuring five dropdown menus for selection: 'Capítulos', 'Grupos', 'Categorías', 'SubCategorías', and 'Enfermedades', each with a 'Codigo' dropdown. A 'Buscar' button is located to the right of these menus.

Below the search form, there is a table with columns for 'Código' and 'Descripción'. There are two 'Datos' buttons positioned below the table. A message below the table reads: 'Escoja los criterios de búsqueda y presione "Buscar".'

The footer contains the same copyright and institutional information as the standard search page: 'Copyright © 2004. RED - Registro Centralizado de Donantes.' and 'SOFTEL- MINSAP Ministerio de Salud Pública.' along with a 'Listo pero con errores en la página.' status indicator and an 'Intranet local' link.

Fig. 32 Buscar Información. Búsqueda Ampliada.