

INTRODUCCIÓN

Cuba enfrenta el reto de informatizar la sociedad con vistas a integrarse plenamente a la infraestructura global de la información, así como hacer un uso óptimo de las nuevas tecnologías. Como parte de la “Batalla de Ideas” y del proceso de informatización de la sociedad, el Ministerio de Salud Pública (MINSAP) desarrolla múltiples tareas orientadas por la dirección del país y uno de los programas que más expectativas y posibilidades abre en el campo del conocimiento, la información científica y la asistencia médica está relacionado con la Informatización del Sistema Nacional de Salud.

El MINSAP, ha definido la informatización del sector como tarea fundamental, en busca de la optimización de los servicios que se brindan a la población, mayor productividad y competencia en el desempeño de sus profesionales y técnicos, control en la administración de sus recursos, eficacia y eficiencia en su gerencia.

Como parte del proceso de informatización han surgido varios proyectos, uno de ellos es implementar un Sistema Integral de Salud, teniendo como centro del mismo al policlínico. El mismo apoya las estrategias y políticas trazadas por la dirección del país y el MINSAP; logrando 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, logrando que las instituciones del país alcancen un elevado nivel de informatización de las actividades que brindan.

Como parte de este proyecto surge el Registro Informatizado de Salud (RIS), una herramienta de ayuda en la toma de decisiones a partir de toda la información que se recopila en sus registros y se encuentra disponible en la red pública de transmisión de datos, pudiendo acceder al mismo desde cualquier computadora bajo el dominio .cu. El RIS a su vez tiene varios registros, uno de ellos será el Registro de Medios de Diagnósticos para el Registro Informatizado de Salud (Laboratorios Clínicos y de Microbiología).

La creación y desarrollo de la Red Telemática de Salud en Cuba (INFOMED), aporta la infraestructura necesaria que comunicará a todas las instituciones y centros de salud del país a través de una densa red de computadoras posibilitando las consultas e intercambio científico,

así como aportando la comunicación necesaria para la interacción de las aplicaciones que forman parte de la solución que se espera.

La medicina clínica experimenta desde los mismos comienzos del siglo XIX un rápido desarrollo, una casi total transformación nacida de las concepciones anátomo-clínica, fisiopatológicas y etiopatológicas de la enfermedad que en ese momento se formulaban y motivado, también, por un acentuado proceso de tecnificación del quehacer médico. Surgieron así la auscultación mediata, varias formas de endoscopias, la electrocardiografía, las pruebas funcionales y los medios diagnósticos de laboratorio, entre otros.

La lucha social contra la enfermedad se orienta hoy claramente al logro de una medicina preventiva que pretende evitar la aparición de la enfermedad.

La prevención y curación de las enfermedades del hombre, objeto de la medicina, precisa para su real eficacia de la coordinación de todas aquellas disciplinas que tienen implicaciones en los tres procesos básicos sobre los que se orienta el quehacer médico de cada día a saber, el **diagnóstico**, el reconocimiento de un agente o factor como causa del estado morbozo y, por último, el terapéutico. Las principales contribuciones son diagnósticas, pronósticas y de sostén.

Para diagnosticar una enfermedad existen diferentes medios de diagnósticos, entre los que podemos nombrar: Anatomía Patológica, Biología Molecular, Citología, Electrocardiografía, Servicios de Imágenes, Medicina Nuclear y Laboratorio Clínico y de Microbiología.

Actualmente en el país en los medios de diagnósticos el manejo de la información resulta complejo debido a la falta de un estándar nacional por el cual los laboratorios puedan regirse. Esto trae como consecuencia que la información sea tratada de distintas formas en los diferentes laboratorios del país. Además se dificulta la recopilación de los exámenes, métodos analíticos y reactivos al nivel nacional.

Luego de un profundo análisis de la situación de los medios de **diagnósticos** en el Sistema Nacional de Salud identificamos el siguiente **problema**: ¿cómo facilitar la gestión de la información estandarizada y homogénea de los medios de diagnósticos relacionada con los laboratorios clínicos y de microbiología?

El **objeto de estudio** de este trabajo lo constituye: todo el proceso de gestión de la información de los medios de diagnósticos del laboratorio clínico y de microbiología utilizados en los diferentes niveles de atención para el Sistema Nacional de Salud, aparejado al estudio de herramientas y metodologías que permitan el diseño informático.

De ello se deriva que el **campo de acción** es: el proceso del registro que gestiona y controla la información estandarizada de los medios de diagnósticos de los laboratorios clínicos y de microbiología en los niveles de atención de salud.

Para dar solución al problema se propone como **objetivo general**: modelar un sistema informático capaz de registrar y mantener actualizados los codificadores nacionales de los medios de diagnósticos de los laboratorios clínicos y de microbiología, logrando una información estándar de los mismos en todos los niveles de atención de salud.

Como **objetivos específicos** se proponen:

1. Realizar un estudio detallado de todos los temas relacionados con los medios de diagnósticos clínicos y de microbiología en el país.
2. Realizar un estudio de las principales tendencias y tecnologías actuales, para darle una solución óptima al sistema.
3. Realizar el análisis y el diseño del sistema de forma tal que posibilite la estandarización de la información relacionada con los medios de diagnósticos clínicos y de microbiología a nivel nacional y el control de los exámenes, métodos analíticos y reactivos que se utilizan en los laboratorios del país.
4. Diseño de una base de datos que almacene la información que se genera de los procesos de forma consistente.

La investigación transcurrirá a través de las siguientes **tareas**:

1. Investigar las actividades que se realizan en los laboratorios clínicos y de microbiología y la información de sus medios de diagnósticos.
2. Elaborar un prototipo que satisfaga las necesidades del cliente.
3. Realizar un estudio de la arquitectura idónea para la aplicación.

4. Estudiar las tendencias y tecnologías actuales, para que la solución que se decida utilizar sea la más óptima.
5. Desarrollar el modelado de la aplicación utilizando el Proceso Unificado de Desarrollo RUP (Rational Unified Process) y el Lenguaje Unificado de Modelación (UML).
6. Documentar todo el proceso de desarrollo.

El presente trabajo está estructurado en cinco capítulos.

El primer capítulo expone la fundamentación del tema, se habla acerca de la asociación de los principales conceptos relacionados con el dominio del tema, necesarios para entender el negocio actual, de los objetivos generales y específicos.

El segundo capítulo hace énfasis en las tendencias y tecnologías más actuales. Se recogen las principales herramientas y el proceso para la modelación de la aplicación; se comparan y se seleccionan las mejores propuestas.

El tercer capítulo plantea el modelo del dominio de la aplicación, los requisitos y casos de uso del sistema, diagramas y la descripción de los casos de uso.

El cuarto capítulo muestra el diagrama de clases del análisis. En la parte del diseño se exponen los detalles relacionados con el diseño del sistema propuesto. También se muestra el diagrama de clases.

El quinto capítulo y final se ocupa del proceso de factibilidad del sistema, mediante un análisis costo / beneficio.

Para finalizar se incluyen en el documento las conclusiones del trabajo, recomendaciones propuestas, bibliografía utilizada y anexos para una mayor documentación de las tareas realizadas.

CAPÍTULO 1 FUNDAMENTACIÓN DEL TEMA

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 del registro de medios de diagnósticos clínicos y de microbiología. Se define además el objeto de estudio, el problema y los objetivos generales y específicos de la presente investigación.

1.1 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]

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), el Control de Enfermedades Transmisibles y Crónicas no Transmisibles, y el de Atención al Adulto Mayor, todos 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]

1.1.1 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 y niveles de atención.
- Disfrutar 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.
- Aumento 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.
- Elevar 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.

1.1.2 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.

1.1.3 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.

1.1.4 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). Ver Anexo-I.

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 Diagnósticos, Registro del Clasificador 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.

1.2 Conceptos asociados al dominio del problema.

Laboratorio: Especialidad de la medicina que se ocupa de estudiar el diagnóstico, el pronóstico y la evolución de las enfermedades mediante métodos químicos.

Medios de Diagnósticos: Clasificación de las Diferentes especialidades diagnósticas. Por ejemplo Laboratorio Clínico, Microbiología, Servicios de imágenes, Electrocardiografía, etc.

Métodos Analíticos: Herramienta de referencia para usar en el trabajo que ayuda al personal a realizar un examen.

Modo de conservación: Forma de almacenamiento de las muestras colectadas a los pacientes para realizar los exámenes de Laboratorio. Ejemplo: Refrigerada, Congelada, A temperatura ambiente, Protegida de la luz, etc.

Examen: Medición o análisis de un espécimen con el propósito del diagnóstico, la prevención o el seguimiento de alguna enfermedad o para la evaluación del estado de salud.

1.3 Objeto de estudio.

1.3.1 Descripción general.

Formando parte de la informatización en el sector de la salud se encuentra el Sistema Integral de Salud (SISalud) que permitirá la integración de los diferentes módulos que se desarrollen para satisfacer las necesidades en los distintos niveles de la salud.

El SISalud estará compuesto por el Registro Informatizado de Salud (RIS), el Sistema Informatizado de Atención Primaria (SIAP) y el Sistema Informatizado de Gestión Hospitalaria (SIGH).

El RIS está formado por los registros que son administrados o gestionados a nivel nacional o central 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, este responde a una distribución centralizada de sus componentes.

El Registro de Medios de Diagnósticos para el Registro Informatizado de Salud (Laboratorios Clínicos y de Microbiología), formará parte del RIS y permite la estandarización de la información de estos medios de diagnósticos a nivel nacional, facilitando la gestión en los laboratorios y la obtención de estadísticas a nivel nacional.

1.3.2 Descripción actual.

Actualmente en el país el manejo de la información de los medios de diagnósticos carece de un estándar nacional por el cual los laboratorios puedan registrarse. Esto trae como consecuencia que la información sea tratada de distintas formas en los diferentes laboratorios, se dificulta controlar cuáles exámenes, métodos analíticos y reactivos se realizan y se utilizan en los laboratorios del país y obtener estadísticas nacionales de los medios de diagnósticos.

1.4 Conclusiones.

Verdaderamente con el auge que ha tomado Internet y la consecuente digitalización de la información se ha acelerado gradualmente el conocimiento científico y la evolución tecnológica. Por ello es muy importante una excelente selección de la tecnología que necesita la aplicación que se debe realizar, para que se logren los mejores resultados y se explote el potencial tecnológico requerido.

Las condiciones y problemas que rodean el objeto de estudio; y a través de los conceptos y definiciones planteadas, se determinaron las condiciones específicas que rodean al problema y sobre esta base se obtuvieron los objetivos generales y específicos para este trabajo. Este tiene gran importancia porque constituye la base para el futuro desarrollo del registro de medios de diagnósticos del laboratorio clínico y de microbiología.

CAPITULO 2 TENDENCIAS Y TECNOLOGÍAS ACTUALES

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.

2.1 ¿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 Estados Unidos de América crea la Agencia de Proyectos de Investigaciones Avanzadas (ARPA)¹, como respuesta al lanzamiento por la Unión Soviética del primer satélite artificial de la tierra. Esta agencia se encargaría de desarrollar proyectos técnicos e investigativos aplicables en el campo militar. [8]

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. [9]

Es conveniente recalcar que la meta de ARPA no era la creación de la red internacional de computadoras existente hoy día, sino 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.

¹ ARPA: Advanced Research Projects Agency.

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. [9]

La red ARPANET conectaba las Universidades de California en Santa Bárbara y en los Ángeles, la de UTAH en Salt Lake City y la Universidad de Standford. Esta tecnología possibilitaba 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 (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 Estados Unidos de América 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. [10]

2.1.1 ¿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. [11]

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.

2.1.2 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). [12]

2.2 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 para lo cual 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. [13]

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 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.

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 sistema. 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. [14]

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. [15]

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. [16]

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. [17]

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:

- 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.

2.5 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 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 ¿Qué. 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. [18]

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. [19]

2.5.1 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. [20]

2.6 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. 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 (stateless), 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 ediumtic y se desconecta.

2.6.1 PERL (Practical Extracting and Reporting Language).

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]

2.6.2 ASP (Active Server Pages).

ASP 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]

2.6.3 PHP (Personal Home Page).

PHP 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. [24]

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 encontrar ayuda, documentación, artículos, noticias, y demás recursos. [25]

Desventaja:

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

2.6.4 JSP (Java Server Pages).

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. [26]

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.

2.6.5 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. [27]

2.6.6 XSLT.

XSLT es el acrónimo de Extensible Stylesheet Lénguage 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. [28]

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.

2.6.7 ¿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.

2.7 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. [29]

2.7.1 SQL SERVER.

Microsoft SQL Server 2000 es uno de los mejores SGBD 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.

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. [30]

Desventajas:

- Licencias con costos altos.
- Plataformas Windows.

2.7.2 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.

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. [31]

2.7.3 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 *Medium*. 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 subconsultas, 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. [32]

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.

2.8 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.

2.8.1 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.

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.

2.8.2 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 necesaria 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 Software (RUP):

- 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

2.9 Plataforma de Servicio (PLASER).

La plataforma de Servicios (PLASER) está conformada fundamentalmente por varias clases en PHP, una librería, que puede ser usada opcionalmente para que un componente se integre al Sistema Integral de Salud (SISalud), pero de no ser usada, la seguridad del sistema corre a cuenta del programador. En esta versión PLASER sólo 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 está concebido completamente sobre Arquitectura Basada en Componentes y Orientada a Servicios, usando el paradigma de XML Web Services específicamente SOAP. En su concepción se han utilizado estándares actuales y normas abiertas. PLASER constituye una plataforma sobre la que se pueden desplegar aplicaciones XML – Web Services, con la ventaja de que el programador no tiene que preocuparse por implementar la seguridad del Sistema, ya que esta es una de las tareas que asume PLASER, además facilita la programación y homogeneidad de los componentes. PLASER desde el punto de vista estructural permite trabajar con cualquier base de datos que cumpla con la norma SQL ANSI 92; pero desde el

punto de vista de implementación sólo 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.

2.10 Herramientas a utilizar.

Se decidió que se utilizaría el RationalRose 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.

2.11 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, fundamentándose las elecciones del lenguaje, el sistema gestor de bases de datos, y la metodología a utilizar.

CAPÍTULO 3 DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA

El desarrollo de la aplicación se ha basado en el Proceso Unificado de Desarrollo de Software (RUP), que hace uso del Lenguaje Unificado de Modelado (UML) utilizando la herramienta case Rational Rose. En el campo del software también se necesita la creación de modelos que organicen y presenten los detalles importantes de los problemas reales que se vinculan con el sistema informático a construir. Uno de los modelos útiles previo al desarrollo de un software es el modelo del negocio, técnica que se usa para comprender los procesos del negocio; a partir del mismo se derivan los requerimientos para la aplicación.

En el presente capítulo se presenta el modelo del dominio del Registro de Medios de diagnósticos para los laboratorios clínicos y de microbiología, este modelo se considera en RUP un subconjunto del modelo de objetos del negocio, captura los tipos más importantes de objetos que existen, no incluye las responsabilidades de las personas que ejecutan las actividades y se utiliza debido a la poca estructuración del proceso.

Se refinan y estructuran los requerimientos obtenidos, describiéndose lo que se espera del sistema a través de los casos de uso del sistema, organizados en paquetes.

3.1 Descripción de los procesos del negocio propuestos.

La terminología relacionada con los laboratorios clínicos y los microbiológicos es muy diversa. En ocasiones se utilizan diferentes términos para denominar un objeto o actividad y en otras, un mismo término se presenta con varias definiciones, dependiendo del medio en que se utilice. Esto trae como consecuencia que no exista un estándar a nivel nacional y por ende que no sea homogénea la información que se maneja en los medios de diagnósticos.

3. 2 Conceptos del dominio.

Algoritmo: Expresión de cálculo utilizada para obtener el resultado de un examen y que puede involucrar el resultado de otros exámenes.

Clave de Rechazo: Requisitos técnicos por los cuales el personal técnico del laboratorio es capaz de rechazar muestras útiles en el laboratorio. Ejemplo: Muestra de suero icterico.

Conservación de la muestra: Forma o modo de almacenar la muestra y/o la adición de algún reactivo de conservación, con el objetivo de alargar su período útil o prepararla para la realización de exámenes de Laboratorio.

Diccionario: Sirve para almacenar grandes volúmenes de información referente a todos los resultados posibles de los exámenes. Por ejemplo para introducir los resultados de los Antibiogramas se pueden crear diccionarios de Bacterias, Parásitos, Hongos, virus, etc.

Forma de expresión del resultado: Forma en que puede expresarse el resultado de un examen, pueden ser Cualitativo, Cuantitativo, Diccionario, Texto, Imagen, etc.

Laboratorios de Referencia: Laboratorios con capacidad para acreditar la calidad del trabajo de otros Laboratorios

Preparación del Paciente: Forma de preparación del paciente para la realización del examen.

Reactivo o medio de cultivo: Sustancia que se utilizan en el laboratorio para conservar o preparar las muestras.

Recipiente de Conservación: Recipiente donde se conserva la muestra.

Sistema de calibración: Material de referencia que se adquiere y que permite estandarizar las mediciones en los laboratorios (calibrar).

Tipo de conservación de la muestra: Describe las diferentes formas y procedimientos de conservación para cada tipo de muestra.

Tipos de Examen: Clasificación que se le da al examen. Ejemplo Simple, Compuesto o Perfil.

Tipo de Muestra: Clasificación de las muestras teniendo en cuenta sus características individuales. Ejemplo: Espujo, Sangre, Orina, Esperma, Tejido.

Tipo de resultado: Clasificador de los diferentes tipos de resultados posibles de los exámenes.

Unidad de medida: Define las diferentes unidades de medidas que se utilizan en la adición de reactivos de conservación de las muestras y en la cantidad de muestra preparada en la realización de los exámenes.

Valores Diccionario: Para cada diccionario es posible registrar todos los posibles valores del resultado de exámenes cuya forma de expresión es precisamente un diccionario.

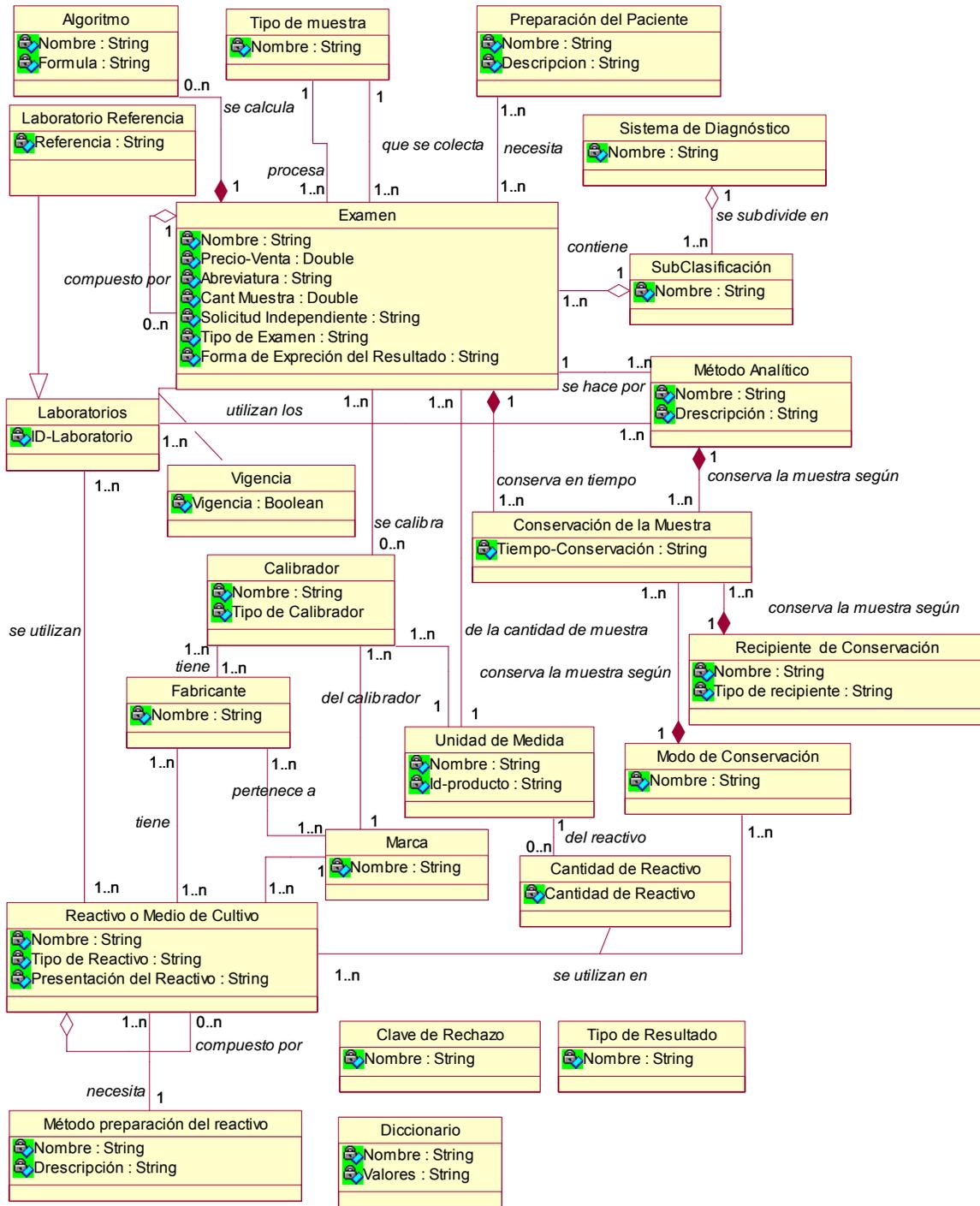


Figura 3.1 Modelo de Dominio

3.3 Requerimientos funcionales.

RF1. Crear un codificador nacional para los diferentes exámenes que pueda ser utilizado desde cualquier sistema informático o de información en este Sistema de Diagnósticos.

RF2. Gestionar la información de los Exámenes.

2.1 Código.

2.2 Nombre.

2.3 Preparación del Paciente.

Un Examen puede tener varias Preparaciones.

2.4 Algoritmos a utilizar para los cálculos.

Un Examen puede tener varios Algoritmos.

2.5 Tipo de Examen.

2.6 Precio de venta.

2.7 *Tipo de Muestra* del Examen.

2.8 Conservación de la *muestra*, depende del método utilizado.

2.9 Abreviatura.

2.10 Formas de Expresión del Resultado.

Un Examen puede tener varias Formas de Expresión del resultado.

2.11 Sistema de Calibración.

Un Examen puede tener varios Sistemas de Calibración.

2.12 *Tipo de Muestra* Colectada.

2.13 Cantidad de Muestra Colectada.

2.14 Unidad de Medida de la Muestra Colectada.

2.15 Se solicita independiente.

Un examen puede estar compuesto por varios exámenes.

RF3. Trabajar con el Sistema Internacional de Unidades para las Unidades de Medida.

3.1 Nombre.

3.2 Identificador.

www.sc.ehu.es/sbweb/fisica/unidades/unidades/unidades.htm

RF4. Gestionar la información de los Métodos Analíticos.

4.1 Nombre del Método.

4.2 Descripción.

- RF5. Configurar para cada Examen los diferentes Métodos Analíticos con que este puede realizarse.
- RF6. Gestionar la información de los Sistemas de Calibración.
 - 6.1 Nombre.
 - 6.2 Tipo de Calibrador.
 - 6.3 Unidad de medida.
 - 6.4 Fabricante.
 - 6.5 Marca.
- RF7. Gestionar la información de los diferentes Tipos de Muestra.
 - 7.1 Nombre del Tipo de Muestra.
- RF8. Permitir crear Perfiles de exámenes.
- RF9. Gestionar la información de la Preparación de los Pacientes
 - 9.1 Nombre de la Preparación del Paciente.
 - 9.2 Descripción.
- RF10. Gestionar la información de los Diccionarios.
 - 10.1 Nombre del Diccionario.
 - 10.2 Valores del Diccionario (Un Diccionario puede tener varios valores).
- RF11. Configurar los algoritmos a utilizar para los cálculos:
 - 11.1 Nombre del Algoritmo.
 - 11.2 Fórmulas (puede incluir operaciones y funciones matemáticas sobre los exámenes.)
- RF12. Gestionar la información de los Laboratorios y Laboratorios de referencia si existen:
 - 12.1 *Área de Salud.*
 - 12.2 Unidad de Salud a que pertenece.
 - 12.3 Exámenes que se realizan.
 - 12.4 Reactivos que Utiliza.
 - 12.5 Métodos Analíticos que Utiliza por Examen.
- RF13. Acceder al Registro de Unidades de Salud para obtener los datos de la unidad.
- RF14. Gestionar la información de las Claves de Rechazo.
 - 14.1 Nombre de la Clave de Rechazo.
- RF15. Visualizar por cada Examen su información y sus Exámenes asociados.
- RF16. Gestionar los Tipos de Exámenes.
 - 16.1 Nombre.

Debe tener por defecto los valores Simple, Compuesto y Perfil. No se pueden borrar ni modificar.

RF17. Gestionar los Modos de Conservación de la muestra.

17.1 Nombre.

17.2 Para cada Modo de Conservación, especificar:

17.2.1 Reactivos

17.2.2 Cantidad de reactivo

17.2.3 Unidad de medida del reactivo

RF18. Gestionar la Conservación de la Muestra por cada examen.

18.1 Modo de conservación de la muestra.

18.2 Método Analítico.

18.3 Recipiente de Conservación.

18.4 Tiempo de conservación.

RF19. Gestionar las Formas de expresión del resultado.

19.1 Nombre.

Debe tener por defecto los valores: Cuantitativo, Cualitativo, Diccionario, Texto, Imagen y Combinado (combinación de las formas anteriores). No se deben borrar ni modificar.

RF20. Gestionar los Tipos de Resultados.

20.1 Nombre.

RF21. Gestionar los Reactivos.

21.1 Nombre.

21.2 Identificador internacional.

21.3 Tipo de reactivo.

21.4 Presentación.

21.5 Método de preparación.

21.6 Marca.

21.7 Fabricante.

Un reactivo puede estar compuesto por varios reactivos.

RF22. Gestionar los diferentes Sistemas de Diagnósticos y las Subclasificaciones de cada uno.

22.1 Nombre.

RF23. Gestionar los Fabricantes.

23.1 Nombre.

RF24. Gestionar las Marcas.

24.1 Nombre.

RF25. Gestionar Recipientes de Conservación.

25.1 Nombre.

25.2 Tipo de Reciente.

RF26. Gestionar la información de los Métodos de Preparación del Reactivo.

26.1 Nombre.

26.2 Descripción.

RF27. Gestionar la Presentación de los Reactivos.

27.1 Nombre.

RF28. Definir los exámenes por tipo de unidad.

RF29. Gestionar la información de los tipos de reactivos y tipos de calibradores.

29.1 Nombre.

3.4 Requerimientos no funcionales.

Los requerimientos no funcionales se refieren a las cualidades del Sistema:

Apariencia o Interfaz Externa.

Tener conectividad con el Módulo Unidades de Salud y Áreas de Salud para asociar los Laboratorios a las unidades y a las áreas.

RNF2 La interfaz debe ser sencilla, intuitiva, amigable, debido a que los usuarios no son expertos en el uso de Aplicaciones Web.

El sistema debe estar diseñado para que la navegación y el acceso sean rápidos y fáciles..

Rendimiento.

RNF4 Debe mantener un diseño similar en todas las páginas, las mismas estarán poco cargadas de imágenes, para garantizar una respuesta más rápida del Sistema.

Usabilidad.

Capacitación del personal con conocimientos elementales de Informática y adiestramiento en el sistema informático que se instale en las clínicas

RNF6 La aplicación garantizará un acceso fácil y rápido a los usuarios. El sistema podrá ser usado por cualquier persona que posea conocimientos básicos en el manejo de la computadora

y de un ambiente Web en sentido general, por lo que el personal que trabaja con el Registro debe contar con el nivel técnico requerido mediante adiestramiento de servicio.

Garantizar que los **Laboratorios** tengan acceso a la información que se genera en este sistema para lograr la calidad en la información y la unicidad de la misma.

Portabilidad.

RNF8 Permitir que el sistema se ejecute sobre el Sistema Operativo Linux, Windows 98 o superior.

Autonomía.

RNF9 Cada capa se debe construir como componente independiente, facilitando el mantenimiento del software.

Seguridad.

RNF10 Disponer de un mecanismo de seguridad basado en el modelo de Autenticación, Autorización y Auditoría (AAA).

Confiabilidad.

RNF11 La información manejada por el sistema está protegida de acceso no autorizado. El sistema debe prevenir posibles fallos y/o errores y presentar facilidades para una rápida recuperación en dichos casos.

Integridad.

RNF12 La información manejada por el sistema será objeto de cuidadosa protección contra la corrupción y estados inconsistentes. Deberán existir mecanismos de chequeo de integridad. Se permitirá la creación de copias de respaldo que puedan restaurar el sistema en caso de fallo crítico o pérdida total de la información.

Disponibilidad.

RNF13 Los usuarios autorizados tendrán acceso a la información en todo momento. 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.

Escalabilidad.

RNF14 La arquitectura del sistema debe desarrollarse siguiendo un modelo cliente servidor de varias capas, Interfaz de usuario (presentación), Lógica de negocio, Acceso a Datos y Base de datos, con una arquitectura orientada a servicios y basada en componentes. La capa de negocio estará disponible a otras aplicaciones utilizando las tecnologías de xml web services.

RNF15 Todos los componentes del sistema deben desarrollarse siguiendo el principio de máxima cohesión y mínimo acoplamiento.

Software.

RNF16 Los clientes tendrán acceso al Registro de Medios de Diagnósticos a través de cualquier navegador Web. Recomendados: Mozilla 1.5, Internet Explorer 4.0 o superior.

Diseñar la recolección de los datos con preguntas cerradas y en los casos que sea posible ofrecer una lista de posibles respuestas.

El Sistema debe estar disponible para ser accedido por los **Laboratorios** del país en los que se instale el Sistema Informático del Laboratorio.

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.

Requerimientos mínimos del sistema:

- Procesador 486DX / 66 MHz o superior.
- 16 MB de memoria; a más memoria mayor rendimiento.
- Monitor VGA o superior.
- Ratón Microsoft o compatible.
- No existen restricciones específicas en cuanto al Servidor.
- RNF16 Impresora local o de red para imprimir los reportes solicitados.

3.5 Descripción de la solución propuesta.

El Registro de Medios de Diagnósticos al formar parte del RIS, tiene una arquitectura basada en componentes y orientada a servicios, es una aplicación con tecnología XML Web Services. Será desarrollado usando PHP como lenguaje de programación y MySQL como gestor de datos. Utilizará el framework PLASER realizado con la concepción del patrón de diseño Modelo de Vista Controlador (MVC) y conformada por varias clases en PHP, que pueden ser usadas y que garantizan la seguridad del sistema.

Un sistema grande se debe dividir en unidades más pequeñas, de modo que pueda ser entendido por las personas que necesiten consultarlo y además que los equipos de trabajo no interfieran con el trabajo de los otros.

Para satisfacer los objetivos propuestos al inicio de este trabajo y teniendo en cuenta los requerimientos planteados, así como las reglas que propone el control de calidad en los laboratorios; el sistema que se propone tiene los siguientes paquetes: (Figura 3.2):

1. Fase Pre-Analítica.
2. Fase Analítica.
3. Fase Pos-Analítica.
4. Laboratorio.
5. Configuración.

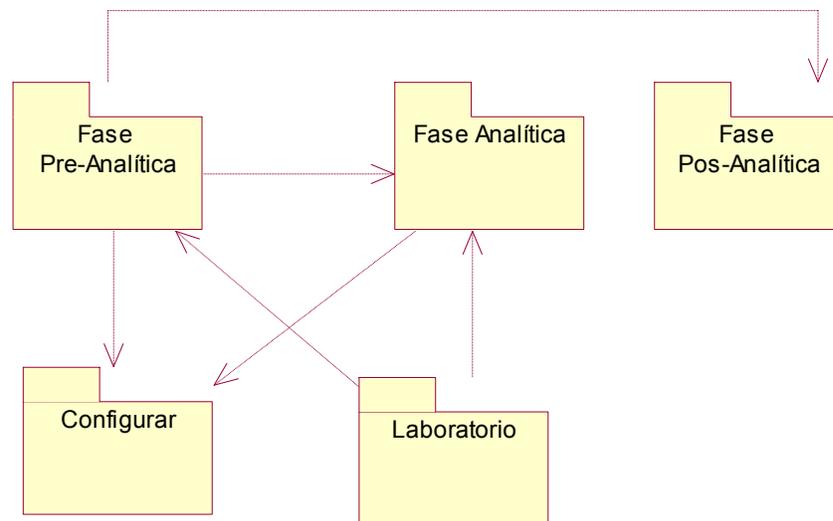


Figura 3.2 Diagrama de Paquetes.

FASE PRE-ANALÍTICA

Pasos del comienzo en orden cronológico para las especificaciones clínicas que incluyen las pruebas, la preparación del paciente, la colección de la muestra primaria, la transportación y todo lo que se le adicione dentro del laboratorio y finalmente cuando comienzan los procedimientos de ensayo.

FASE ANALÍTICA

En la fase analítica se realizan las mediciones y observaciones en las diversas áreas que cubre el laboratorio. Cada procedimiento de análisis debe describir lo relacionado con las mediciones, las características de ejecución que pretende el autor del procedimiento o el fabricante del sistema analítico y los procedimientos de control que corresponden a cada medición.

FASE POST-ANALÍTICA

Todos los procesos que incluyen una revisión sistemática de los mismos, con autorización formal para la liberación, el informe de los resultados, la comunicación de los resultados y el almacenamiento de las muestras de ensayo.

LABORATORIO

Se gestiona la información propia de cada laboratorio: exámenes, métodos analíticos y reactivos. Además se definen los laboratorios de referencia.

CONFIGURAR

Se configuran los codificadores que se utilizan para gestionar la información de los medios de diagnósticos.

Los **actores** del sistema pueden representar el rol que juega una o varias personas, un equipo o un sistema automatizado, son parte del sistema, y pueden intercambiar información con él o ser recipientes pasivos de información. En este caso los actores que interactúan con el sistema se definen a continuación en la tabla 2.3:

Actores del Sistema	Justificación
Grupo Nacional de Expertos	Registrar y mantener actualizados los codificadores nacionales de los Medios de Diagnósticos del Laboratorio para elevar la calidad en la gestión y controlar los exámenes que se realizan en cada laboratorio así como los reactivos y métodos analíticos que se utilizan.
Jefe de Laboratorio	Registrar y mantener actualizados los codificadores propios de cada laboratorio (exámenes, métodos analíticos y reactivos).
Personal de Laboratorio	Trabajador del laboratorio interesado en consultar la información de los codificadores creados a nivel nacional y en el propio laboratorio.
Registro de Unidad de Salud	Web Service que contiene y brinda los datos de las unidades de salud del país.
Registro de Ubicación	Web Service que contiene y brinda los datos de las provincias, municipios y localidades del país.
Registro de Área de Salud	Web Service que contiene y brinda los datos de las Áreas de Salud, Grupo Básico de Trabajo y Equipo Básico de Salud.

Tabla 3.1 Descripción de los actores del sistema.

3.6 Diagramas de casos de Uso del Sistema.

El diagrama de casos de uso del sistema es un modelo que contiene actores, casos de uso y sus relaciones. Los actores son terceros fuera del sistema que interactúan con él. Los casos de uso son fragmentos de funcionalidad que el sistema ofrece para aportar un resultado de valor para sus actores.

A continuación se muestra en la figura 3.3 el diagrama de casos de uso del sistema Registro de Medios de Diagnósticos del Laboratorio Clínico y de Microbiología.

Un caso de uso es un documento narrativo que describe la secuencia de un actor (agente externo) que utiliza un sistema para completar un proceso. A continuación se presentan los casos de uso del sistema que satisfacen los requerimientos funcionales.

3.6.1 Fase Pre-Analítica.

CASO DE USO DEL SISTEMA (1)	GESTIONAR PREPARACIÓN DEL PACIENTE
Actores:	Personal de Medios de Diagnósticos (Inicia)
Descripción:	El Caso de uso se inicia cuando el Personal de Medios de Diagnósticos accede al Registro de Medios de diagnósticos para actualizar o consultar la información de las preparaciones del paciente. Para actualizarla solo puede acceder el Grupo Nacional de Expertos. El caso de uso finaliza cuando se actualiza o se imprime la información.
Referencias:	RF9.
Precondiciones:	-
Poscondiciones:	Para crear: se crea una instancia de la clase Preparacion_del_Paciente. Para modificar: se modifican las entidades Nombre, Descripcion. Para eliminar: se elimina una instancia de la clase Preparacion_del_Paciente.
Requerimientos especiales:	Para modificar y eliminar tiene que existir la preparación.
Prototipo:	Anexo-III.

CASO DE USO DEL SISTEMA (2)	Gestionar Tipos de Muestras
Actores:	Personal de Medios de Diagnósticos (Inicia)
Descripción:	El Caso de uso se inicia cuando el Personal de Medios de Diagnósticos accede al Registro de Medios de diagnósticos para actualizar o consultar la información de los Tipos de Muestras. Para actualizarlo solo puede acceder el Grupo Nacional de Expertos. El caso de uso finaliza cuando se actualiza o se imprime la información.
Referencias:	RF7.
Precondiciones:	-

Poscondiciones:	Para crear: se crea una instancia de la clase Tipos_de_Muestra. Para modificar: se modifica la entidad Nombre. Para eliminar: se elimina una instancia de la clase Tipos_de_Muestra.
Requerimientos especiales:	Para modificar y eliminar tiene que existir el tipo de muestra.
Prototipo:	Anexo-IV.

CASO DE USO DEL Sistema (3)	GESTIONAR MODOS DE CONSERVACIÓN
Actores:	Personal de Medios de Diagnósticos (Inicia)
Descripción:	El Caso de uso se inicia cuando el Personal de Medios de Diagnósticos accede al Registro de Medios de diagnósticos para actualizar o consultar los Modos de Conservación. Para actualizarlo solo puede acceder el Grupo Nacional de Expertos. El caso de uso finaliza cuando se actualiza o se imprime la información.
Referencias:	RF3, RF16, RF20.
Precondiciones:	Debe haber sido gestionada previamente la información de los Reactivos y las Unidades de Medidas; deben existir en la base de datos.
Poscondiciones:	Para crear: se crea una instancia de la clase Modo_de_Conservacion. Para modificar: se modifica la entidad Nombre. Para eliminar: se elimina una instancia de la clase Modo_de_Conservacion.
Requerimientos especiales:	Para modificar y eliminar tiene que existir el modo de conservación.
Prototipo:	Anexo-V

ASO DE USO DEL SISTEMA (4)	GESTIONAR EXÁMENES
Actores:	Personal de Medios de Diagnósticos (Inicia)
Descripción:	<p>El Caso de uso se inicia cuando el Personal de MEDIOS DE DIAGNÓSTICOS accede al Registro de Medios de diagnósticos para actualizar o consultar la información de los exámenes. Para actualizarlo solo puede acceder el Grupo Nacional de Expertos. Para un examen se puede especificar también la información relacionada con los algoritmos que utiliza para los cálculos, preparación del examen, métodos analíticos por los que puede realizarse, conservación, sistemas de calibración, así como los exámenes por los que están compuestos. El caso de uso finaliza cuando se actualiza o se imprime la información.</p>
Referencias:	<p>RF2, RF3, RF4, RF5, RF6, RF7, RF8, RF9, RF11, RF14, RF15, RF16, RF17, RF18.</p>
Precondiciones:	<p>Debe haber sido gestionada previamente la información de las Preparaciones del Paciente, los Métodos Analíticos, los Sistemas de Calibración, los Tipos de Muestra, las Unidades de Medida, las Formas de expresión del resultado, los Tipos de exámenes, los Modos de Conservación, los Sistemas de Diagnósticos y las Subclasificaciones; estas entidades deben existir en la base de datos.</p> <p>Para gestionar los exámenes compuestos tienen que existir los exámenes.</p>
Poscondiciones:	<p>Para crear: se crea una instancia de la clase Examen.</p> <p>Para modificar: se modifican las entidades Nombre, PrecioVenta, Abreviatura, CantMuestra, SolicitudIndependiente.</p> <p>Para eliminar: se elimina una instancia de la clase Examen.</p>
Requerimientos especiales:	Para modificar y eliminar tiene que existir el examen.
Prototipo:	Anexo VII.

Caso de Uso del SISTEMA (5)	Definir Exámenes por Tipo de Unidad
Actores:	Personal de Medios de Diagnósticos (Inicia)
Descripción:	El Caso de uso se inicia cuando el Personal de MEDIOS DE DIAGNÓSTICOS accede al Registro de Medios de diagnósticos para asociar, eliminar la asociación de los Exámenes por Tipo de Unidad o imprimir esta información. Para asociar o eliminar la asociación solo puede acceder el Grupo Nacional de Expertos. El caso de uso finaliza cuando se actualiza o se imprime la información.
Referencias:	RF2, RF28. Tiene asociado el CU Buscar datos externos (Include).
Precondiciones:	Tiene que existir en la base de datos examen con su información gestionada previamente.
Poscondiciones:	Para asociar: se crea una relación entre la clase Examen y Tipo_Unidad_Examen. Para visualizar: se visualizan los datos de Examen. Para desasociar: se elimina la relación entre la clase Examen y Tipo_Unidad_Examen.
Requerimientos especiales:	Para visualizar y desasociar tiene que existir la relación.
Prototipo:	Anexo-VI.

CASO DE USO DEL SISTEMA (6)	BUSCAR EXAMEN
Actores:	Personal de Medios de Diagnósticos
Descripción:	El caso de uso se inicia cuando el personal de medios de diagnósticos necesita buscar exámenes. Dado un criterio de búsqueda devuelve los datos el (los) examen(s) correspondiente(s).
Referencias:	
Precondiciones:	Debe estar disponible y actualizada la información del Examen.
Poscondiciones:	-
Requerimientos especiales	-
Prototipo:	Anexo-VIII.

3.6.2 Fase Analítica.

CASO DE USO DEL SISTEMA (7)	GESTIONAR Métodos Analíticos
Actores:	Personal de Medios de Diagnósticos (Inicia)
Descripción:	El Caso de uso se inicia cuando el Personal de MEDIOS DE DIAGNÓSTICOS accede al Registro de Medios de diagnósticos para actualizar o consultar los Métodos Analíticos. Para actualizarlo solo puede acceder el Grupo Nacional de Expertos. El caso de uso finaliza cuando se actualiza o se imprime la información.
Referencias:	
Precondiciones:	-
Poscondiciones:	Se registra(n) lo(s) datos(s) de los Métodos Analíticos o se modifican los mismos. El sistema deja actualizada la información de los Sistemas Métodos Analíticos.
Requerimientos especiales:	Para modificar y eliminar tiene que existir el método.
Prototipo:	Anexo-VIII.

CASO DE USO DEL SISTEMA (8)	GESTIONAR SISTEMAS DE CALIBRACIÓN
Actores:	Personal de Medios de Diagnósticos (Inicia)
Descripción:	El Caso de uso se inicia cuando el Personal de MEDIOS DE DIAGNÓSTICOS accede al Registro de Medios de diagnósticos para actualizar o consultar los Sistemas de Calibración. Para actualizarlo solo puede acceder el Grupo Nacional de Expertos. El caso de uso finaliza cuando se actualiza o se imprime la información.
Referencias:	RF6
Precondiciones:	Debe haber sido gestionada previamente la información de los tipos de calibradores, unidades de medidas, fabricantes y marcas; las cuales deben existir en la base de datos.
Poscondiciones:	Se registra(n) lo(s) datos(s) del Sistema de Calibración o se modifican los mismos. El sistema deja actualizada la información de los Sistemas de Calibración.

Requerimientos especiales:	Para modificar y eliminar tiene que existir el sistema de calibración.
Prototipo:	Anexo-X

Caso de Uso del SISTEMA(9)	GESTIONAR REACTIVOS
Actores:	Personal de Medios de Diagnósticos (Inicia)
Descripción:	El Caso de uso se inicia cuando el Personal de MEDIOS DE DIAGNÓSTICOS accede al Registro de Medios de diagnósticos para actualizar o consultar los Reactivos. Para actualizarlo sólo puede acceder el Grupo Nacional de Expertos. El caso de uso finaliza cuando se actualiza o se imprime la información.
Referencias:	RF20 CU: Buscar reactivos (Extend).
Precondiciones:	El usuario debe haber sido autenticado al Sistema, sino el sistema le muestra al usuario un mensaje. Debe haber sido gestionada previamente la información de la presentación del reactivo, métodos de preparación, tipos de reactivos, fabricantes y marcas, y deben existir en la base de datos.
Poscondiciones:	Se registra(n) lo(s) datos(s) del Reactivo o se modifican los mismos. El sistema deja actualizada la información de los Reactivos.
Requerimientos especiales	Para modificar y eliminar tiene que existir el reactivo.
Prototipo:	Anexo-XII.

CASO DE USO DEL SISTEMA (10)	BUSCAR REACTIVO
Actores:	Personal de Medios de Diagnósticos
Descripción:	El caso de uso se inicia cuando el personal de medios de diagnósticos necesita buscar reactivos. Dado un criterio de búsqueda devuelve el (los) reactivo(s) correspondiente(s).
Referencias:	
Precondiciones:	Debe estar disponible y actualizada la información del Reactivo.
Poscondiciones:	-
Requerimientos especiales	-
Prototipo:	Anexo-XI.

CASO DE USO DEL SISTEMA (11)	GESTIONAR UNIDADES DE MEDIDA
Actores:	Personal de Medios de Diagnósticos. (Inicia)
Descripción:	El Caso de uso se inicia cuando el Personal de Medios de Diagnósticos accede al Registro de Medios de diagnósticos para actualizar o consultar la información de las Unidades de Medidas. Para actualizarlo solo puede acceder el Grupo Nacional de Expertos. El caso de uso finaliza cuando se actualiza o se imprime la información.
Referencias:	RF3
Precondiciones:	-
Poscondiciones:	Se registra(n) lo(s) datos(s) de las unidades de medidas o se modifican los mismos. El sistema deja actualizada la información de las unidades de medida.
Requerimientos especiales	Para modificar y eliminar tiene que existir la unidad de medida.
Prototipo:	Anexo-XIII.

3.6.3 Pos-Analítica.

CASO DE USO DEL SISTEMA (12)	GESTIONAR CODIFICADORES PARA RESULTADOS
Actores:	Personal de Medios de Diagnósticos (Inicia)
Descripción: El Caso de uso se inicia cuando el Personal de Medios de Diagnósticos accede al Registro de Medios de diagnósticos para actualizar o consultar la información de los siguientes codificadores, que se utilizarán para introducir los resultados de los exámenes: Tipos de Resultado, Claves de Rechazo, Diccionarios, Sistemas de diagnósticos y Subclasificación. Para actualizarlos solo puede acceder el Grupo Nacional de Expertos. El caso de uso finaliza cuando se actualiza o se imprime la información.	
Referencias:	RF10, RF13, RF19, RF21
Precondiciones:	Para gestionar las Subclasificaciones deben haber sido gestionados previamente los Sistemas de Diagnósticos.
Poscondiciones:	Se registra(n) el/los elemento(s) de los codificadores para resultados o se modifican los mismos. El sistema deja actualizada la información de estos codificadores.
Requerimientos especiales	-
Prototipo:	Anexo II.

3.6.4 Laboratorio.

CASO DE USO DEL SISTEMA (13)	Definir Reactivos por Laboratorio
Actores:	Jefe de laboratorio (Inicia)
Descripción: El Caso de uso se inicia cuando el Jefe de Laboratorio accede al Registro de Medios de diagnósticos para actualizar o consultar los Reactivos por Laboratorio. El caso de uso finaliza cuando se actualiza o se imprime la información.	
Referencias:	RF12 CU: Buscar datos externos (Incluye)

Precondiciones:	Debe estar disponible y actualizada la información en el Registro Unidades de Salud. Debe haber sido gestionada previamente la información de los reactivos.
Poscondiciones:	Se actualiza o se imprime la información de los exámenes por laboratorio.
Requerimientos especiales	-
Prototipo:	Anexo XV.

CASO DE USO DEL SISTEMA (14)	Definir Exámenes por Laboratorio
Actores:	Jefe de Laboratorio (Inicia)
Descripción:	El Caso de uso se inicia cuando el Jefe de Laboratorio accede al Registro de Medios de diagnósticos para actualizar o consultar los exámenes por laboratorios. El caso de uso finaliza cuando se actualiza o se imprime la información.
Referencias:	RF12 CU: Buscar datos externos (Incluye) CU: Buscar Exámenes (Incluye) CU: Definir métodos por exámenes del laboratorio (Extiende)
Precondiciones:	Debe estar disponible y actualizada la información en el Registro de Unidades de Salud. Debe haber sido gestionada previamente la información de los exámenes.
Poscondiciones:	Se actualiza o se imprime la información de los exámenes por laboratorio.
Requerimientos especiales	-
Prototipo:	Anexo XIV.

CASO DE USO DEL SISTEMA (15)	DEFINIR LABORATORIOS DE REFERENCIA
Actores:	Personal de Medios de Diagnósticos (Inicia)
Descripción:	

El Caso de uso se inicia cuando el Personal de Medios de Diagnósticos accede al Registro de Medios de diagnósticos para definir o consultar los Laboratorios de Referencia. Para definir solo puede acceder el Grupo Nacional de Expertos. El caso de uso finaliza cuando se actualiza o se imprime la información.	
Referencias:	RF12 CU: Buscar datos externos (Incluye)
Precondiciones:	Debe estar disponible y actualizada la información en el Registro de Unidades de Salud. Debe estar disponible y actualizada la información en el Registro de Áreas de Salud.
Poscondiciones:	Se actualiza o se imprime la información de los laboratorios de referencia.
Requerimientos especiales	-
Prototipo:	Anexo XVI.

CASO DE USO DEL SISTEMA (16)	LISTAR LABORATORIOS
Actores:	Personal de Medios de Diagnósticos (Inicia)
Descripción:	El Caso de uso se inicia cuando el Personal de Medios de Diagnósticos accede al Registro de Medios de diagnósticos para listar o consultar los Laboratorios que existen. Se puede consultar la información de los Exámenes, Reactivos y Métodos del laboratorio. El caso de uso finaliza cuando se visualiza o se imprime la información.
Referencias:	RF12
Precondiciones:	Debe estar disponible y actualizada la información en el Registro de Unidades de Salud. Debe estar disponible y actualizada la información en el Registro de Áreas de Salud. Debe haber sido gestionada previamente la información de los Laboratorios.
Poscondiciones:	Se imprime la información de los laboratorios.
Requerimientos especiales	-
Prototipo:	Anexo XVII.

CASO DE USO DEL SISTEMA(17)	Buscar Datos Externos
Actores:	Registro Unidades de Salud, Registro Áreas de Salud y Registro de Ubicación.
Descripción:	El caso de uso se inicia cuando el sistema necesita los datos externos, automáticamente el sistema se conecta al sistema externo y dado un criterio de búsqueda devuelve los datos correspondientes.
Referencias:	
Precondiciones:	Debe estar disponible y actualizada la información en los registros de: Unidades de Salud, Áreas de Salud y Ubicación.
Poscondiciones:	-
Requerimientos especiales	-

3.6.5 Configurar.

CASO DE USO DEL SISTEMA (18)	CONFIGURAR
Actores:	Personal de Medios de Diagnósticos (Inicia)
Descripción:	<p>El Caso de uso se inicia cuando el Personal de Medios de Diagnósticos accede al Registro de Medios de diagnósticos para actualizar o consultar la información de los siguientes codificadores: El Caso de uso se inicia cuando el Personal de Medios de Diagnósticos accede al Registro de Medios de diagnósticos para actualizar o consultar la información de los siguientes codificadores: Formas de expresar el resultado, Tipos de exámenes, Tipo de recipiente, Recipientes de conservación, Presentación del reactivo, Tipos de reactivos, Tipos de calibradores, Métodos de preparación del reactivo, Marcas y Fabricantes.</p> <p>Para actualizarlos sólo puede acceder el Grupo Nacional de Expertos. El caso de uso finaliza cuando se actualiza o se imprime la información.</p>
Referencias:	RF15, RF18, RF22, RF23, RF24, RF25, RF26, RF28
Precondiciones:	Para gestionar las marcas debe haber sido gestionada previamente la información de los fabricantes.

	Para gestionar los Recipientes de conservación debe haber sido gestionada previamente la información de los tipos de recipientes.
Poscondiciones:	Se registra(n) lo(s) elemento(s) de los codificadores o se modifican los mismos. El sistema deja actualizada la información de estos codificadores.
Requerimientos especiales	-
Prototipo:	Anexo XVIII.

3.7 Conclusiones.

En este capítulo se definieron los principales conceptos del dominio, que se tienen en cuenta para el desarrollo del sistema y se muestra el correspondiente modelo del dominio.

Se plasman los requerimientos del sistema, obteniéndose así las funcionalidades que debe tener la aplicación a desarrollar. Se definieron los paquetes del sistema y para cada uno de ellos se visualiza el diagrama de sus casos de uso. Se muestra el listado de los requerimientos no funcionales a tener en cuenta en el desarrollo de la aplicación.

CAPITULO 4 CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

El proceso iniciado en la etapa de análisis, que permite adentrarse en el problema a resolver para representar una vista interna del sistema con mayores refinamientos, se continúa en la etapa de diseño, que tiene el propósito de formular los modelos que se centran en los requisitos no funcionales y en el dominio de la solución y que prepara para las posteriores etapas de implementación y prueba del sistema en desarrollo.

En el presente capítulo se muestran los objetos que intervienen en los principales flujos de diseño: los diagramas de clases de los casos de uso definidos organizados por paquetes, el diagrama de clases persistentes, el modelo de datos obtenido, los estándares de codificación y el diagrama de despliegue como ilustración del modelo de implementación. Además se exponen los principios generales del diseño, dentro de ellos se presentan los estándares en la interfaz de la aplicación, el formato de los reportes, la concepción general de la ayuda y el tratamiento de excepciones.

4.1 Diagrama de clases del Diseño.

El diagrama de clases de diseño contienen los conceptos que modelan el comportamiento del sistema, las asociaciones entre ellos con todas las definiciones que requieren y los atributos que describen a cada clase identificada, todo esto heredado de la fase de análisis. En esta etapa se incluyen además las operaciones, parámetros, atributos, tipos, etc., necesario para la implementación en el lenguaje de programación seleccionado.

Las clases que se explican a continuación se utilizan en el diseño de la aplicación:

- **DBZ_class:** Clase que se encuentra en la Capa de Acceso a Datos y se utiliza para la conexión con bases de datos MySQL, usa el módulo dbx de PHP para su funcionalidad. Crea un objeto conexión que permite hacer consultas, y recuperar los resultados, insertar, eliminar y actualizar datos.
- **Fachada:** Clase general que se encuentra dentro del framework PLASER, que permite crear un encapsulamiento del mismo.
- **FachadaMD:** Clase que implementa el patrón de diseño Fachada, hereda de la clase Fachada disminuyendo así la carga del negocio.

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.[33]

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". [34]

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.

La aplicación está dividida en cinco paquetes lógicos que permiten una mejor visualización de las funcionalidades, para cada uno de los cuales se muestra su respectivo diagrama de clases del diseño.

4.1.1 Paquete Pre-Analítica.

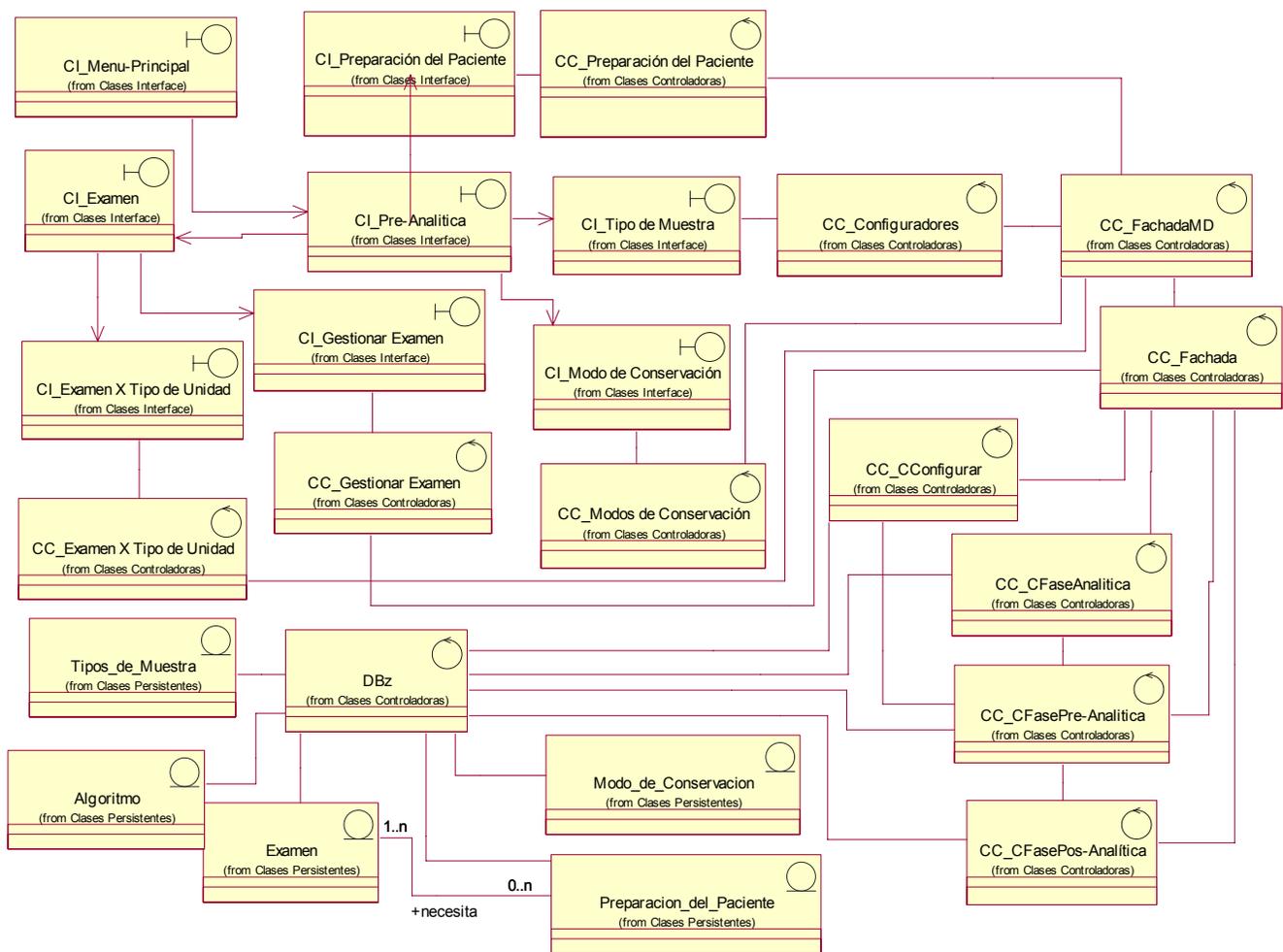


Figura 4.1 Diagrama de Clases del Diseño de la Fase Pre-Analítica.

4.1.2 Paquete Analítica.

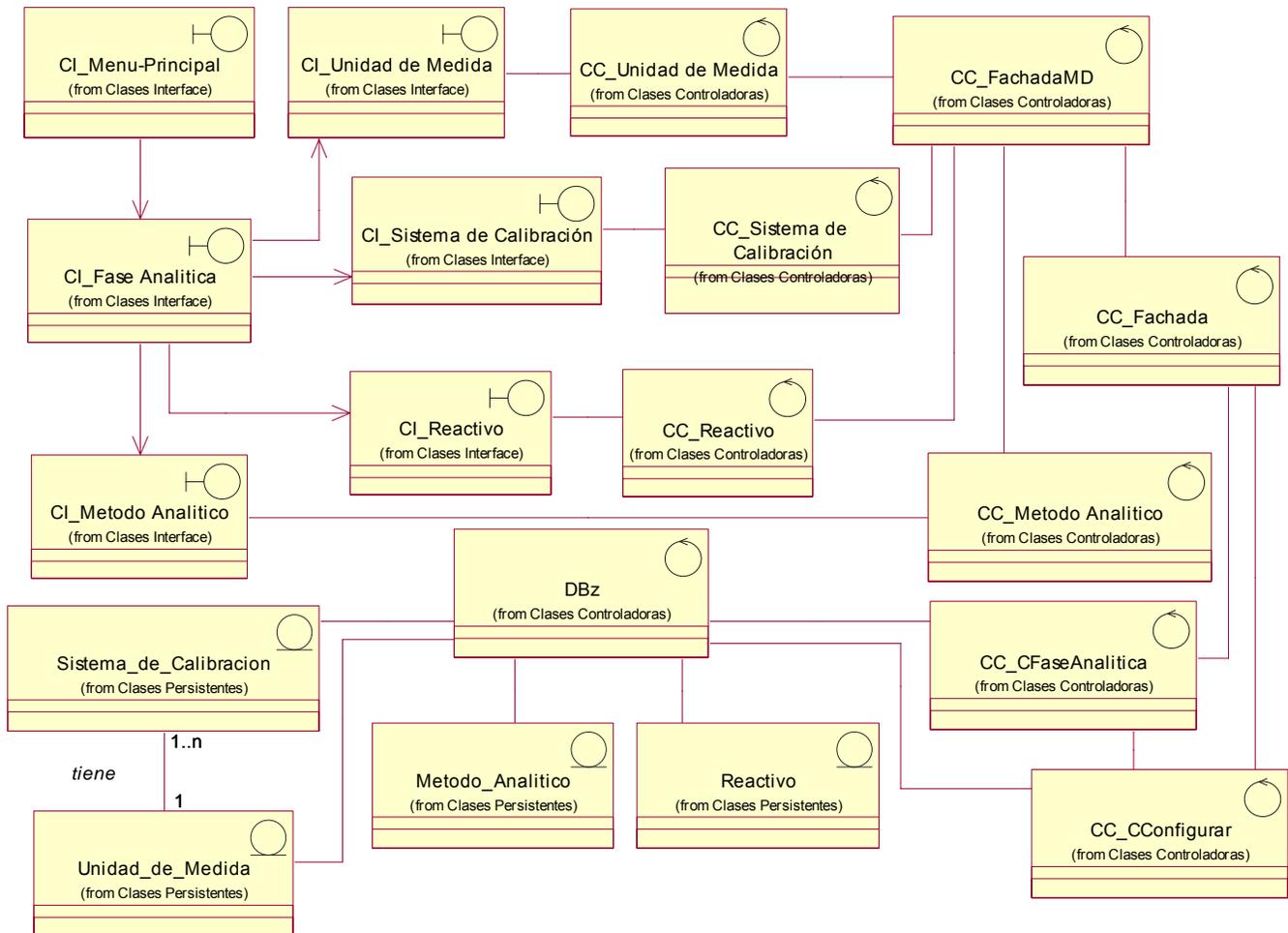


Figura 4.2 Diagrama de Clases del Diseño de la Fase Analítica.

4.1.3 Paquete Pos-Analítica.

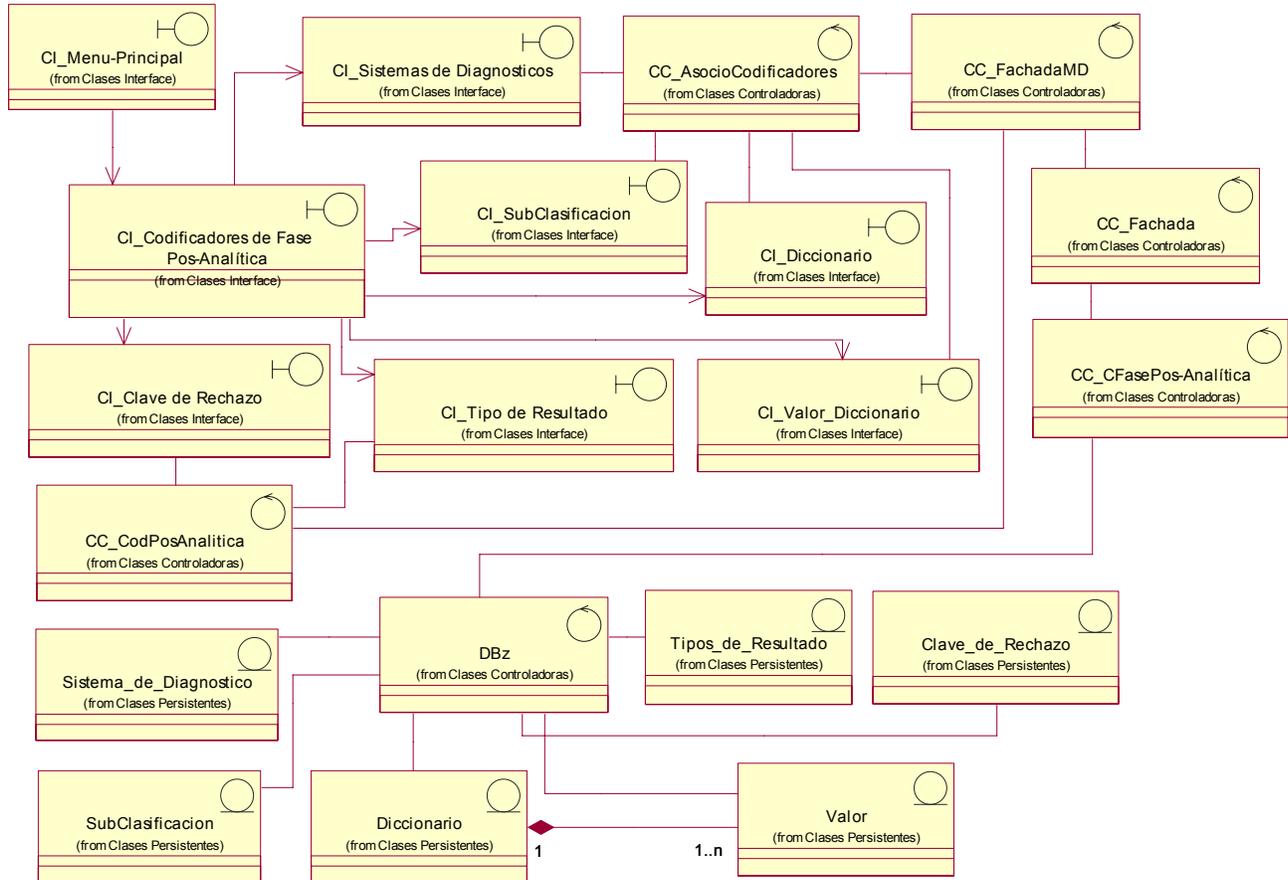


Figura 4.3 Diagrama de Clases del Diseño de la Fase Pos-Analítica.

4.1.4 Paquete Configurar.

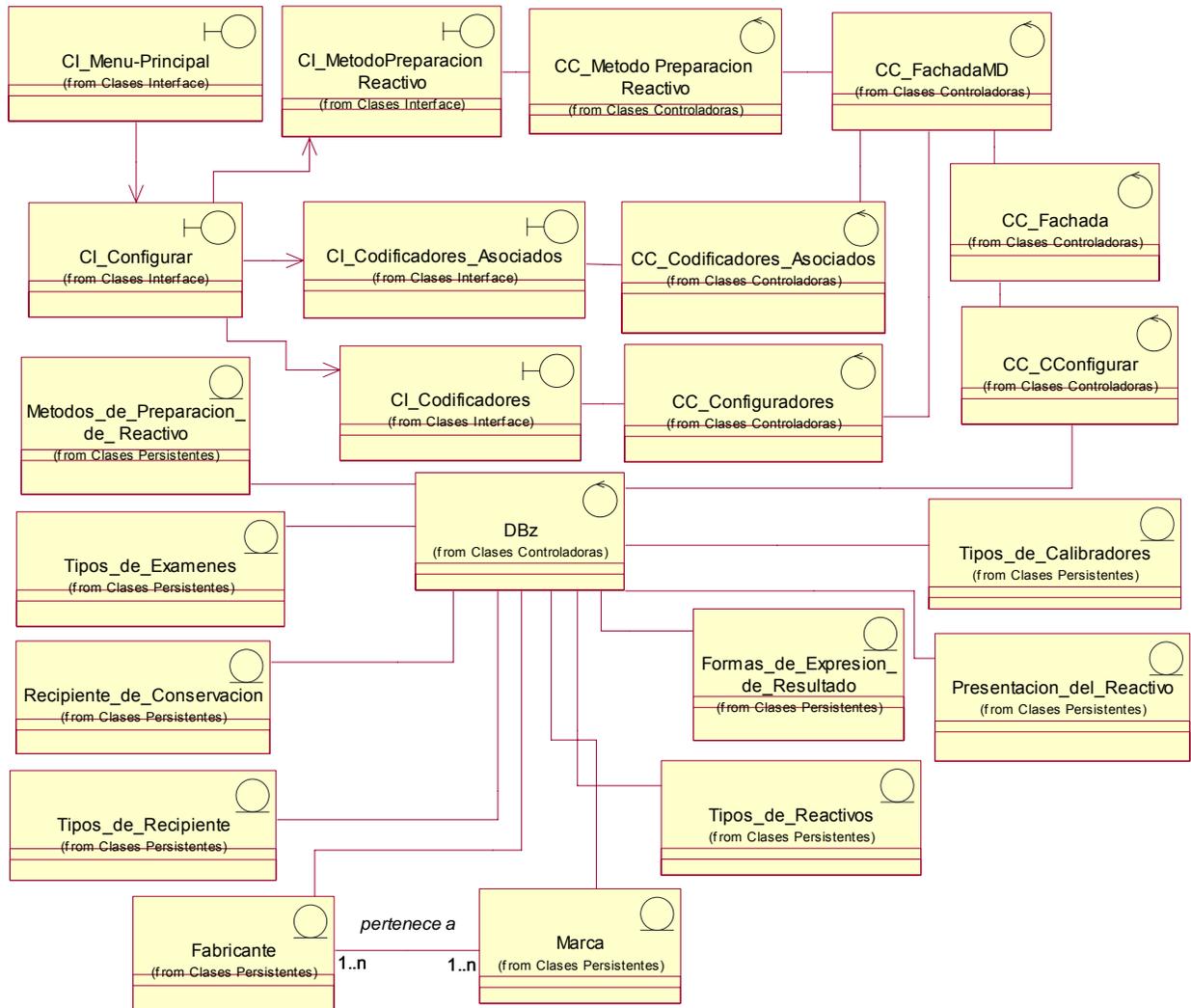


Figura 4.4 Diagrama de Clases del Diseño del Paquete Configurar.

4.1.5 Paquete Laboratorio.

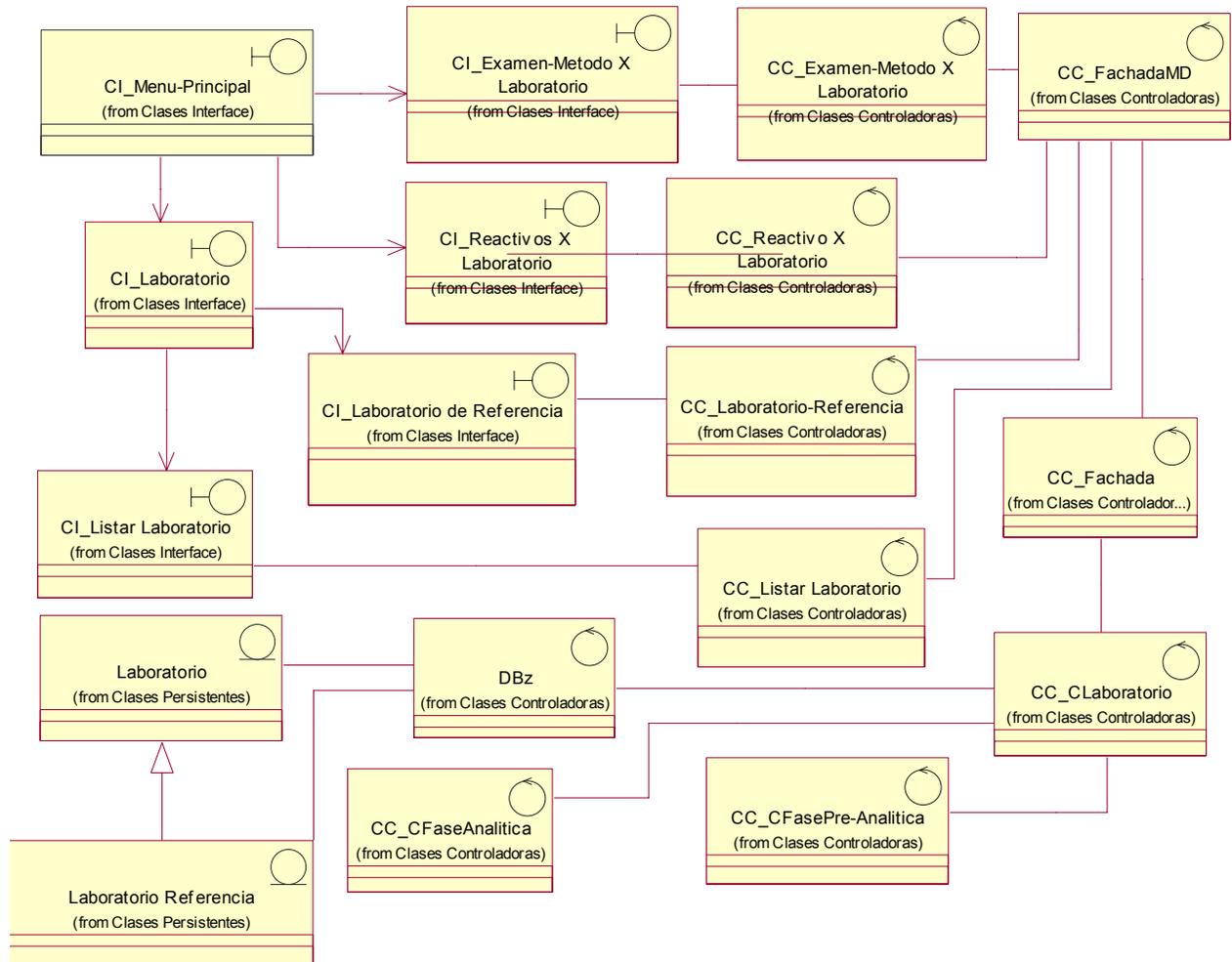


Figura 4.5 Diagrama de Clases del Diseño del Paquete Laboratorio.

4.2 Diseño de la base de datos.

La definición de la base de datos se concreta en decidir cuáles clases se desea que sean persistentes de las anteriormente definidas en el proceso de desarrollo del software.

4.2.1 Diagrama de clases persistentes.

Las clases persistentes son las clases que necesitan ser capaces de guardar su estado en un medio permanente, esta necesidad está dada por el almacenamiento físico permanente de la información de la clase, y para realizar la copia de seguridad en caso de fracaso del sistema, o solo para el intercambio de información.

El diagrama de clases persistentes se muestra a continuación en la figura 4.6.

4.2.2 Modelo de datos.

El modelo de datos visualiza la forma en que va a ser almacenada lógicamente y físicamente la información que tiene persistencia en el sistema a través de las tablas que van a conformar la base de datos y sus relaciones. También incluye cualquier conducta definida en esta, como los procedimientos almacenados, triggers, y constraints. Este modelo se crea a partir del diagrama de clases persistentes definido con anterioridad en el proceso.

El diagrama del modelo de datos se muestra a continuación en la figura 4.7.

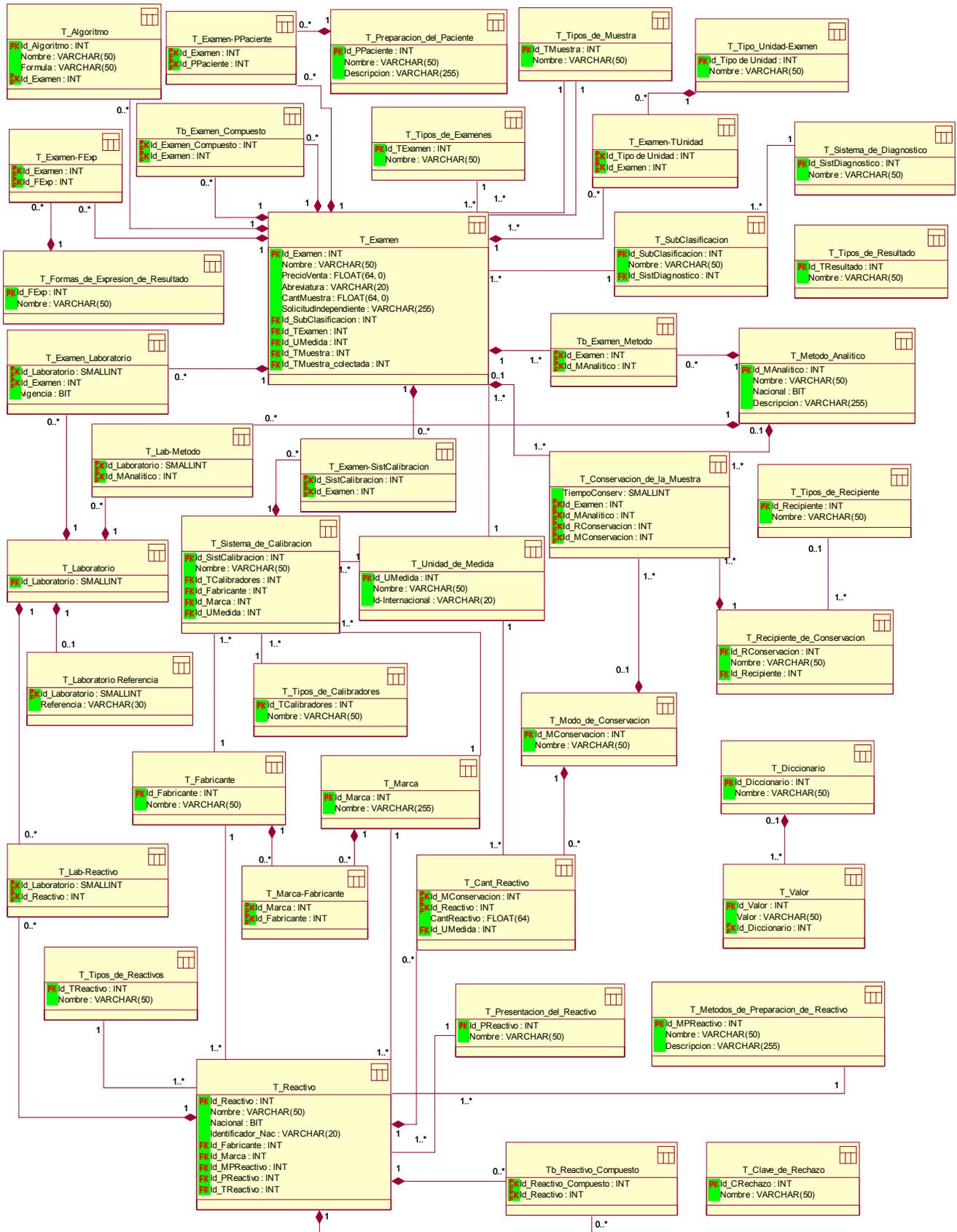


Figura 4.7 Modelo de Datos.

4.3 Principios de diseño.

4.3.1 Estándares en la interfaz de la aplicación.

Los módulos que se desarrollan como parte del Proyecto de APS corresponden a diferentes sistemas que mantienen una profunda interrelación, permitiendo un acoplamiento entre ellos para ofrecer las respuestas que demanda el negocio de la Atención Primaria propuesto por el Sistema Nacional de Salud. Este hecho tiene una repercusión determinante en la definición de la interfaz gráfica que se propone.

Todos los módulos están incluidos en un conjunto de aplicaciones que forman 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).

A continuación se presentan las pautas generales de diseño a tener en cuenta para este grupo de aplicaciones, ya que todas serán desarrolladas bajo el marco del Proyecto APS, según la distribución que se propone a continuación.

Diseño de Interfaz Gráfica del Proyecto APS

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 [ISO 9241-11].

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 édiu*m, 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.

4.3.2 Formato de reportes.

Los reportes se obtendrán en tablas que en algunos casos pueden tener una gran cantidad de elementos en dependencia de la información a visualizar, por lo que debe quedar pautado que hasta un total de 25 resultados la tabla funcione con scroll, y para más de esta cantidad será entonces por paginado, organizado por números consecutivos, con enlaces a los resultados restantes, exceptuando el activo en ese preciso momento.

En algunos casos se hará uso de pequeñas imágenes que indicarán funcionalidades surgidas a partir de la visualización de estos reportes.

Los reportes serán concebidos sobre ventanas, utilizando un formato de letra clara, legible y con colores claros para no recargar y hacer engorrosa su impresión.

4.3.3 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.

4.3.4 Tratamiento de excepciones.

Una excepción es un evento que ocurre durante la ejecución del programa que interrumpe el flujo normal de las sentencias. Son una forma clara para controlar los errores sin confundir el código con muchas instrucciones de control del error. Cuando se verifica un error se pone en marcha una excepción que, si se recibe enseguida, permite gestionar un error.

Las excepciones son condiciones excepcionales que pueden ocurrir dentro del programa durante su ejecución (por ejemplo una división por cero, que se agote la memoria disponible, que se pierda la comunicación, que no se produzca el resultado esperado ante alguna petición, etc.) y que requieren recursos especiales para su control.

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

Los errores en la capa de negocio serán tratados devolviendo un SOAP_FAULT, cuyos elementos FaultCode, FaultString, FaultActor describiremos a continuación:

FaultCode:

1. Código de texto utilizado para indicar la clase de error, codificado de la siguiente manera.
2. 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 Partos y Nacimientos perteneciente al Proyecto APS.

FaultString:

Una explicación del error asequible al humano (legible). 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.

Estos errores una vez en la capa de presentación, serán depurados mediante funciones del lenguaje *Client Side* Java Script, a través de mensajes de alerta.



Figura 4.8 Mensaje de Error.

4.4 Estándares de codificación.

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

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

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

Para nombrar las variables se seguirá la regla de escribir los identificadores con letras minúsculas y en español, utilizando como separador para las palabras el carácter “_” tratando de usar nombres sugerentes a la acción de la variable.

Todos los campos id van a comenzar con el identificador (id) seguido del nombre del campo. Ejemplo id _ examen.

Los arreglos empezarán con el identificador array y las palabras no se separaran con el carácter “_”. Ejemplo Arrayidtipo_de_examen.

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

4.5 Diagrama de despliegue.

Un diagrama de despliegue es un grafo de nodos unidos por conexiones de comunicación que muestra las relaciones físicas entre los componentes *hardware* que forman la topología sobre la que se ejecuta el sistema y la distribución de las partes de este en ellos.

La división entre cliente y servidor en un sistema es complicada ya que implica tomar algunas decisiones sobre dónde colocar físicamente sus componentes software, qué cantidad de software debe residir en el cliente, etc. Por lo que para modelarlo hay que identificar los nodos que representan los procesadores cliente y servidor del sistema de acuerdo a las capas que se van a implementar, destacar los dispositivos relacionados con el comportamiento del sistema, proporcionar señales visuales para esos procesadores y dispositivos a través de estereotipos y modelar la tipología de esos nodos mediante un diagrama de despliegue.

Aunque UML no es un lenguaje de especificación de hardware de propósito general, se ha diseñado para modelar muchos de los aspectos hardware de un sistema a un nivel suficiente para que un ingeniero de software pueda especificar la plataformas sobre la que se ejecuta el software del sistema y para que un ingeniero de sistemas pueda manejar la frontera entre el hardware y el software. El diagrama de despliegue se muestra a continuación en la figura 4.9.

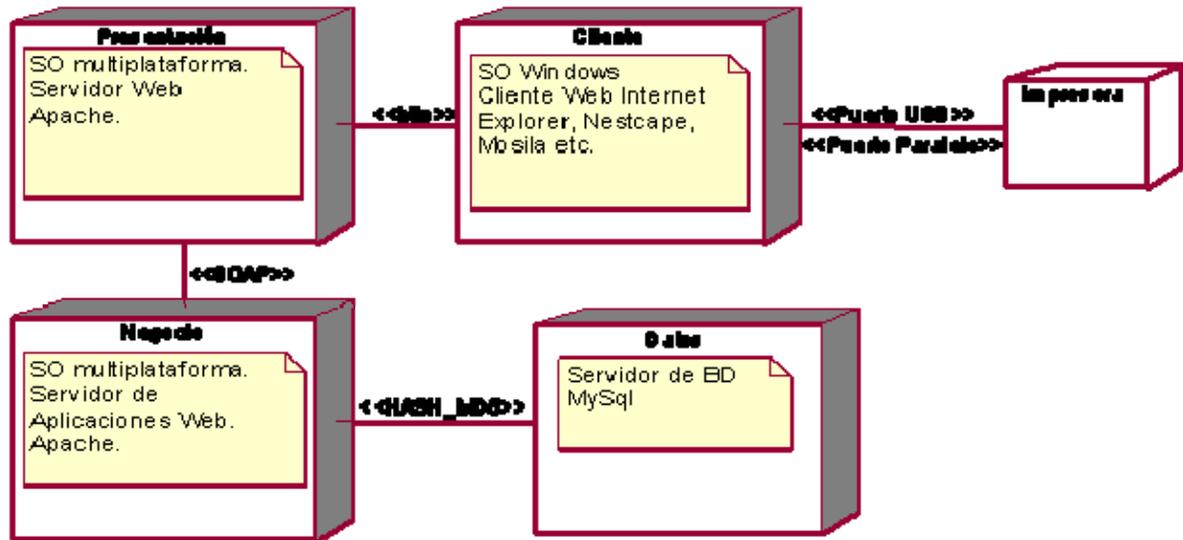


Figura 4.9 Diagrama de Despliegue.

4.6 Conclusiones.

Se ha logrado obtener un modelo bien detallado de la aplicación que se desea, Luego de transitar, durante el desarrollo de esta investigación, por las distintas etapas que propone RUP para el desarrollo de un software. De esta forma se le da solución a la situación problemática existente.

CAPITULO 5 ESTUDIO DE FACTIBILIDAD

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

Existen deferentes modelos para calcular la estimación del costo de un sistema. Actualmente el más difundido es el Modelo Constructivo de Costes (COCOMO), uno de los más documentados en la actualidad, de muy fácil utilización y por tanto aplicado a este sistema.

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. [35]

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 la preparación del paciente.	1	2	Simple
Modificar datos de la preparación del paciente.	1	2	Simple
Eliminar datos de la preparación del paciente.	1	2	Simple
Insertar Expresión del Resultado.	1	1	Simple
Modificar Expresión del Resultado.	1	1	Simple

Eliminar Expresión del Resultado.	1	1	Simple
Insertar Tipos de exámenes.	1	1	Simple
Modificar Tipos de exámenes.	1	1	Simple
Eliminar datos de Tipo de exámenes.	1	1	Simple
Insertar Tipos de Recipientes.	1	1	Simple
Modificar Tipos de Recipientes.	1	1	Simple
Eliminar Tipos de Recipientes.	1	1	Simple
Insertar datos de Recipiente de conservación.	1	2	Simple
Modificar datos de Recipiente de conservación.	1	2	Simple
Eliminar datos de Recipiente de conservación.	1	2	Simple
Insertar Presentación del Reactivo.	1	1	Simple
Modificar Presentación del Reactivo.	1	1	Simple
Eliminar Presentación del Reactivo.	1	1	Simple
Insertar Tipos de Reactivos.	1	1	Simple
Modificar Tipos de Reactivos.	1	1	Simple
Eliminar Tipos de Reactivos.	1	1	Simple
Insertar Tipos de Calibradores.	1	1	Simple
Modificar Tipos de Calibradores.	1	1	Simple
Eliminar Tipos de Calibradores.	1	1	Simple
Insertar Métodos de Preparación del Reactivo.	1	2	Simple
Modificar Métodos de Preparación del Reactivo.	1	2	Simple
Eliminar Métodos de Preparación del Reactivo.	1	2	Simple
Insertar Fabricantes.	1	1	Simple
Modificar Fabricantes.	1	1	Simple
Eliminar Fabricantes.	1	1	Simple
Insertar Marca.	1	2	Simple

Modificar Marca.	1	2	Simple
Eliminar Marca.	1	2	Simple
Insertar Claves de Rechazo.	1	1	Simple
Modificar Claves de Rechazo.	1	1	Simple
Eliminar Claves de Rechazo.	1	1	Simple
Insertar Tipos de Resultados.	1	1	Simple
Modificar Tipos de Resultados.	1	1	Simple
Eliminar Tipos de Resultados.	1	1	Simple
Insertar Diccionario.	1	1	Simple
Modificar Diccionario.	1	1	Simple
Eliminar Diccionario.	1	1	Simple
Insertar Valores del Diccionario.	1	2	Simple
Modificar Valores del Diccionario.	1	2	Simple
Eliminar Valores del Diccionario.	1	2	Simple
Insertar Sistemas de Diagnósticos.	1	1	Simple
Modificar Sistemas de Diagnósticos.	1	1	Simple
Eliminar Sistemas de Diagnósticos.	1	1	Simple
Insertar Subclasificación.	1	2	Simple
Modificar Subclasificación.	1	2	Simple
Eliminar Subclasificación.	1	2	Simple
Insertar Métodos Analíticos.	1	2	Simple
Modificar Métodos Analíticos.	1	2	Simple
Eliminar Métodos Analíticos.	1	2	Simple
Insertar datos del Reactivo.	1	6	Simple
Modificar datos del Reactivo.	1	6	Simple
Eliminar datos del Reactivo.	1	6	Simple
Insertar datos Sistemas de Calibración.	1	5	Simple
Modificar datos Sistemas de Calibración.	1	5	Simple
Eliminar datos Sistemas de Calibración.	1	5	Simple
Insertar Unidades de Medidas.	1	2	Simple
Modificar Unidades de Medidas.	1	2	Simple
Eliminar Unidades de Medidas.	1	2	Simple

Insertar Tipos de Muestras.	1	1	Simple
Modificar Tipos de Muestras.	1	1	Simple
Eliminar Tipos de Muestras.	1	1	Simple
Insertar datos de Modo de Conservación.	1	4	Simple
Modificar datos de Modo de Conservación.	1	4	Simple
Eliminar datos de Modo de Conservación.	1	4	Simple
Insertar datos del Examen.	1	10	Simple
Modificar datos del Examen.	1	10	Simple
Eliminar datos del Examen.	1	10	Simple
Asociar Examen por Tipo de Unidad.	1	2	Simple
Eliminar Asociación de Examen por Tipo de Unidad.	1	2	Simple
Asociar Examen al Laboratorio.	1	2	Simple
Eliminar Asociación de Examen del Laboratorio.	1	2	Simple
Asociar Reactivo al Laboratorio	1	2	Simple
Eliminar Asociación del Reactivo al Laboratorio.	1	2	Simple
Definir Laboratorio de Referencia.	1	2	Simple
Eliminar la Referencia del Laboratorio.	1	2	Simple
Insertar y Asociar al Examen Algoritmo.	1	3	Simple
Eliminar y eliminar Asociación Algoritmo.	1	2	Simple
Modificar Algoritmo.	1	2	Simple
Eliminar y eliminar asociación del Algoritmo del Examen.	1	3	Simple
Asociar Preparación del Paciente al Examen.	1	2	Simple
Eliminar Asociación de la Preparación del Paciente del Examen.	1	2	Simple
Asociar Método Analítico al Examen.	1	2	Simple
Eliminar Método Analítico del Examen.	1	2	Simple

Asociar Sistema de Calibración al Examen.	1	2	Simple
Eliminar Sistema de Calibración del Examen.	1	2	Simple
Insertar y Asociar Conservación de la Muestra.	1	5	Simple
Modificar Conservación de la Muestra.	1	4	Simple
Eliminar y eliminar asociación de Conservación de la Muestra del Examen.	1	5	Simple
Asociar Forma de Expresión del Resultado al Examen.	1	2	Simple
Eliminar Asociación de la Forma de Expresión del Resultado del Examen.	1	2	Simple
Componer Examen con Examen.	1	1	Simple
Eliminar composición del examen.	1	1	Simple
TOTAL	Simple: 97, Media: 0, Compleja: 0.		

Tabla 5.1 Entradas Externas.

Salidas Externas	Ficheros	Elementos de datos	Clasificación
Mostrar Preparación del paciente.	1	2	Simple
Mostrar datos de la Preparación.	1	2	Simple
Mostrar Modo de conservación.	1	1	Simple
Mostrar datos del Modo de conservación.	4	4	Media
Mostrar Tipo de Muestra.	1	1	Simple
Mostrar Método Analítico.	1	1	Simple
Mostrar datos de Método Analítico.	1	2	Simple
Mostrar Sistema de Calibración.	5	5	Media
Mostrar Unidades de Medida.	1	2	Simple
Mostrar Claves de Rechazo.	1	1	Simple
Mostrar Tipos de Resultados.	1	1	Simple
Mostrar Diccionarios.	1	1	Simple
Mostrar Diccionarios y sus valores.	2	2	Simple

Mostrar Sistema de Diagnósticos	1	1	Simple
Mostrar Sistema de Diagnósticos y sus subclasificaciones.	2	2	Simple
Mostrar Forma de Expresión del Resultado.	1	1	Simple
Mostrar Tipos de Exámenes.	1	1	Simple
Mostrar Tipos de Recipientes.	1	1	Simple
Mostrar Recipientes de Conservación.	2	2	Simple
Mostrar Presentación del Reactivo.	1	1	Simple
Mostrar Tipos de Reactivos.	1	1	Simple
Mostrar Tipos de Calibradores.	1	1	Simple
Mostrar Métodos de Preparación del Reactivo.	1	2	Simple
Mostrar Fabricantes.	1	1	Simple
Mostrar Marcas.	1	1	Simple
Mostrar datos de la Marca.	2	2	Simple
Mostrar Laboratorio y sus exámenes.	2	2	Simple
Mostrar datos del Examen.	6	10	Compleja
Mostrar Algoritmos del Examen.	1	2	Simple
Mostrar Preparación del Paciente del Examen.	1	2	Simple
Mostrar Métodos Analíticos del Examen.	1	2	Simple
Mostrar Sistemas de Calibración del Examen.	1	5	Simple
Mostrar Conservación de la Muestra del Examen.	1	4	Simple
Mostrar Forma de Expresión del Resultado del Examen.	1	1	Simple
Mostrar Exámenes que componen al examen.	1	1	Simple
TOTAL	Simple: 32, Media: 2, Compleja: 1.		

Tabla 5.2 Salidas Externas.

Peticiones	Ficheros	Elementos de datos	Clasificación
Buscar Examen.	6	8	Compleja
Buscar Reactivo.	7	7	Compleja
TOTAL	Simple: 0, Media: 0, Compleja: 2.		

Tabla 5.3 Peticiones.

Ficheros Lógicos Internos	Records	Elementos de datos	Clasificación
Preparación del paciente.	1	2	Simple
Tipos de Muestra.	1	1	Simple
Algoritmo.	1	2	Simple
Tipo de Examen.	1	1	Simple
Forma de expresión del resultado.	1	1	Simple
Sistema de Diagnósticos	1	1	Simple
Subclasificación.	1	2	Simple
Método Analítico.	1	2	Simple
Tipo de Recipiente.	1	1	Simple
Recipiente de Conservación.	1	2	Simple
Unidad de Medida.	1	2	Simple
Sistema de Calibración.	1	5	Simple
Tipos de Calibradores.	1	1	Simple
Examen.	1	10	Simple
Fabricante.	1	1	Simple
Marca	1	2	Simple
Reactivo.	1	7	Simple
Tipos de Reactivos.	1	1	Simple
Métodos de Preparación del Reactivo.	1	2	Simple
Presentación del Reactivo.	1	1	Simple
Modo de Conservación.	1	1	Simple
Diccionario	1	1	Simple
Valor del Diccionario	1	2	Simple

Tipos de Resultado	1	1	Simple
Clave de Rechazo	1	1	Simple
TOTAL	Simple: 25, Media: 0, Compleja: 0.		

Tabla 5.4 Ficheros Lógicos Internos.

Interfaz Externa	Records	Elementos de datos	Clasificación
Buscar datos externos.	1	5	Simple
TOTAL	Simple: 1, 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.	25	*7=175					175
Ficheros de interfaces externas	1	*5=5					5
Entradas externas.	97	*3=291					291
Salidas externas.	32	*4=128	2	*5=10	1	*7=7	155
Peticiones.					1	*6=6	9
TOTAL							635

Tabla 5.6 Puntos de Función sin ajustar.

Características	Valor		
Puntos de función desajustados.	635		
Lenguajes.	PHP	Java Script	SQL
% de utilización en la aplicación.	(85%)	(10%)	(5%)
Instrucciones fuentes por puntos de función.	60 (1)	58	39
Instrucciones fuentes por lenguaje (Miles de instrucciones fuentes).	32385	3556	1238.25
Instrucciones fuentes (Miles de instrucciones fuentes).	37179.25		

Tabla 5.7 Instrucciones fuentes.

1: Se utilizó el ratio del lenguaje Perl (60) pues el de PHP no se encontró y este es un lenguaje muy parecido.

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

Factores de escala	Valor
Precedencia (PREC)	1
Flexibilidad (FLEX)	3,04
Riesgos (RESL)	0.87
Cohesión del Equipo (TEAM)	0.83
Madurez de las Capacidades (PMAT)	1.33
ΣSF_i	12,24

Tabla 5.8 Definición de los valores de los Factores de Escala (Sfi).

Multiplicador	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
ME_j		1,09

Tabla 5.9 Definición de los Multiplicadores de Esfuerzo (Mej).

Valores Calibrados:

$$A = 2,94; B = 0,91; C = 3,67; D = 0,24$$

Cálculo del esfuerzo (PM).

$$E = B + 0,01 * \Sigma SFi = 0,91 + 0,01 * 12,24 = 1,03.$$

$$MSLOC = 37.179.$$

$$PM = A * (MSLOC)^E * \Pi ME_j = 2,94 * (37,179)^{1,03} * 1,09 = 2,94 * 41,43 * 1,09 = 132,76 \text{ hombres/mes.}$$

Tiempo de desarrollo (TDEV).

$$F = D + 0,2 * (E - B) = 0,24 + 0,2 * (1,03 - 0,91) = 0,264.$$

$$TDEV = C * PM^F = 3,67 * 132,76^{0,264} = 3,67 * 3,56 = 13 \text{ meses}$$

Cálculo de la cantidad de hombres.

$$CH = PM/TDEV = 132,76 / 13 = 18 \text{ hombres}$$

Ajustando para 5 hombres en el proyecto:

$$CH^* = 5 \text{ hombres.}$$

$$TEDV = PM/CH^* = 132,76 / 5 = 26.552 \text{ meses.}$$

Costo del proyecto

$$\text{SalarioE} = \$ 50,00.$$

$$\text{SalarioT} = \$ 225,00.$$

$$\text{SalarioP} = \$85.00$$

$$CHM = 5 * \text{SalarioP} = \$ 425.$$

$$C = CHM * PM = \$ 425 * 132,76 \text{ hombres/mes} = \$ 56 423.$$

Cálculo de:	Valor
Esfuerzo(PM: Hombres – mes)	132,761
Tiempo estimado de desarrollo (meses)	13
Cantidad estimada de hombres	18
Para 5 hombres cantidad de meses estimada	26
Salario medio real (pesos)	\$ 85,00
Costo real (pesos)	\$56 423.

Tabla 5.8 Cálculo del esfuerzo, tiempo de desarrollo, cantidad de hombres y costo.

5.3 Beneficios tangibles e intangibles.

Hasta esta etapa el Registro de Medios de Diagnósticos está pensado para su uso dentro del Sistema Nacional de Salud. Por la gran importancia que tiene el software para la salud como renglón exportable en el país, su inserción en el mercado internacional bajo la licencia de software libre, principalmente en el mercado latinoamericano, contribuirá a que los beneficios tangibles sean materializados e invertidos incluso en el posterior desarrollo del sistema y del país en general.

Como consecuencia del diseño de la aplicación también será disminuida la carga basada en papeles que significa el almacenamiento de toda la información que se necesita. Se dispondrá de un sistema automatizado para esta gestión que posibilitará una mayor disponibilidad de la información en todos los niveles y un mayor control, seguridad y confiabilidad de los datos. Todo esto será revertido en una mejor atención a los pacientes y un mayor control de la actividad en los laboratorios clínicos y de microbiología.

5.4 Análisis de costos y beneficios.

El costo en el desarrollo de un sistema informático puede estar justificado por los beneficios que aporte al proceso para el cual fue creado. Este sistema centraliza toda la información referente a los laboratorios clínicos y de microbiología del país lo que posibilita un mayor control a nivel nacional de las instituciones con este servicio y una estandarización y homogenización de la información que es compartida por ellos a lo largo de la Isla, permitiendo el proceso de toma de decisiones más oportuno y eficiente.

Por la forma de desarrollo que tiene este sistema, en colaboración de la UCI con la empresa SOFTEL encargada de su confección, la inversión real que requiere no es alta y sin embargo sí lo es, el aporte que constituye al Sistema Nacional de Salud para una atención de mayor calidad a los ciudadanos y su contribución a la informatización de la sociedad cubana.

5.5 Conclusiones.

Una vez terminado el estudio de factibilidad del sistema, se estima un tiempo de 26 meses para su construcción por 5 hombres y su costo asciende a \$56 423.00. No caben dudas de que una óptima planificación del proyecto contribuye a un desarrollo organizado y por tanto a una mejor calidad del producto obtenido.

No sólo se comprendió el ámbito del trabajo a realizar, los recursos que se necesitan y las tareas específicas a ejecutar, sino que además se tuvo en cuenta el costo y la factibilidad ya que esto puede influir grandemente en la aceptación y desarrollo del sistema deseado.

Además de realizar los cálculos manualmente, se utilizó el Demo COCOMO II, que es un sistema para el cálculo de todas las variables planteadas por Boehm, y así se pudieron corroborar todos los datos obtenidos.

CONCLUSIONES

Con el presente trabajo se ha concluido el desarrollo del Registro de Medios de Diagnósticos para el Registro Informatizado de Salud (Laboratorios Clínicos y de Microbiología), en sus fases de análisis y diseño, dando cumplimiento a los objetivos general y específicos planteados anteriormente; logrando:

1. Definir los conceptos objeto de automatización, una vez estudiado el negocio de este registro.
2. Definir la arquitectura de la aplicación, así como el lenguaje de programación y el sistema gestor de base de datos.
3. Modelar un sistema que permita gestionar el registro de los medios de diagnósticos de los laboratorios clínicos y de microbiología de forma tal, que facilite la estandarización de la información a nivel nacional, su consulta en los diferentes niveles de salud, la gestión de la información de cada laboratorio y la toma de decisiones.
4. Diseñar una base de datos que permita el almacenamiento eficiente de la información de los medios de diagnósticos y los laboratorios de todo el país.

RECOMEDACIONES

Se recomienda terminar el primer ciclo de desarrollo del sistema con la implementación del Registro de Medios de Diagnósticos para el Registro Informatizado de Salud (Laboratorios Clínicos y de Microbiología), a partir de los resultados obtenidos al modelar la aplicación.

Se debe continuar la etapa de implementación como está prevista en la organización del Proyecto APS, logrando la incorporación de los estudiantes de tercer año de la UCI para la culminación del desarrollo y su posterior implantación piloto en las unidades de salud del país.

REFERENCIAS BIBLIOGRAFICAS

- [1] 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
- [2] José A. de la Osa, *La semilla del desarrollo de la salud pública en Cuba*.
- [3] Escuela Nacional de Salud Pública, *El cuidado de la salud en Cuba. Ministerio de Salud Pública*, 2003.
- [4] 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.
- [5] Granma, *Discurso pronunciado en la Clausura del VI Seminario Internacional de Atención Primaria, Ciudad de la Habana*, 28 de noviembre de 1997.
- [6] Fidel Castro Ruz. Discurso en el acto de inauguración de obras del extraordinario programa de salud. Teatro Astral. 7 de abril del 2003
<http://www.granma.cu/documento/espanol03/014.html>
- [7] Ramírez Márquez, Abelardo *El Sistema Nacional de Salud de Cuba*. / Pastor Castell-Florit Serrate; Guillermo Mesa. ENSAP, 2003.
- [8] "Hitos y mitos".
<http://www.el-planeta.com/modem/hist3.htm> (03/03/05)
- [9] "Internet; ¿Que es Internet?".
<http://cbtis18bezai.galeon.com/aficiones1182987.html> (03/03/05)
- [10] Manual de Introducción a Internet. Segunda Parte.
http://www.mayckmarck.com/Manual_Internet_2.htm (03/03/05)
- [11] "Orígenes de Internet: ARPANET".
http://html.rincondelvago.com/origenes-de-internet_arpanet.html (03/03/05)
- [12] Principales definiciones de los términos más usados en Internet.
<http://www.informaticamilenium.com.mx/paginas/espanol/sitioweb.htm> (03/03/05)
- [13] XML: ¿Otro Acrónimo Mas?.
<http://www.dcc.uchile.cl/~rbaeza/inf/xml.html> (03/03/05)
- [14] XML.
<http://es.wikipedia.org/wiki/XML> (03/03/05)
- [15] Cliente-Servidor.

<http://es.wikipedia.org/wiki/Cliente-servidor> (03/03/05)

[16] Diseño de Aplicaciones Three Tier.

<http://www.fpess.com/revista/Num9711/Nov97.htm> (03/03/05)

[17] El modelo de tres capas o "Three Tier".

http://www.lagash.com/papers/paper_threetier.html (03/03/05)

[18] El Servidor Web Apache.

<http://es.tldp.org/Tutoriales/doc-servir-web-escuela/doc-servir-web-escuela-html/apache.html>

(03/03/05)

[19] Desarrollo de n-capas.

<http://html.rincondelvago.com/desarrollo-de-n-capas.html> (03/03/05)

[20] Diseñando Aplicaciones Distribuidas.

<http://www.monografias.com/trabajos14/aplicacion-distrib/aplicacion-distrib.shtml> (03/03/05)

[21] PERL.

<http://es.wikipedia.org/wiki/Perl> (03/03/05)

[22] ASP.

<http://es.wikipedia.org/wiki/ASP> (03/03/05)

[23] PHP.

<http://es.wikipedia.org/wiki/PHP> (03/03/05)

[24] Qué es PHP.

<http://64.233.161.104/search?q=cache:BYvi4R0KuY0J:elvallenato.com/sac/index2.php%3Foptio>
[n%3Dcom_content%26do_pdf%3D1%26id%3D23+capacidades+de+PHP+&hl=es&lr=lang_es](http://64.233.161.104/search?q=cache:BYvi4R0KuY0J:elvallenato.com/sac/index2.php%3Foptio)

(03/03/05)

[25] ¿Porqué elegir PHP?

<http://www.abiertos.org/modules.php?name=News&file=article&sid=1736> (03/03/05)

[26] Java Server Pages.

<http://es.wikipedia.org/wiki/JSP> (03/03/05)

[27] JavaScript.

<http://es.wikipedia.org/wiki/JavaScript> (03/03/05)

[28] XSLT.

<http://es.wikipedia.org/wiki/XSLT> (03/03/05)

[29] Sistemas Gestores de Bases de Datos.

http://es.wikipedia.org/wiki/Sistemas_Gestores_de_Bases_de_Datos (03/03/05)

[30] Instalación y configuración de Microsoft SQL Server 2000.

<http://usuarios.lycos.es/cursosgbd/UD5.htm> (03/03/05)

[31] MySQL.

<http://www.software-shop.com/Productos/MySQL/mysql.html> (03/03/05)

[32] PostgreSQL.

<http://www.google.com.cu/url?sa=X&start=0&oi=define&q=http://es.wikipedia.org/wiki/PostgreSQL> (03/03/05)

[33] Desarrollo de aplicaciones WEB con UML.

http://www.vico.org/TRAD_obert/TRAD_WAE_abierto.html#anchor (15/03/2005)

[34] UML para proyectos Web, http://www.cadius.org/pipermail/lista_cadius.org/2003-August/000204.html (10/05/2005)

[35] Peralta, M. *Estimación del esfuerzo basada en casos de usos*. Reportes Técnicos en Ingeniería del Software. ISSN 1667-5002. Volumen 6, Nro 1, Pág. 1-16. 2004

<http://www.itba.edu.ar/capis/webcapis/planma-esp.html> (22/05/05)

GLOSARIO DE TERMINOS

ASP (*Active Server Pages*): Es una tecnología dinámica basada en el lenguaje Visual Basic con sentencias SQL. Depende del servidor donde se hospede, puesto que sólo serán válidos los que se basan en tecnología NT de Microsoft utilizando a su vez el programa que gestiona las funciones de Visual Basic, conocido como *IIS (Internet Information Server)*.

Dreamweaver MX: Herramienta fácil para crear y editar páginas compatibles con cualquier explorador y plataforma. Además de ser un editor visual profesional para la creación y administración de sitios y aplicaciones Web.

COCOMO (*Constructive Cost Model*): Modelo que permite estimar el coste, esfuerzo y tiempo cuando se planifica una nueva actividad de desarrollo de software.

HTML (*Hypertext Markup Language*): Lenguaje utilizado para escribir documentos para servidores World Wide Web. Es una aplicación de la ISO Standard 8879:1986.

HTTP (*Hypertext Transport Protocol*): Es un conjunto de especificaciones para el intercambio de ficheros (texto, gráfico, imagen, sonido, video) en la Web.

Internet: Sistema de redes de computación ligadas entre sí, con alcance mundial, que facilita servicios de comunicación de datos como registro remoto, transferencia de archivos, correo electrónico y grupos de noticias. Internet es una forma de conectar las redes de computación existentes que amplía en gran medida el alcance de cada sistema participante.

JSP (*Java Server Pages*): Es una tecnología basada en Java que simplifica el desarrollo de páginas web con contenido dinámico.

Microsoft SQL Server: Es un sistema gestor de bases de datos relacionales

Perl (*Practical Extracting and Reporting Language*): Es un lenguaje de programación desarrollado por Larry Wall, inspirado en otras herramientas de UNIX como son: sed, grep, awk, c-shell, para la administración de tareas propias de sistemas UNIX.

PHP (*Professional Home Pages*): Es un ambiente script del lado del servidor que permite crear y ejecutar aplicaciones Web dinámicas e interactivas. Con PHP se pueden combinar páginas HTML y scripts. Con el objetivo de crear aplicaciones potentes. Se caracteriza por ser tecnología (open source), y por su fácil desarrollo y mantenimiento.

RUP (*Rational Unified Process*): Metodología para el desarrollo de Software.

Sitio Web: Sistema de computación que corre en un servidor Web y que se ha establecido para editar documentos en Web.

Software: Programas de sistema, utilerías o aplicaciones expresados en un lenguaje de maquina.

ANEXOS

ÍNDICE

INTRODUCCIÓN.....	1
CAPÍTULO 1 FUNDAMENTACIÓN DEL TEMA	5
1.1 SISTEMA NACIONAL DE SALUD.	5
1.1.1 <i>Informatización del Sistema Nacional de Salud.</i>	6
1.1.2 <i>Informatización de la Atención Primaria de Salud.</i>	9
1.1.3 <i>Registro Informatizado de Salud (RIS).</i>	10
1.1.4 <i>Solución integral propuesta para la Informatización del SNS.</i>	10
1.2 CONCEPTOS ASOCIADOS AL DOMINIO DEL PROBLEMA.	12
1.3 OBJETO DE ESTUDIO.	12
1.3.1 <i>Descripción general.</i>	12
1.3.2 <i>Descripción actual.</i>	13
1.4 CONCLUSIONES.	13
CAPITULO 2 TENDENCIAS Y TECNOLOGÍAS ACTUALES.....	14
2.1 ¿QUÉ ES INTERNET?.....	14
2.1.1 <i>¿Cómo funciona Internet?</i>	16
2.1.2 <i>La información a través de Internet. El World Wide Web.</i>	17
2.2 APLICACIONES WEB VS. SITIOS WEB.	17
2.3 TECNOLOGÍA XML/WEBSERVICES (SOA).....	18
2.4 ENTORNOS DISTRIBUIDOS. MODELO CLIENTE SERVIDOR.	19
2.4.1 <i>Modelo Cliente Servidor de Dos Capas (Two Tier).</i>	20
2.4.2 <i>Modelo Cliente Servidor de tres Capas (Three Tier).</i>	20
2.4.3 <i>Consideraciones sobre el hardware de la red y el software en los entornos cliente/servidor.</i>	21
2.5 SERVIDOR WEB APACHE.	21
2.5.1 <i>Arquitectura Basada en Componentes (CBA).</i>	22
2.6 LENGUAJES DE PROGRAMACIÓN WEB.....	22
2.6.1 <i>PERL (Practical Extracting and Reporting Language).</i>	23
2.6.2 <i>ASP (Active Server Pages).</i>	23
2.6.3 <i>PHP (Personal Home Page).</i>	24
2.6.4 <i>JSP (Java Server Pages).</i>	25
2.6.5 <i>JAVA SCRIPT.</i>	26

2.6.6 XSLT.....	26
2.6.7 ¿Por qué PHP y XSLT?.....	26
2.7 SISTEMAS DE GESTIÓN DE BASES DE DATOS (SGBD).	26
2.7.1 SQL SERVER.....	27
2.7.2 MySQL.....	28
2.7.3 POSTGRESQL.....	28
2.8 DESARROLLO BASADO EN RUP BAJO LA HERRAMIENTA RATIONAL ROSE.	29
2.8.1 UML (<i>Unified Modeling Language</i>).	29
2.8.2 <i>Rational Rose</i>	30
2.9 PLATAFORMA DE SERVICIO (PLASER).....	31
2.10 HERRAMIENTAS A UTILIZAR.	32
2.11 CONCLUSIONES.	32
CAPÍTULO 3 DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA.....	33
3.1 DESCRIPCIÓN DE LOS PROCESOS DEL NEGOCIO PROPUESTOS.	33
3.2 CONCEPTOS DEL DOMINIO.....	33
3.3 REQUERIMIENTOS FUNCIONALES.	37
3.4 REQUERIMIENTOS NO FUNCIONALES.	40
3.5 DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA.	43
3.6 DIAGRAMAS DE CASOS DE USO DEL SISTEMA.	46
3.6.1 <i>Fase Pre-Analítica</i>	48
3.6.2 <i>Fase Analítica</i>	52
3.6.3 <i>Pos-Analítica</i>	55
3.6.4 <i>Laboratorio</i>	55
3.6.5 <i>Configurar</i>	58
3.7 CONCLUSIONES.	59
CAPITULO 4 CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA.....	60
4.1 DIAGRAMA DE CLASES DEL DISEÑO.....	60
4.1.1 <i>Paquete Pre-Analítica</i>	62
4.1.2 <i>Paquete Analítica</i>	63
4.1.3 <i>Paquete Pos-Analítica</i>	64
4.1.4 <i>Paquete Configurar</i>	65
4.1.5 <i>Paquete Laboratorio</i>	66
4.2 DISEÑO DE LA BASE DE DATOS.	67

4.2.1 Diagrama de clases persistentes.....	67
4.2.2 Modelo de datos.....	67
4.3 PRINCIPIOS DE DISEÑO.....	70
4.3.1 Estándares en la interfaz de la aplicación.....	70
4.3.2 Formato de reportes.....	73
4.3.3 Concepción general de la ayuda.....	73
4.3.4 Tratamiento de excepciones.....	75
4.4 ESTÁNDARES DE CODIFICACIÓN.....	76
4.5 DIAGRAMA DE DESPLIEGUE.....	77
4.6 CONCLUSIONES.....	78
CAPITULO 5 ESTUDIO DE FACTIBILIDAD.....	79
5.1 PLANIFICACIÓN.....	79
5.2 CÁLCULO DEL ESFUERZO, TIEMPO DE DESARROLLO, CANTIDAD DE HOMBRES Y COSTO.....	87
5.3 BENEFICIOS TANGIBLES E INTANGIBLES.....	89
5.4 ANÁLISIS DE COSTOS Y BENEFICIOS.....	89
5.5 CONCLUSIONES.....	90
CONCLUSIONES.....	91
RECOMEDACIONES.....	92
REFERENCIAS BIBLIOGRAFICAS.....	93
GLOSARIO DE TERMINOS.....	96
ANEXOS.....	97



INSTITUTO SUPERIOR POLITÉCNICO "JOSÉ ANTONIO ECHEVERRÍA"
FACULTAD DE INGENIERÍA INDUSTRIAL
CENTRO DE ESTUDIOS DE INGENIERÍA Y SISTEMA

**REGISTRO DE MEDIOS DE DIAGNÓSTICOS
PARA EL REGISTRO INFORMATIZADO DE SALUD
(Laboratorios Clínicos y de Microbiología)**

Trabajo de diploma para optar por el título de Ingeniería Informática

Autor: Luis Noa Leyva.

Tutor: Ing. Yamilka Gómez León

Consultante: Lic. Caridad Guzmán Vitón

Ciudad de La Habana

Junio del 2005

DECLARACIÓN DE AUTORÍA

Declaro que soy el único autor de este trabajo y autorizo a la Empresa SOFTEL, perteneciente al Ministerio de Informática y Comunicaciones (MIC), a que haga el uso que estime pertinente con este trabajo.

Para que así conste firmo la presente a los 23 días del mes de junio del 2005.

Firma del Autor

Firma del Tutor

AGRADECIMIENTOS

“”A todos que de una forma u otra me ayudaron a lo largo de toda mi carrera.

...hay que gozar la fiesta del presente, las
lecciones del pasado y los sueños del futuro...

PauloCoelho

RESUMEN

En la actualidad las posibilidades que ofrece el creciente desarrollo de Internet han hecho posible que se lleven a cabo una serie de transformaciones en la informatización de diferentes procesos. Nuestro país, por su parte se encuentra actualmente inmerso en un proceso de informatización de la sociedad fomentando el uso de la informática en los diferentes sectores, llevándose a cabo el “Programa Rector para la Informatización de la Sociedad Cubana”. El sector de la salud ha propuesto numerosas soluciones que mejoran la calidad de los servicios y que involucran las nuevas tecnologías.

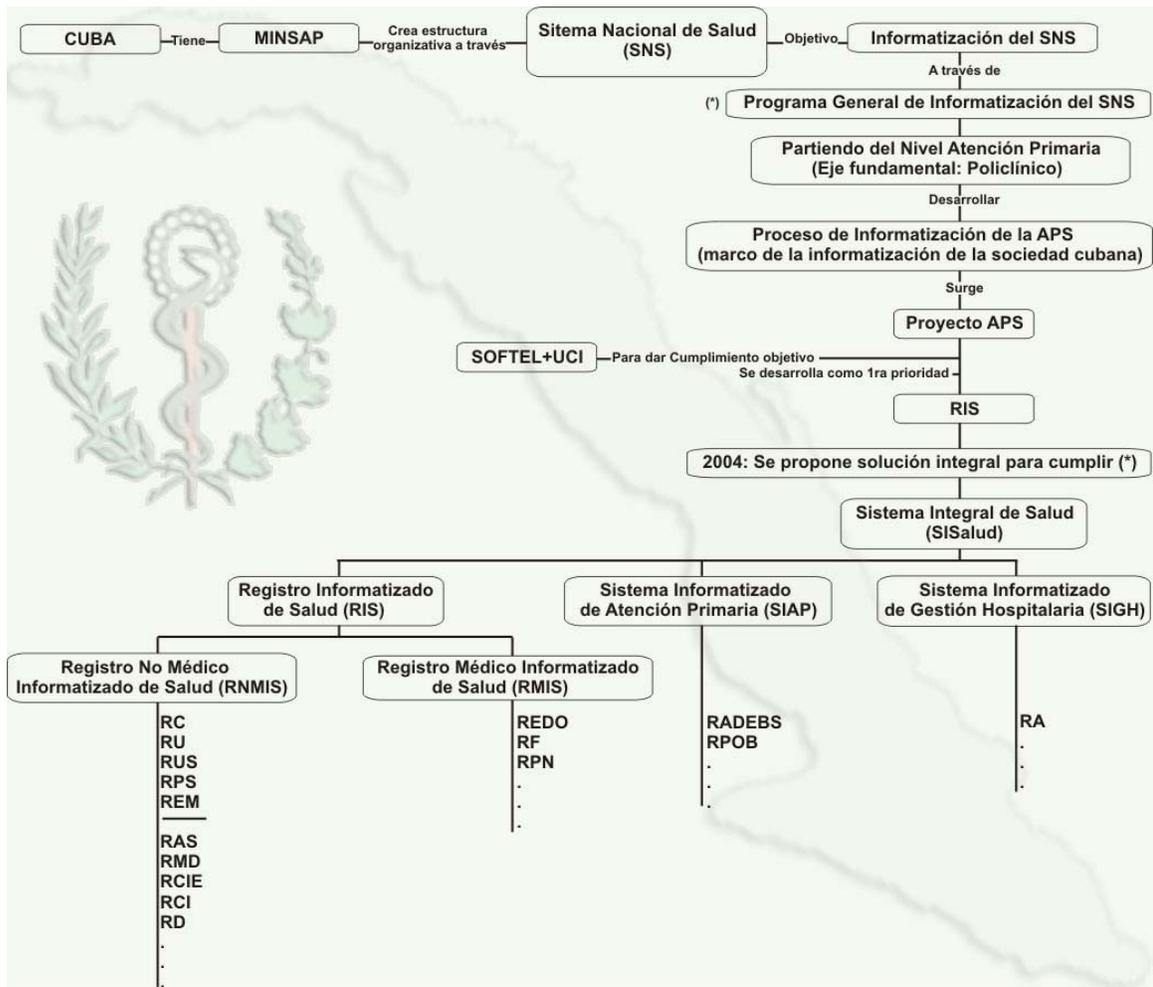
Como resultado del proceso de informatización de la salud cubana, la Empresa SOFTEL en conjunto con la UCI tienen la misión de desarrollar “Sistema Integral de Salud” (SIS), el cual permitirá mejorar los servicios que se brindan a la población, un mejor desempeño de sus profesionales y la obtención de estadísticas de calidad a nivel nacional.

De esta forma surge el Registro Informatizado de Salud (RIS) por la necesidad de contar con registros de información estándar a nivel nacional que facilite la obtención de estadísticas a todos los niveles y la toma de decisiones.

El Registro de Medios de Diagnósticos para el Registro Informatizado de Salud (Laboratorios Clínicos y de Microbiología) tiene como objetivos modelar un sistema informático capaz de registrar y mantener actualizados los codificadores nacionales de los medios de diagnósticos de los laboratorios clínicos y de microbiología, logrando una información estándar de los mismos en todos los niveles de atención de salud y será el contenido fundamental tratado en el presente trabajo.

Se realizó para ello el modelado de la aplicación utilizando el Proceso Unificado de Desarrollo RUP (Rational Unified Process) y el Lenguaje Unificado de Modelación (UML).

ANEXOS



Anexo- I

Listado general - <<Pos-Analítica>>

<input type="button" value="Imprimir"/>	<input type="button" value="Agregar"/>	<input type="button" value="Eliminar"/>
Nombre		
<input type="radio"/>	Muestra contaminada	
<input type="radio"/>		
1 - 1/1		
<input type="button" value="Imprimir"/>	<input type="button" value="Agregar"/>	<input type="button" value="Eliminar"/>

Anexo-II GESTIONAR CODIFICADORES PARA RESULTADOS.

Editar - PREPARACION DEL PACIENTE

<input type="button" value="Agregar"/>		<input type="button" value="Eliminar"/>	
Nombre	<input type="text" value="Ayuna"/>	*	
Descripción	<input type="text" value="el paciente debe....."/>	*	
(*) Los campos señalados son de entrada obligatoria.			
		<input type="button" value="Aceptar"/>	<input type="button" value="Cancelar"/>

Anexo-III GESTIONAR PREPARACIÓN DEL PACIENTE.

Editar - TIPOS DE MUESTRA

<input type="button" value="Agregar"/>		<input type="button" value="Eliminar"/>	
Nombre	<input type="text" value="Sangre"/>	*	
(*) Los campos señalados son de entrada obligatoria.			
		<input type="button" value="Aceptar"/>	<input type="button" value="Cancelar"/>

Anexo-IV Gestionar Tipos de Muestras.

Editar - MODOS DE CONSERVACION

<input type="button" value="Agregar"/> <input type="button" value="Eliminar"/>	
Nombre	<input type="text" value="Refrigeración"/> *
(*) Los campos señalados son de entrada obligatoria.	
<input type="button" value="Aceptar"/> <input type="button" value="Cancelar"/>	

Anexo-V GESTIONAR MODOS DE CONSERVACIÓN.

<input type="button" value="Gestionar Exámenes"/>	Exámenes por Tipo de Unidad
<input checked="" type="checkbox"/> Listado General	EXAMENES POR TIPO DE UNIDAD
Tipo de Unidad	<input type="text" value=""/> ▼
Asociar	
<input type="button" value="Imprimir"/> <input type="button" value="Agregar"/> <input type="button" value="Eliminar"/>	
Nombre	
<input type="radio"/> Exámen1	
<input type="radio"/>	
<input type="radio"/>	
1 - 1/1	
<input type="button" value="Imprimir"/> <input type="button" value="Agregar"/> <input type="button" value="Eliminar"/>	

AnexoVI Definir Exámenes por Tipo de Unidad.

Listado General - MODIFICAR EXAMENEN

Sistemas Diagnóstico ▼	Susclasificación ▼
Nombre <input type="text"/>	Abreviatura <input type="text"/>
Tipo de Exámen ▼	Precio de Venta ▼
Tipo de Muestra del Exámen ▼	Tipo de Muestra Colectada ▼
Cantidad de Muestra Colectada ▼	UM Muestra Colectada ▼
<input type="button" value="Aceptar"/> <input type="button" value="Cancelar"/>	

Algoritmos
 Preparación del Paciente
 Métodos Analíticos
 Sistemas de Calibración
 Conservación de la Muestra
 Formas de Expresión del Resultado
 Compuesto por

Anexo-VII GESTIONAR EXÁMENES.

Listado General - EXAMENES

 **Buscar por:**

Sistemas Diagnóstico <input type="button" value="v"/>	Susclasificación <input type="button" value="v"/>
Nombre <input type="text"/>	Tipo de Exámen <input type="button" value="v"/>
Sistemas de Calibración <input type="button" value="v"/>	Método Analítico <input type="button" value="v"/>
Tipo de Muestra del Exámen <input type="button" value="v"/>	Preparación del Paciente <input type="button" value="v"/>
Tipo de Muestra Colectada <input type="button" value="v"/>	

Ordenar por **DESC**

Escoja los criterios de búsqueda y presione "Buscar".

Anexo-VIII BUSCAR EXAMEN.

Listado general - METODOS ANALITICOS

<input type="button" value="Imprimir"/>	<input type="button" value="Agregar"/>	<input type="button" value="Eliminar"/>
Nombre	Descripción	
<input type="radio"/> Uracic	Consiste en ...	
<input type="radio"/>		
1 - 1/1		
<input type="button" value="Imprimir"/>	<input type="button" value="Agregar"/>	<input type="button" value="Eliminar"/>

Anexo-IX GESTIONAR Métodos Analíticos.

Listado general - SISTEMAS DE CALIBRACION

<input type="button" value="Imprimir"/>	<input type="button" value="Agregar"/>	<input type="button" value="Eliminar"/>		
Nombre	Tipo de Calibrador	Unidad de Medida	Fabricante	Marca
<input type="radio"/> Draking	bbbb	ml	Empresa Finlay	Beriger
<input type="radio"/>				
1 - 1/1				
<input type="button" value="Imprimir"/>	<input type="button" value="Agregar"/>	<input type="button" value="Eliminar"/>		

Anexo-X GESTIONAR SISTEMAS DE CALIBRACIÓN.

☑ **Listado General - NUEVO REACTIVO**

Nombre del Reactivo	<input type="text"/>	Identificador Internacional	<input type="text"/>
Fabricante	<input type="text"/>	Marca	<input type="text"/>
Tipo de Reactivo	<input type="text"/>	Método de Preparación	<input type="text"/>
Presentación	<input type="text"/>		
			<input type="button" value="Aceptar"/> <input type="button" value="Cancelar"/>
Componer Reactivos	<input type="button" value="Eliminar"/>		

Anexo-XI BUSCAR REACTIVO.

☑ **Listado General - GESTIONAR REACTIVOS**

 Buscar por:			
Nombre del Reactivo	<input type="text"/>	Identificador Internacional	<input type="text"/>
Fabricante	<input type="text"/>	Marca	<input type="text"/>
Presentación	<input type="text"/>	Método de Preparación	<input type="text"/>
Tipo de Reactivo	<input type="text"/>		
Ordenar por		Nombre del Reactivo <input type="text"/>	<input type="checkbox"/> DESC <input type="button" value="Buscar"/> <input type="button" value="Mostrar todos"/>
Escoja los criterios de búsqueda y presione "Buscar".			

Anexo-XII GESTIONAR REACTIVOS.

Editar - UNIDADES DE MEDIDA

<input type="button" value="Agregar"/> <input type="button" value="Eliminar"/>	
Nombre	<input type="text" value="Mililitros"/> *
Id-Producto	<input type="text" value="Ml"/> *
(*) Los campos señalados son de entrada obligatoria.	
<input type="button" value="Aceptar"/> <input type="button" value="Cancelar"/>	

Anexo-XIII GESTIONAR UNIDADES DE MEDIDA

☒ Listado General
DEFINIR EXAMENES POR LABORATORIO

Laboratorio

Asociar Examen del Laboratorio

Imprimir	Agregar	Eliminar
Nombre	Vigente	
<input type="radio"/> Examen1	Si	
<input type="radio"/>		
		1 - 1/1
Imprimir	Agregar	Eliminar

Anexo-XIV Definir Exámenes por Laboratorio.

☒ Listado General
DEFINIR REACTIVOS POR LABORATORIO

Laboratorio

Asociar Reactivo del Laboratorio

Imprimir	Agregar	Eliminar
Nombre		
<input type="radio"/> XSXS		
<input type="radio"/>		
		1 - 1/1
Imprimir	Agregar	Eliminar

Anexo-XV Definir Reactivos por Laboratorio.

☒ Listado General
DEFINIR LABORATORIOS DE REFERENCIA

Buscar por:

Provincia Municipio
 Localidad Nombre
 Referencia Si No

Ordenar por DESC Buscar Mostrar todos

Imprimir Agregar Eliminar

Anexo-XVI DEFINIR LABORATORIOS DE REFERENCIA.

Buscar por:

Provincia
 Localidad
 Tipo de Unidad
 Subclasificación

Municipio
 Area de Salud
 Sistema Diagnóstico
 Exámen

Referencia Si No

Ordenar por Nombre DESC

Nombre	Teléfono	Correo Electrónico	Dirección	Referencia
<input type="radio"/> Lab1	2425	correo@uci.cu	San Antonio.....	aaa
<input type="radio"/>				

1 - 1/1

Anexo-XVII LISTAR LABORATORIOS.

Editar - << Codificadores >>

Nombre *

(*) Los campos señalados son de entrada obligatoria.

Anexo-XVIII CONFIGURAR.