

UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS
FACULTAD 7



*“Módulo de Reportes Estadísticos del Sistema de Gestión para
la Ingeniería Clínica y Electromedicina”*

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

Autora: Ismaray Morera Aragón

Tutores: Ing. Zoila Esther Morales Tabares
Ing. Dennys Javier Hernández Peña

La Habana, 2 de junio de 2011.
“Año 53 de la Revolución”

“Tengo mi propia versión del optimismo. Si no puedo cruzar una puerta, cruzaré otra o haré otra puerta. Algo maravilloso vendrá, no importa lo oscuro que esté el presente.”

Rabindranath Tagore

DATOS DE CONTACTO

Nombre: Dennys Javier.

Apellidos: Hernández Peña.

Ing. Informático graduado del curso 2004-2005. CUJAE. Profesor Asistente. Líder de proyecto del Sistema de Gestión para la Ingeniería Clínica y Electromedicina. Ha impartido las asignaturas de Seguridad Informática, Sistema Operativo e Ingeniería de Software. Ha sido tribunal de tesis 4 años consecutivos y tutor de más de 20 tesis de grado. Actualmente profesor del Departamento Sistemas de Apoyo a la Salud del Centro de Informática Médica y profesor adjunto a la Facultad # 7.

Correo electrónico: dhernandezp@uci.cu

Nombre: Zoila Esther.

Apellidos: Morales Tabares.

Ingeniera en Ciencias Informáticas, graduada del curso 2007-2008. Universidad de Ciencias Informáticas. Profesor Instructor. Profesora de Contabilidad y Finanzas, Analista del Proyecto Sistema de Gestión para la Ingeniería Clínica y Electromedicina. Ha impartido las asignaturas de Lenguaje y Técnicas de Programación, Bases de Datos y Programación Visual II. Ha sido tutora de Tesis de Enseñanza Politécnica. Actualmente profesora de la Facultad # 7 y adjunta al Departamento Sistemas de Apoyo a la Salud del Centro de Informática Médica.

Correo electrónico: zemorales@uci.cu

DEDICATORIA

A mi madre, a mi padre, a mis hermanos y a mi Chichito, por ser ellos los que siempre están junto a mí apoyándome, por ser ellos los que comparten de forma incondicional mis risas, mis lágrimas y mis miedos.

Ismaray Morera Aragón.

RESUMEN

La correcta gestión de reportes estadísticos en las empresas por parte del personal involucrado, trae consigo un aumento en la calidad de la toma de decisiones. Varias herramientas y sistemas informáticos actualmente, brindan esta posibilidad a sus usuarios finales, tanto a nivel internacional como nacional. El Centro de Ingeniería Clínica y Electromedicina, encargado de la gestión de la información referente a los equipos médicos existentes en Cuba y los recursos asociados a estos; realiza actualmente estas actividades de forma manual o mediante la utilización de medios obsoletos, debido a que no cuenta con un sistema informático que le ofrezca las funcionalidades requeridas.

Por lo señalado anteriormente, se ha propuesto la creación de un Módulo de Reportes Estadísticos que forme parte del Sistema de Gestión para la Ingeniería Clínica y Electromedicina. Este sistema actualmente, se concibe para la utilización en cada Organización Básica de Ingeniería Clínica y Electromedicina. Para el desarrollo de dicho módulo se ha asimilado la plataforma de trabajo definida por el Centro de Informática Médica y el Departamento de Sistemas de Apoyo a la Salud, conformado el siguiente ambiente de desarrollo: Proceso Unificado de Desarrollo como metodología de desarrollo apoyado en el Lenguaje Unificado de Modelado 2.1 y sustentado por el Visual Paradigm 3.6 y DB Designer Fork 1.0 como herramientas CASE. Además, de la utilización de los frameworks Symfony 1.4 y ExtJS 2.2, soportado sobre una plataforma LAMP, integrado por las tecnologías Ubuntu 10.4, Apache 2.2, MySQL 5.1 y PHP 5.1.

PALABRAS CLAVES: estadísticas, reportes, CICHEM, SIGICHEM, Symfony, ExtJS.

TABLA DE CONTENIDO

INTRODUCCIÓN	1
CAPÍTULO I: FUNDAMENTACIÓN TEÓRICA	7
1.1 Términos relacionados con la investigación.....	7
1.2 Sistemas para la generación de reportes estadísticos.....	8
1.3 Metodologías, tecnologías y herramientas.....	11
CAPÍTULO II: CARACTERÍSTICAS DEL SISTEMA.....	17
2.1 Modelo de negocio.....	17
2.2 Breve descripción del problema.....	17
2.3 Especificación de los requisitos del software.....	19
2.4 Listado de casos de uso del sistema (CUS).....	23
2.5 Diagrama de casos de uso del sistema.....	25
CAPÍTULO III: ANÁLISIS Y DISEÑO DEL SISTEMA.....	30
3.1 Modelo de análisis.....	30
3.2 Modelo de diseño.....	33
3.3 Patrón arquitectónico.....	42
3.4 Patrones de diseño.....	44
3.5 Modelo de despliegue.....	45
CAPÍTULO IV: IMPLEMENTACIÓN Y PRUEBAS DEL SISTEMA.....	47
4.1 Modelo de datos.....	47
4.2 Modelo de implementación.....	48
4.3 Tratamiento de excepciones.....	49
4.4 Estándares y estilos de codificación.....	50
4.5 Modelo de pruebas.....	51
CONCLUSIONES GENERALES.....	58
RECOMENDACIONES.....	59
REFERENCIAS BIBLIOGRÁFICAS.....	60
BIBLIOGRAFÍA.....	62
ANEXOS.....	66
GLOSARIO DE TÉRMINOS.....	67

INTRODUCCIÓN

Desde la creación el 1ro de agosto de 1961 del Ministerio de Salud Pública (MINSAP), esta institución se ha convertido en el organismo rector del Sistema Nacional de Salud (SNS). Se adopta como misión: “*dirigir, ejecutar y controlar la aplicación de la política del Estado y del Gobierno en cuanto a la Salud Pública, el desarrollo de las Ciencias Médicas y la Industria Médico Farmacéutica*”.¹ Además, definiéndose entre sus principios rectores:

- × *Carácter estatal y social de la medicina.*
- × *Accesibilidad y gratuidad de los servicios.*
- × *Orientación profiláctica.*
- × *Aplicación adecuada de los adelantos de la ciencia y la técnica.*
- × *Participación de la comunidad e intersectorialidad.*
- × *Colaboración internacional.*
- × *Centralización normativa y descentralización ejecutiva*².

Con su surgimiento, quedan establecidas las bases del futuro desarrollo de la Salud Pública en el país, dándose paso a la creación de nuevas estructuras e instituciones de salud. Estas necesitaban ser equipadas con nuevas tecnologías médicas, lo que provoca que al aumentar la cantidad de equipos médicos existentes en el país fuera inevitable la creación de talleres que repararan y dieran mantenimiento a estos. En la década del 80, el sector de la salud comienza a poblarse de nuevas tecnologías de punta, la creación de un nuevo programa que involucrara la electromedicina en Cuba se hace impostergable, nace así una Dirección Nacional separada del SNS, aunque se subordinaría a este, denominado Centro Nacional de Electromedicina (CNE).

En el año 1983 el CNE se plantea la misión de garantizar la sostenibilidad de la tecnología médica mediante el correcto asesoramiento de las adquisiciones de equipos y piezas, y la prestación de servicios técnicos de excelencia durante el tiempo de explotación de cada tecnología. Paralelamente a estos

¹ INFOMED. Portal de Salud de Cuba. **Aspectos Generales del Sistema Nacional de Salud.** [Consultado: 15/10/2010.] [En línea]: http://www.sld.cu/sistema_de_salud/aspectos.html.

² Ídem. 1.

cambios, a nivel mundial tenían lugar nuevos adelantos en cuanto al surgimiento y desarrollo de las Tecnologías de la Informática y las Comunicaciones (TIC's). Las nacientes innovaciones en las comunicaciones y en las tecnologías de la computación comienzan a acelerar el cambio al producir el derrumbe de las barreras de información existentes. Cuba no queda aislada de este desarrollo y se da la tarea de comenzar el proceso de informatización de manera gradual en sus distintos sectores. Durante el V Congreso del Partido Comunista de Cuba celebrado en 1997, se expresa en la Resolución Económica:

*“El país debe encaminarse resueltamente a la modernización informática mediante un programa integral que involucre a las organizaciones que deben proveer los recursos materiales, financieros e intelectuales y a las entidades económicas, políticas y sociales que deben traducirlos en más y mejores productos y servicios. La industria de los servicios informáticos deberá asegurar la modernidad de su base técnica y organizativa, y la elevación constante del nivel científico-técnico de sus especialistas con vistas a garantizar esos propósitos”.*³

Para dar respuesta a este llamado, el SNS comienza a dar sus primeros pasos con el fin de establecer las bases para su informatización. Su aplicación ha permitido al sector de la salud, contar con diversos avances en esta esfera.

El CNE en la batalla por no quedar atrás en este proceso, el 28 de junio del 2006, crea el Grupo de Automatización y Desarrollo de Electromedicina (GADE), en busca del objetivo de apoyar la gestión informativa, investigativa y administrativa del Sistema Nacional de Electromedicina (SNE). En noviembre de este mismo año se inicia el desarrollo de un sistema automatizado que permitiría realizar las gestiones específicas de las actividades del CNE e integrar las informaciones de carácter general del SNS, denominado: Sistema Integral de Gestión para Electromedicina (SIGEM).

Con la reestructuración de la misión y el alcance del CNE, se decide cambiar el nombre por Centro de Ingeniería Clínica y Electromedicina (CICEM), el mismo vincularía la ingeniería clínica con la electromedicina. Todo esto trajo consigo la necesidad de cambios en el producto existente SIGEM pues el mismo no respondía a las necesidades reales del centro y se crea el Sistema de Gestión para la Ingeniería Clínica y Electromedicina (SIGICEM), que actualmente se encuentra en desarrollo por un

³ Sitio oficial del MIC. **Tecnología de la información**. [Consultado: 15/10/2010.] [En línea:] <http://www.mic.gov.cu/HThemEmp.aspx?1>.

equipo de trabajo perteneciente al Departamento Sistemas de Apoyo a la Salud (SAS) del Centro de Informática Médica (CESIM), núcleo productivo de la Facultad # 7, de la Universidad de las Ciencias Informáticas (UCI).

Previamente al desarrollo de este sistema se han identificado un conjunto de funcionalidades que permiten la recogida de la información gestionada por el CICEM referente a los equipos médicos existentes en el país, como son: ubicación de equipos médicos, nomencladores, mantenimiento, recursos humanos y otros. Esto provoca que resulte engorrosa la manipulación de los distintos datos, debido a que el volumen de información es cada vez mayor.

Actualmente, el SIGICEM, no posee dentro de sus funcionalidades un módulo que proporcione la elaboración de reportes estadísticos; por tanto estos son realizados manualmente por el personal perteneciente a los centros de electromedicina. Esta situación provoca que el trabajo en ocasiones se torne engorroso, por lo que no se gestiona de forma adecuada la información, lo que provoca su pérdida y que se incumplan en algunos casos con las fechas definidas para la entrega de estos reportes estadísticos.

Por lo anteriormente expuesto se tiene el siguiente **problema a resolver**: ¿Cómo contribuir a la gestión de reportes estadísticos en el Sistema de Gestión para la Ingeniería Clínica y Electromedicina?; se tiene como **objeto de estudio**: Proceso de gestión de la información en el Centro de Ingeniería Clínica y Electromedicina, enmarcado en el **campo de acción**: Proceso de gestión de la información en el Sistema de Gestión para la Ingeniería Clínica y Electromedicina.

En busca de establecer una guía de la investigación a desarrollar se ha propuesto como **objetivo general**: Desarrollar el módulo de Reportes Estadísticos del Sistema de Gestión para la Ingeniería Clínica y Electromedicina. Para dar cumplimiento al objetivo planteado para esta investigación se han definido las siguientes **tareas de investigación**:

- ✘ Analizar los sistemas informáticos existentes a nivel internacional y nacional para la generación de reportes estadísticos, estableciéndose similitudes con la investigación en curso.
- ✘ Determinar los procesos de negocio asociados a la gestión de la información de los reportes estadísticos para la obtención de una guía para la implementación del sistema.

- ✘ Aplicar las herramientas y tecnologías definidas por el Centro de Informática Médica (CESIM) y el Departamento Sistemas de Apoyo a la Salud (SAS), como ambiente de desarrollo.
- ✘ Generar los artefactos correspondientes a las Disciplinas de Trabajo: Modelado de Negocio, Gestión de Requerimientos, Análisis y Diseño, Implementación y Prueba, guiados por el Proceso Unificado de Desarrollo (RUP).
- ✘ Implementar el Módulo de Reportes Estadísticos del Sistema de Gestión para la Ingeniería Clínica y Electromedicina (SIGICEM).

Entender los detalles del problema existente y qué se desea resolver, puede resultar en ocasiones engorroso, debido a esto es necesario al comienzo de la investigación definir los principales **métodos científicos** que posteriormente se utilizarán. Estos métodos permiten estudiar las características que no son observables y las fenomenológicas, respectivamente.

Según plantea Rolando Alfredo, estos métodos: *“representan un nivel de la investigación cuyo contenido procede de la experiencia y es sometido a cierta elaboración racional”*.⁴ Pueden ser encontrados distintos métodos, los cuales pueden ser aplicados a las investigaciones para obtener de manera más rápida y confiable los resultados esperados. Para la presente investigación se han definido los que se listan a continuación:

- ☒ **Histórico – Lógico:** permite la realización del estudio de los fenómenos y acontecimientos en un orden cronológico, para conocer la evolución y desarrollo de la esencia de la investigación en curso.
- ☒ **Analítico-Sintético:** propiciará resumir, enunciar y describir los requerimientos funcionales de la solución propuesta, debido a que este método permite buscar la esencia del problema planteado en la investigación, los rasgos que lo caracterizan y lo distinguen. Además, ofrece la posibilidad de extraer los elementos más importantes que se relacionan con el objeto de estudio.
- ☒ **Modelación:** método teórico que ofrece parte de la información necesaria acerca del objeto que se estudia, se trata de explicar la realidad con la creación de diagramas; los cuales pueden ser presentados en sustitución de la realidad. Mediante su utilización se elaborarán diferentes tipos de

⁴ Hernández, Rolando Alfredo y Coello, Zayda. **El paradigma cuantitativo de la investigación científica**. Editorial Universitaria, 2002.

diagramas (incluidos en la concepción de UML 2.1, especialmente) que brindarán información clara sobre el tema de estudio, mediante el descubrimiento de nuevas relaciones y cualidades del objeto de estudio.

- ☒ **Grupo de Discusión:** método empírico que se basa en efectuar una reunión de varias personas que discuten sobre un tema de interés común, con la ayuda de un coordinador y un secretario. A través de este se busca adquirir más información sobre un tema, tomar decisiones conjuntas o ambas acciones.

Como **aporte y novedad científica** se espera:

La obtención de un módulo de reportes estadísticos que permita manejar la información referente al estado técnico de los equipos instalados en el país. Asimismo, permitirá agilizar la toma de decisiones ante eventos ocurridos con el equipamiento médico. También, ofrecerá la posibilidad de monitorear la información recepcionada en el Sistema de Ingeniería Clínica y Electromedicina. Fortalecerá las bases para el aseguramiento de la continuidad del proceso de gestión tecnológica del equipamiento médico y la consolidación de la actividad de atención técnica postmercado.

El documento de la investigación científica se encuentra estructurado de la siguiente forma:

Capítulo 1: Fundamentación teórica.

Se presenta una visión de los conceptos básicos, así como del estado del arte relacionado con la generación de reportes y estadísticas, a nivel internacional y nacional. También se podrán consultar los métodos científicos de la investigación a utilizar. Se exponen las metodologías, tecnologías y herramientas adoptadas para el desarrollo del trabajo.

Capítulo 2: Características del sistema.

Se realiza una breve descripción del problema a resolver y un estudio de las principales actividades que se efectúan en el CICEM relacionados con los procesos de generación de estadísticas y reportes. Se definen la relación de requisitos funcionales, no funcionales y los casos de uso del sistema descritos a través de los artefactos definidos.

Capítulo 3: Análisis y Diseño del sistema.

Se ofrece una representación de la realización de casos de uso del sistema a través de diagramas de clases del análisis, del diseño y los de interacción, específicamente los de comunicación, así como los patrones de diseños y arquitectónicos utilizados. Además del Modelo de Despliegue que será utilizado para la puesta en marcha del sistema.

Capítulo 4: Implementación y Prueba del sistema.

Se expone una descripción del desarrollo de la solución inicialmente propuesta mediante la representación del Modelo de Implementación. Además de asimilarse los estándares y estilos de codificación a utilizar. Se presentan algunos de los criterios tomados en cuenta para llevar a cabo el tratamiento de errores en el sistema; así como. Se elabora el Modelo de Pruebas necesario para la posterior realización de estas al Módulo de Reportes Estadísticos.

CAPÍTULO I: FUNDAMENTACIÓN TEÓRICA

El presente capítulo tiene como premisa fundamental realizar una breve descripción de los principales conceptos relacionados con el tema de la investigación propuesta; además de dar a conocer los principales resultados del estudio previo realizado, sobre varios aspectos relacionados con la generación de reportes estadísticos. Se ofrece también una caracterización de las principales metodologías, tecnologías y herramientas a utilizar.

1.1 Términos relacionados con la investigación

En la actualidad la transformación de esa enorme red de comunicaciones que en sus inicios fue concebida para uso gubernamental, Internet, ha evolucionado en una gran cantidad de redes privadas interconectadas entre sí. Su desarrollo creciente y el auge que toman sus servicios logran la integración cada vez mayor de nuevas redes y usuarios, por lo que su dominio crece, al tiempo que surgen nuevas vías para aprovechar sus ventajas.

El enorme uso de esta red de redes y de las TIC's ha traído consigo la necesidad de informatizar gran cantidad de los procesos que se realizan cotidianamente en centros escolares, hogares, empresas y otras instituciones. Mediante estas redes se gestionan grandes volúmenes de información, es decir, “**conjunto organizado de datos**, que constituye un **mensaje** sobre un cierto fenómeno o ente. La información permite resolver problemas y tomar decisiones, ya que su uso racional es la base del conocimiento”.⁵ El proceso de gestión de información lleva consigo todas las actividades que sean necesarias para ordenar, disponer u organizar de manera tal que se obtenga la información adecuada.

De todo ese cúmulo de información obtenido es necesario llevar a cabo un control estadístico, la cual utiliza los datos numéricos que han sido recogidos para realizar un estudio de la realidad. La estadística se presenta de forma transversal a una amplia variedad de disciplinas y facilita la toma de decisiones por parte de los directivos en las empresas. En busca de evaluar los resultados, se hace esencial obtener una entrega periódica de información en formatos previamente definidos. Estos deben resultar amenos y entendibles a los usuarios. Además que relacionen los aspectos más significativos para lograr la toma de decisiones, es decir, es necesario la generación de reportes.

⁵ Definición de. **Concepto de información**. [Consultado: 2/11/2010.] [En línea:] <http://definicion.de/infomacion>

Desde el punto de vista de la informática, “los reportes son informes que organizan y exhiben la información contenida en una base de datos. Su función es aplicar un formato determinado a los datos para mostrarlos por medio de un diseño atractivo y que sea fácil de interpretar por los usuarios. El reporte, de esta forma, confiere una mayor utilidad a los datos”.⁶

En la actualidad existen herramientas denominadas generadores de reportes que permiten gestionar los informes solicitados. Una de las acciones más relevantes efectuadas en el interior del proceso de gestión estadísticas es la de generar reportes con la cual se demuestra la eficiencia y la calidad de las aplicaciones realizadas.

1.2 Sistemas para la generación de reportes estadísticos

La concepción de los reportes estadísticos era un proceso llevado a cabo por analistas y especialistas de forma manual, con el surgimiento de máquinas de dibujos y el diseño de croquis por los diseñadores aumentó un poco la calidad de los reportes aunque se elaboraban manualmente. Con el desarrollo tecnológico alcanzado en la historia de la humanidad, han surgido sistemas que hacen más fácil el manejo de las estadísticas y sus respectivos reportes. A continuación se presentan algunas herramientas que se han concebido para la generación de reportes estadísticos.

Sistemas internacionales existentes.

☒ Crystal Reports

“Esta potentísima herramienta que posee diferentes versiones es la herramienta de elaboración de informes estándar para Visual Studio.NET, la cuál permite crear contenido interactivo con una excelente calidad de presentación, con Crystal Reports puede almacenarse informes en aplicaciones de escritorio y web, además se puede publicar informes de Crystal como servicios web de informes en un servidor web”.⁷

Dentro de sus características fundamentales se puede señalar que: soporta gráficos, imágenes y reportes y soporta sentencias de Visual Basic. Crystal Report cuenta con años de experiencia y desarrollo, pero

⁶ Definición de. **Definición de reporte**. [Consultado: 2/11/2010.] [En línea:] <http://definicion.de/reporte>

⁷ Moreno Tamayo, Yoanki. **Análisis y diseño del Módulo Generador de Reportes del proyecto ONE**. Universidad de las Ciencias Informáticas, 2008. [Consultado: 08/03/2011.]

sus desventajas radican en que es propietaria y las licencias son costosas. Además de que para modificar el reporte es preciso recompilar el proyecto donde se encuentra.

☒ **Jasper Reports**

Herramienta libre, multiplataforma, que soporta imágenes y reportes, soporta varios orígenes de datos y exporta salidas en diferentes formatos. Se integra con el JfreeChart (librería para la generación de gráficas escrita en Java). Pero la misma es dependiente de la plataforma Java, complejo de usar y presenta deficiente robustez y eficacia para reportes complejos; características que hacen ver desventajas en este generador de reportes.

☒ **Report Manager**

Sistema multiplataforma (Windows y Linux) y libre, el cual posibilita también el soporte de gráficos, imágenes y reportes, también varios orígenes de datos y líneas de comandos. Esta herramienta presenta como desventajas las siguientes características: viene especialmente diseñado para usar con Delphi, C++ Builder y Kylix; su interfaz gráfica es poco intuitiva y poco amigable y se hace difícil dominarlo y comprender su funcionamiento.

Sistemas nacionales existentes

☒ **Sistema Obstetricia y Ginecología Automatizado**

“El Instituto Superior de Ciencias Médicas, Carlos J. Finlay de la Facultad Tecnología de la Salud de la provincia de Camagüey, concibió realizar un software donde se utilizaron los Registros de Obstetricia y Ginecología (Modelo 66-15) y Registro de Legrado (Modelo 56-18) del Sistema de Información Estadística. El diseño y programación de la base de datos se efectuó en la Facultad de Tecnología de la Salud “Dr. Octavio de la Concepción y de la Pedraja” y el trabajo de planificación se realizó en las dependencias del Sectorial Provincial de Salud de Camagüey, donde se determinó la no existencia en el país de la automatización de este sistema estadístico. El sistema cuenta con varios reportes, los cuales se utilizan para llevar el control estadístico de dicho sistema como son:

- ✗ *Interrupciones por municipios de pacientes con menos de 20 años.*
- ✗ *Interrupciones por municipios de todas las edades.*

- × *Total de nacidos vivos de pacientes menos de 20 años.*
- × *Bajo peso por municipios.*
- × *Totales de tipos de parto (Fisiológicos, cesáreas y formas mecánicas).*
- × *Adolescentes por municipio entre edades de 10 a 14.*
- × *Adolescentes por municipio entre edades de 15 a 19.*
- × *Registros de adolescentes entre edades de 10 a 14, de 15 a 19 de los cuales se informa bajo peso, nacido vivo o muerto, cesáreo y parto.”⁸*

☒ **INFOPAMI**

“Para el procesamiento y análisis de la información estadística del Programa de Atención Materno Infantil, en el nivel de Atención Primaria de la Salud, es diseñado un sistema informático para facilitar la toma de decisiones dentro del programa. Generalmente estos reportes se emiten mensualmente, pero pudiera hacerse indistintamente en cualquier otra ocasión según necesidad del área de salud o de instancias superiores. El sistema se diseñó para que la actualización periódica y sistemática de sus datos, así como la obtención de sus reportes sea realizada por los especialistas médicos. Los reportes que se emiten por el sistema son:

- × *Lactantes con patologías crónicas.*
- × *Lactantes por edad.*
- × *Tipo de lactancia.*
- × *Total de gestantes menores de 20 y mayores de 35 años.*
- × *Total de lactantes con patologías crónicas”.*⁹

☒ **AKADEMOS**

El Sistema Automatizado para la Gestión Académica (Akademos) es una aplicación informática que gestiona las informaciones referentes al proceso docente-educativo dentro de la UCI. Dentro de sus funcionalidades se pueden encontrar algunas de las que dan respuesta a la necesidad de generación de

⁸ Sánchez Hemández, Yanet, Matos Arias, José Manuel. **Reportes Estadísticos del Registro de Partos y Nacimientos.** Universidad de las Ciencias Informáticas, 2009. [Consultado: 08/03/2011.]

⁹ Ídem 7.

reportes. Para el desarrollo del mismo se hizo uso del lenguaje de programación C# con la utilización de Visual Studio.NET, Microsoft SQL Server como servidor de base de datos.

☒ PATDSI

Paquete de herramientas con el fin de propiciar ayuda a la toma de decisiones de las empresas, ideado como una plataforma web para la inteligencia de negocio. Desarrollado por el Departamento de Soluciones Integrales perteneciente al Centro de Tecnologías de Gestión de Datos de la Universidad de las Ciencias Informáticas. Este paquete incluye un Generador de Reportes Dinámicos; el mismo elabora los reportes en tiempo de diseño por estar orientado al programador y no al usuario final; por lo cual no permite la modificación dinámica del diseño del reporte. Además de que no todos los usuarios son capaces de diseñar reportes de manera rápida y con la calidad requerida, debido al nivel de complejidad que la aplicación puede presentar para la creación de un reporte.

1.3 Metodologías, tecnologías y herramientas

Proceso Unificado de Software (RUP)

El Proceso Unificado de Software, (RUP), del inglés Rational Unified Process, es un proceso de desarrollo de software, es decir, “conjunto de actividades necesarias para transformar los requisitos del usuario en un sistema software.”¹⁰ Este permite su adaptación a una gran variedad de sistemas según las especificidades que presente, por lo que señala quién debe hacer qué, cuándo y cómo.

Es necesario destacar que RUP es dirigido por casos de uso, centrado en la arquitectura, iterativo e incremental. Propone una secuencia ordenada y lógica para la puesta en marcha de las diferentes actividades que en su concepción se definen, denominada Flujo de Trabajo o Disciplina, estos se encuentran agrupados en Flujos de Trabajo del Proceso y Flujos de Trabajo de Soporte, estas a su vez transitan por las cuatro fases que forman parte de este proceso.

Cada una de estas fases cuenta con un objetivo específico a cumplir durante su trayectoria, en la fase de Inicio se realizan las actividades necesarias para definir el alcance del proyecto para así en la etapa de Elaboración realizar una planificación adecuada y establecer una línea base.

¹⁰ Jacobson, Ivar, Booch, Grady y Rumbaugh, James. **El Proceso Unificado de Desarrollo de Software**. España: Adison-Wesley.

Lenguaje Unificado de Modelado (UML 2.1)

UML, del inglés Unified Modeling Language, es el resultado de la unión de las metodologías de análisis y diseño creadas respectivamente por Grady Booch, James Rumbaugh e Ivar Jacobson. Mediante la modelación se logran hacer más entendibles distintos puntos de la realidad que sean de interés, todo esto a partir de la complementación de varias técnicas de modelado en una sola. Integrado por un conjunto de diagramas, los cuales se encuentran agrupados en dos categorías fundamentales: diagramas de estructura y diagramas de comportamiento.

Este lenguaje de modelado permite crear un nivel de comprensión y entendimiento entre los analistas, desarrolladores o cualquier personal involucrado y hace más fácil la comunicación existente entre ellos; permite visualizar, especificar, construir y documentar la información con la que se cuenta.

Herramientas CASE

Herramientas de Ingeniería Asistida por Computadora, del inglés Computer Aided Software Engineering (CASE), son aquellas que brindan un entorno de trabajo a todo el equipo de desarrollo de software con el uso de las tecnologías de la información. Permite elevar la productividad durante todo el ciclo de vida del producto y la estructuración del mismo en lenguaje UML.

Visual Paradigm for UML 3.6

Herramienta CASE desarrollada por la compañía Visual Paradigm International, su concepción incluye: importar diagramas de Microsoft Office Visio a Visual Paradigm, dar soporte a patrones de diseño y a las tres formas del Diagrama Entidad Relación, conceptual, lógica y física. Posibilita la mejora de la trazabilidad de elementos mediante el historial de revisiones. Muestra como elemento estereotipado del modelo un icono de imagen del diagrama según se desee.

Esta herramienta ofrece soporte al ciclo de vida completo del software; permite la captura de requisitos, la creación de diagramas UML, la realización de ingeniería inversa, la generación de código PHP y otros lenguajes de programación.

☒ DB Designer Fork 1.0

Sistema utilizado para lograr el desarrollo de aplicaciones que involucren el trabajo con bases de datos que integren diseño, modelado, creación y mantenimiento en un ambiente simple y optimizado. Posibilita la generación del esquema de la base de datos definida por el usuario, permite la sincronización del modelo con la base de datos, soporta índices y todos los tipos de campos de MySQL, al igual que todos los tipos de campos que defina el usuario.

Es posible la construcción de sentencias SQL desde una consola que posee con este propósito, la cual cuenta con un histórico de los comandos SQL utilizados; además, almacena los mismos conjuntamente con el modelo. Este es capaz de generar código para SQL Server, MySQL, Oracle y PostgreSQL (PGSQL). Emplea Licencia Pública General (GPL), lo que ofrece al usuario la posibilidad de compartir el software licenciado bajo ella, así como realizar cambios en él. Puede ser ejecutado en distintos sistemas operativos debido a su condición de ser multiplataforma.

Entorno Integrado de Desarrollo

Un Entorno Integrado de Desarrollo o Integrated Development Environment (IDE) es una herramienta utilizada por los programadores para la implementación del producto deseado. Con su utilización es posible crear la interfaz gráfica de usuario y escribir el código de lo que hará la aplicación de una manera más amena y en una interfaz amigable. Además hace que el código sea más entendible y organizado, pues implementa en su interior ciertos estándares y niveles de indentación, hace más fácil la integración de un equipo de trabajo.

☒ NetBeans 6.9.

La empresa Sun Microsystems funda el proyecto de código abierto NetBeans en junio 2000, a partir de esta fecha se comenzaron a suceder las distintas versiones de este IDE, que logra alcanzar popularidad y una gran comunidad. Posibilita el desarrollo de aplicaciones con mayor sencillez y logra un aumento en la productividad.

“La integración con Symfony permite desarrollar aplicaciones de forma más sencilla y productiva. En primer lugar, es posible crear nuevos proyectos y aplicaciones directamente desde el IDE. También se

*pueden ejecutar todas las tareas de Symfony, incluso pasándole argumentos y opciones, visualizando el resultado sin necesidad de utilizar una consola de comandos externa”.*¹¹

Frameworks

*“Un framework simplifica el desarrollo de una aplicación mediante la automatización de algunos de los patrones utilizados para resolver las tareas comunes. Además, un framework proporciona estructura al código fuente, forzando al desarrollador a crear código más legible y más fácil de mantener. Por último, un framework facilita la programación de aplicaciones, ya que encapsula operaciones complejas en instrucciones sencillas”.*¹²

Symfony 1.4

*“Framework diseñado para optimizar, gracias a sus características, el desarrollo de las aplicaciones web. Para empezar, separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web. Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja. Además, automatiza las tareas más comunes, permitiendo al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación”.*¹³

Su primera versión fue publicada en Octubre de 2005. Para mantener su enfoque orientado a objetos, hace uso de Propel, un mapeo de objetos a bases de datos, ORM, de sus siglas en inglés object-relational mapping. Este hace que el programador se evite realizar grandes consultas SQL, simplemente a través de llamadas a los objetos, este las traduce a optimizadas consultas SQL sin tener en cuenta el gestor de base de datos utilizados.

ExtJS 2.2

Framework de presentación JavaScript del lado del cliente para el desarrollo de aplicaciones web. Ofrece múltiples opciones para el trabajo con las validaciones y manejo de errores en el cliente. La personalización de temas de estilos es posible en su utilización, además que provee el trabajo con una amplia configuración e intenso trabajo con las Hojas de Estilo en Cascada (CSS, Cascading Style Sheets).

¹¹ Sitio oficial. **NetBeans ya incluye soporte para Symfony.** [Consultado: 17/11/2010.] [En línea] <http://www.symfony.es/2009/10/05/netbeans-ya-incluye-soporte-para-symfony/>

¹² Potencier, Fabien y Zaninotto, François. **Symfony 1.2, la guía definitiva.** 2008.

¹³ Ídem 12.

Toda su funcionalidad en JavaScript se logra mediante librerías YUI (Yahoo User Interface), jQuery, o con la utilización de la librería nativa, así en tiempo de ejecución carga y crea todos los objetos de Lenguaje de Marcado de Hipertexto (HTML, HyperText Markup Language) a través del uso intenso del Modelo de Objetos de Documento (DOM, Document Object Model). Cuenta con dos licencias, una comercial y otra de código abierto.

Este presenta código reutilizable, además de ser independiente o adaptable a diferentes frameworks (Prototype, JQuery, YUI) y orientada a la programación de interfaces tipo escritorio en la web. Permite la implementación de interfaces visuales muy potentes, ofrece componentes para la implementación de tablas, árboles, formularios, contenedores y otros. Es una herramienta multiplataforma, integrada a la tecnología JavaScript Asíncrono y XML (AJAX, Asynchronous JavaScript And XML) que evita las recargas de páginas completas al solicitarse nuevos contenidos. Actualmente cuenta con una extensa comunidad de usuarios.

Plataforma LAMP

*“En el mundo del código libre, la plataforma más extendida para el desarrollo es la conocida como LAMP (por Linux, Apache, MySQL y PHP)”.*¹⁴ Complementación que resulta utilizada en gran escala para las soluciones basadas en la web. Para su utilización se ha conformado mediante la unión de Ubuntu 10.4, Apache 2.2, MySQL 5.1 y PHP 5.1.

Linux es el sistema operativo del servidor, *“Apache es el servidor web, que se ejecuta sobre el sistema operativo, y sobre el cual correrán las aplicaciones de la plataforma. MySQL es la aplicación de bases de datos que permite almacenar, acceder y gestionar informaciones. PHP es el lenguaje de programación de mayor uso para el desarrollo de aplicaciones web. Diseñado específicamente para aplicaciones web, facilita el despliegue de informaciones en línea y consultas en bases de datos”.*¹⁵

¹⁴ Miranda Levis, Carlos. **LAMP: La Plataforma Más Popular para Software Códigos Libres.** [Consultado: 06/03/2011.] [En línea:] <http://www.socinfo.com/codigolibre/lamp>

¹⁵ Ídem 9

Otros componentes utilizados

- Componentes de SIGICEM.

Con el objetivo principal de agilizar el proceso de desarrollo en el SIGICEM y en aras de disminuir el tiempo utilizado para el desarrollo de los implementadores de dicho sistema, han sido concebidas varias clases auxiliares que agilizan las tareas más comunes realizadas.

Se cuenta con funcionalidades que se encargan del trabajo con las interfaces visuales, que hacen que la creación de componentes incluidos en el framework de presentación ExtJS sean gestionados de una forma más fácil desde el framework de desarrollo Symfony. También posee clases externas que han sido ajustadas a las pautas de uso del sistema, por ejemplo, aquellas que permiten la importación y exportación de documentos en formato PDF y XLS.

En el desarrollo del capítulo se ha tratado de hacer notar la necesidad existente de desarrollar una herramienta informática que lleve a cabo el proceso de gestión de reportes estadísticos en el CICEM, producto a que las herramientas existentes hasta el momento no responden a las necesidades reales que se tienen. Varias de las herramientas estudiadas son propietarias y cuentan con costos elevados para la adquisición de sus licencias. Por otra parte, otras son dependientes a plataformas de trabajo predeterminadas o presentan características propias para una institución determinada.

Para el ambiente de desarrollo se aplicaron a las pautas establecidas por el CESIM y el Departamento de SAS: metodología de desarrollo RUP, apoyado en el lenguaje de modelado UML 2.1. Las herramientas: NetBeans 6.9 como IDE, así como Visual Paradigm 3.6 y DB Designer 4 Fork 1.0 como herramientas CASE. Además, de la utilización de los frameworks Symfony 1.4 y ExtJS 2.2, soportado sobre una plataforma LAMP.

CAPÍTULO II: CARACTERÍSTICAS DEL SISTEMA

Exponer las características fundamentales del módulo propuesto es el principal objetivo del presente capítulo, tomándose como punto de partida una breve descripción del problema existente. Por lo que se abordan elementos correspondientes al modelado del negocio, presentándose además los casos de usos del sistema; así como actores y trabajadores del mismo.

2.1 Modelo de negocio

Conocer y familiarizarse con la misión y el funcionamiento de las actividades que se realizan, determinar y dominar sus procesos; además de entender la estructura y la interacción de los procesos de la organización para la cual se desarrollará la solución, resulta esencial durante la etapa inicial del desarrollo del software. Durante este tiempo se debe lograr comprender el problema actual existente e identificar las posibles mejoras, en busca de asegurar que los clientes, usuarios finales y el equipo de trabajo en general presenten un entendimiento común de lo que se desea lograr.

2.2 Breve descripción del problema

El CICHEM cuenta con un sistema informático para gestionar la información referente a la tecnología médica existente en el país y los recursos asociados a estas; el SIGICHEM, carece de funcionalidades que faciliten la adquisición de reportes estadísticos. Antes de dar inicio a las actividades necesarias para la obtención de los reportes estadísticos es preciso tomar en consideración que los Directivos serán los responsables de solicitar los reportes deseados (Ver figura # 1).

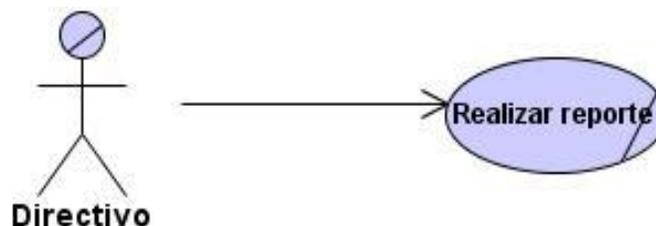


Figura # 1: Diagrama de casos de uso del negocio.

Los especialistas y los directivos tendrán acceso únicamente a la información dentro del área a la que pertenecen. Además, se debe verificar que el reporte presente el formato correcto para no incurrir en inconformidades o inconvenientes posteriores.

La inicialización de las tareas efectuadas con estos fines se encuentra a cargo del Directivo, actor del negocio existente, este solicita dichas reportes estadísticos al especialista; este último, trabajador del negocio, consulta la documentación necesaria para la estructuración del informe que posteriormente será entregado al directivo, concluyendo de esta forma el flujo de actividades. En caso de no existir información se le informa al directivo y finaliza la secuencia de eventos. (Ver figura # 2)

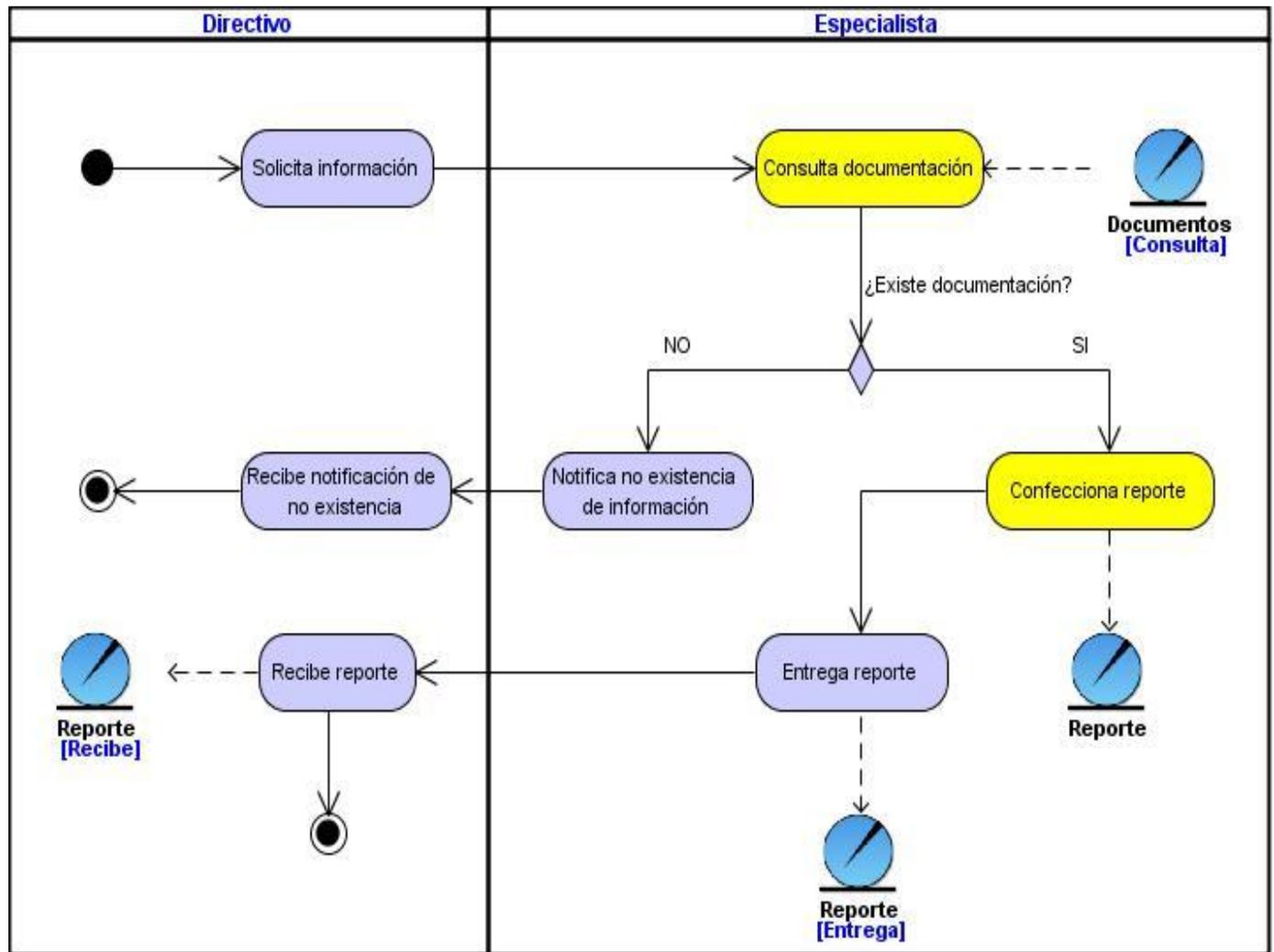


Figura # 2: Diagrama de actividades del caso de uso del negocio Realizar reporte.

Para esclarecer en mayor medida la comprensión por parte de los lectores, se debe señalar que el caso de uso del negocio definido como Realizar reporte, hace referencia a la documentación consultada (Ver figura # 3).



Figura # 3: Modelo de objetos.

Este término se utiliza de manera general, pues según la información solicitada serán utilizados distintos documentos, estos pueden ser: expedientes de equipos médicos, libros de control de entrada a taller, libros de control de salida a taller, planes de mantenimiento y/u otros. Señálese, que el actor del negocio Directivo se encuentra igualmente representado de forma genérica, este se refiere a cualquier personal que solicite información en el área que le compete. Estos pueden ser: especialistas nacionales, especialistas de la Organización Básica de Ingeniería Clínica y Electromedicina (OBICEM), subdirectores técnicos, jefes de brigada u otros.

Se desea por parte del personal involucrado con el SIGICEM, que le sean incorporadas las funcionalidades requeridas mediante las cuales se logren el análisis y procesamiento de reportes estadísticos que contribuyan a la toma de decisiones.

2.3 Especificación de los requisitos del software

La gestión de requisitos contribuye a lograr el éxito durante la construcción del software, por lo que propicia un entendimiento común entre el cliente y el grupo de desarrolladores de aquellos requisitos que deben ser tomados en cuenta en la concepción del producto final. Esta etapa es una de las más difíciles y de ser errada afecta en gran medida la construcción del sistema y más difíciles son de corregir los problemas resultantes.

A través de estos se define qué tiene que hacer como mínimo el sistema que se construya, convirtiéndose estos en el contrato que se debe cumplir, de modo que los usuarios finales tienen que comprender y aceptar los requisitos que se propongan. Los requisitos se encuentran divididos en dos grupos: los funcionales que representan la funcionalidad del sistema y los no funcionales que son aquellos atributos con los que debe contar, pero que no son una función concreta.

Requisitos funcionales

- RF 1 Generar reporte del nomenclador general de ubicación.
- RF 2 Buscar datos del nomenclador general de ubicación.
- RF 3 Generar reporte del nomenclador general de equipos médicos.
- RF 4 Buscar datos del nomenclador general de equipos médicos.
- RF 5 Generar reporte del nomenclador de unidad de salud.
- RF 6 Buscar datos del nomenclador de unidades de salud.
- RF 7 Generar reporte de unidades de salud existentes.
- RF 8 Buscar datos de las unidades de salud existentes.
- RF 9 Generar reporte de especialistas.
- RF 10 Buscar datos de los especialistas existentes.
- RF 11 Generar reporte de piezas.
- RF 12 Buscar datos de las piezas existentes.
- RF 13 Generar reporte del inventario de equipos médicos por países.
- RF 14 Generar reporte del inventario de equipos médicos por regiones.
- RF 15 Generar reporte del inventario de equipos médicos por provincias.
- RF 16 Generar reporte del inventario de equipos médicos por especialidad.
- RF 17 Generar reporte del inventario de equipos médicos por estado.
- RF 18 Generar reporte del inventario de equipos médicos afectados por países.
- RF 19 Generar reporte del inventario de equipos médicos afectados por regiones.
- RF 20 Generar reporte del inventario de equipos médicos afectados por provincias.
- RF 21 Generar reporte del inventario de equipos médicos afectados por especialidad.
- RF 22 Generar resumen de equipos médicos en inventario general.
- RF 23 Generar resumen de equipos médicos afectados en inventario general.
- RF 24 Generar resumen de equipos médicos por países.
- RF 25 Generar resumen de equipos médicos por regiones.
- RF 26 Generar resumen de equipos médicos por provincias.
- RF 27 Generar resumen de equipos médicos por estado.
- RF 28 Generar resumen de equipos médicos por especialidades.
- RF 29 Generar reporte personalizado de equipos médicos.

- RF 30 Buscar datos de equipos médicos.
- RF 31 Generar reporte personalizado de piezas.
- RF 32 Generar reporte personalizado de especialistas.
- RF 33 Generar reporte personalizado de mantenimiento.
- RF 34 Buscar datos de los mantenimientos.
- RF 35 Generar reporte personalizado de órdenes de servicio.
- RF 36 Buscar datos de las órdenes de servicio.
- RF 37 Agrupar resultados por un campo determinado.
- RF 38 Ordenar resultados.
- RF 39 Exportar resultados en PDF.
- RF 40 Listar equipos médicos.

Requisitos no funcionales (RNF)

RNF 1 Usabilidad: el sistema debe ser capaz de ofrecer facilidades de uso por parte de los usuarios finales, lo que favorece el buen entendimiento y aceptación del producto. Debe ser sencillo a la vista de los usuarios.

RNF 2 Fiabilidad: es necesario garantizar la confiabilidad y precisión de la información que se suministra al usuario en cuanto a su veracidad e integridad desde su recopilación y durante toda su manipulación, para evitar cualquier tipo de error. Esto garantiza que los usuarios no avanzados perciban todas sus salidas y pueden trabajar con él cómodamente. La interfaz se realizará fundamentalmente con campos de selección para garantizar la seguridad del sistema y de la base de datos al evitar la entrada de información incorrecta.

RNF 3 Disponibilidad: el sistema deberá estar disponible a tiempo completo, lo que permite el trabajo de los usuarios y la realización de reportes estadísticos en el momento que sea necesario.

RNF 4 Eficiencia: el sistema debe ser de rápida respuesta ante la petición de un usuario, tanto en el momento de realizar las actualizaciones como en el momento de consultar la información. Con fin de lograr un buen rendimiento, debe tener un rápido procesamiento de los datos pues los usuarios estarán en constante intercambio de información. Al estar concebida para un ambiente cliente/servidor, debe garantizar la rapidez de respuesta del sistema ante las solicitudes de los usuarios, al igual que la velocidad de procesamiento de la información, para lo cual se realizará la validación de los datos y la manipulación de

eventos en el cliente y en el servidor aquellas que por cuestiones de seguridad o de acceso a los datos lo requieran. Esto permitirá lograr un tiempo de respuesta más corto, una mayor velocidad de procesamiento, y un mayor aprovechamiento de los recursos.

RNF 5 Soporte: con la puesta en marcha la aplicación, se recopilara la información referente a los defectos y no conformidades existentes para incorporar las mejoras sugeridas al sistema.

RNF 6 Interfaz de usuario: el sistema será diseñado con una interfaz amigable que resulte fácil de usar, mediante el uso de un interfaz semejante a la del sistema operativo Windows, para garantizar que al personal que interactúe con él le resulte familiar, de manera que agilice y facilite el trabajo con el software. Se tendrá en cuenta para la selección de los colores y demás criterios necesarios, las pautas definidas en el Manual de Entidad Corporativa del CICEM, ajustándose parcialmente a las definidas por el Departamento de SAS.

RNF 7 Documentación de usuarios en línea y Ayuda del sistema: el sistema contendrá una ayuda, en la que se explicarán de forma clara los detalles del uso de sus distintas funcionalidades, lo cual permite el buen desempeño de los usuarios a la hora de interactuar con el mismo.

RNF 8 Requisitos de Licencia: el sistema será desarrollado en base a las políticas del software libre, que fueron ajustadas al Sistema Nacional de Salud.

RNF 9 Portabilidad: el sistema será multiplataforma (Linux-Windows), por lo que es posible su ejecución sobre diferentes sistemas operativos sin importar sus versiones, y sin necesidad de modificar su código fuente.

RNF 10 Seguridad: el sistema contará con una política de seguridad diseñada en base a la restricción de usuarios en dependencia del nivel jerárquico que cumpla su rol dentro del mismo.

RNF 11 Políticos-culturales: se deberá hacer un uso correcto del idioma español en la interfaz de la aplicación, con logotipos e imágenes que se encuentren en correspondencia con el carácter de la misma.

RNF 12 Software

Servidor:	<ul style="list-style-type: none"> • Symfony 1.4 • ExtJS 2.2 • Plataforma LAMP.
Cliente:	<ul style="list-style-type: none"> • Mozilla Firefox 3.6 o superior.

Tabla # 1: Requisitos no funcionales de software.

RNF 13 Hardware

Requerimientos mínimos para el servidor:	Requerimientos mínimos para la conexión del cliente:
<ul style="list-style-type: none"> • Microprocesador superior a 2.0 GHz • 512 MB RAM o superior. • 40 GB de Disco Duro como mínimo. 	<ul style="list-style-type: none"> • Microprocesador superior a 1.0 GHz • 128 MB RAM o superior. • MODEM o red con TCP-IP para conexión al servidor.

Tabla # 2: Requisitos no funcionales de hardware.

Definición de los actores del sistema

Se define como actor de un sistema determinado a toda aquella persona, hardware externo u otro sistema que haga uso o interactúe con el sistema en cuestión, los distintos tipos de usuarios que se tengan concebidos serán actores del sistema. Todo aquel actor del negocio que posee posibles actividades a automatizar es un candidato a actor del sistema, así como todo trabajador del negocio, aunque este fuera un sistema previamente existente.

Actor	Descripción
Especialista	Encargado de la gestión de las estadísticas y los reportes. Este es quien los solicita y se encarga de realizar cualquier operación relacionada con estas tareas en el sistema.

Tabla # 3: Descripción de actores del sistema.

2.4 Listado de casos de uso del sistema (CUS)

- CUS 1** Generar reporte del nomenclador general de ubicación.
- CUS 2** Generar reporte del nomenclador general de equipos médicos.
- CUS 3** Generar reporte del nomenclador de unidad de salud.
- CUS 4** Generar reporte de unidades de salud existentes.

- CUS 5** Generar reporte de especialistas.
- CUS 6** Generar reporte de piezas.
- CUS 7** Generar reporte del inventario de equipos médicos por *países*.
- CUS 8** Generar reporte del inventario de equipos médicos por *regiones*.
- CUS 9** Generar reporte del inventario de equipos médicos por *provincias*.
- CUS 10** Generar reporte del inventario de equipos médicos por *especialidad*.
- CUS 11** Generar reporte del inventario de equipos médicos por *estado*.
- CUS 12** Generar reporte del inventario de equipos médicos afectados por *países*.
- CUS 13** Generar reporte del inventario de equipos médicos afectados por *regiones*.
- CUS 14** Generar reporte del inventario de equipos médicos afectados por *provincias*.
- CUS 15** Generar reporte del inventario de equipos médicos afectados por *especialidad*.
- CUS 16** Generar reporte personalizado de equipos médicos.
- CUS 17** Generar reporte personalizado de piezas.
- CUS 18** Generar reporte personalizado de especialistas.
- CUS 19** Generar reporte personalizado de mantenimiento.
- CUS 20** Generar reporte personalizado de órdenes de servicio.
- CUS 21** Listar equipos médicos.
- CUS 22** Exportar resultados a PDF.
- CUS 23** Generar resumen de equipos médicos en inventario general.
- CUS 24** Generar resumen de equipos médicos afectados en inventario general.
- CUS 25** Generar resumen de equipos médicos por países.
- CUS 26** Generar resumen de equipos médicos por regiones.
- CUS 27** Generar resumen de equipos médicos por provincias.
- CUS 28** Generar resumen de equipos médicos por estado.
- CUS 29** Generar resumen de equipos médicos por especialidades.

2.5 Diagrama de casos de uso del sistema



Figura # 4: Paquete Reportes generales.

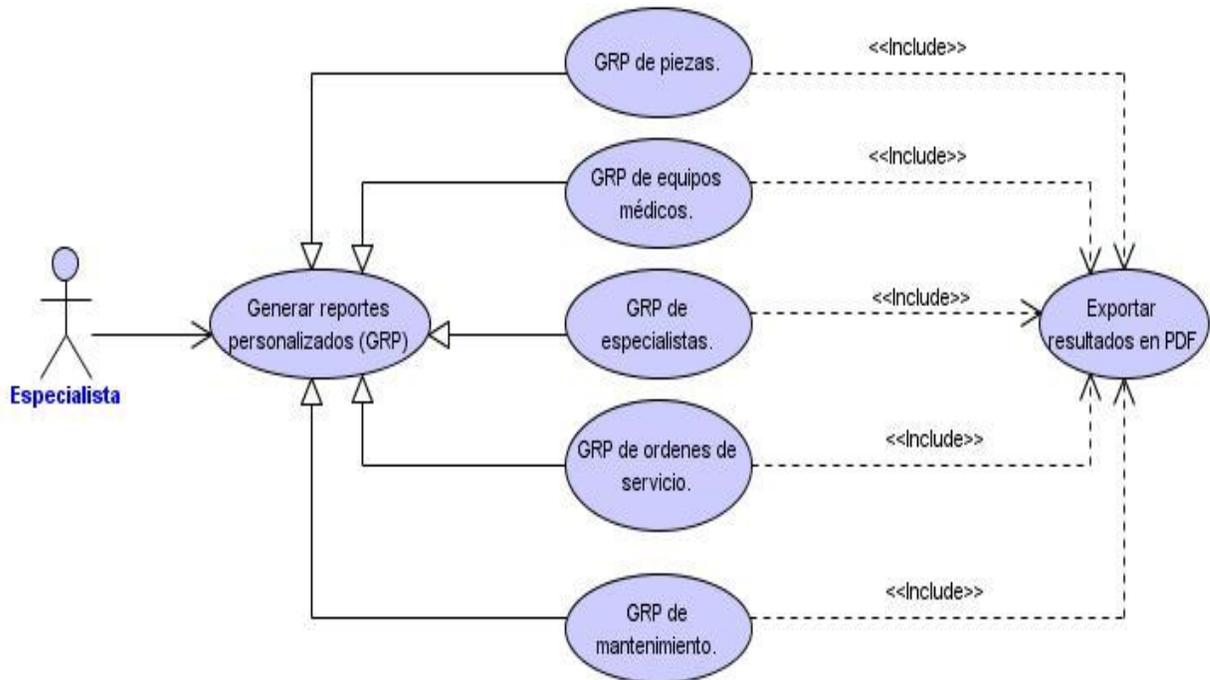


Figura # 5: Paquete Reportes personalizados.



Figura # 6: Paquete Reportes de inventario de equipos médicos.

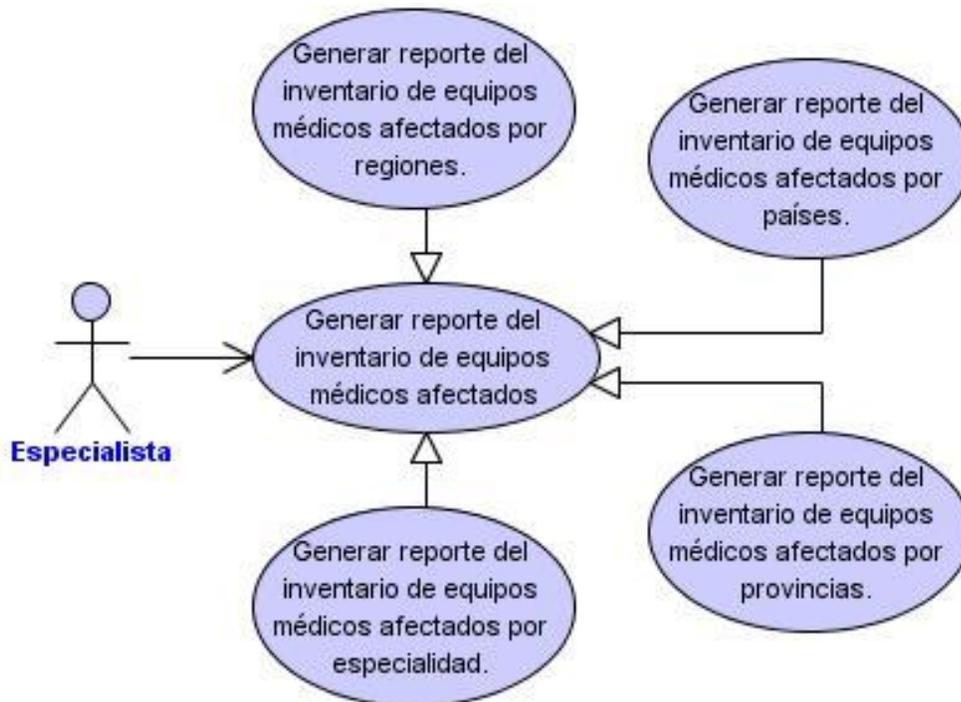


Figura # 7: Paquete Reportes de inventario de equipos médicos afectados.

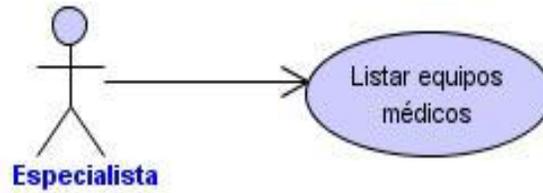


Figura # 8: Paquete Listar equipos médicos.



Figura # 9: Paquete Resúmenes estadísticos.

Caso de Uso:	Generar reporte personalizado de especialista.
Actores:	Especialista
Resumen:	El caso de uso inicia cuando el actor accede a la opción Reportes estadísticos en el menú inicio y presiona el icono que indica la acción Reporte personalizado de especialistas, el sistema brinda la posibilidad de introducir los criterios para generar el reporte, el actor introduce los datos que considera como criterios para generar el reporte, el sistema a partir de los criterios seleccionados y la consulta de las entidades correspondientes, genera el reporte de información. El actor puede imprimir el reporte, consultarlo o guardarlo según sus necesidades, el caso de uso termina
Referencias	RF 10, RF 32, RF 37, RF 38, RF 39.

Tabla # 4: CUS. Generar reporte personalizado de especialistas.

Caso de Uso:	Listar equipos médicos.
Actores:	Especialista.
Resumen:	El caso de uso inicia cuando el actor accede a la opción Reportes estadísticos en el menú inicio y presiona el icono que indica la acción Visualizador de Equipos Médicos, el sistema brinda la posibilidad de seleccionar país, provincia, municipio y unidad de salud del cual el usuario desea visualizar los equipos médicos existentes, el sistema a partir de los criterios seleccionados y la consulta de las entidades correspondientes, muestra los resultados.
Referencias	RF 30, RF 40.

Tabla # 5: CUS. Listar equipos médicos.

Caso de Uso:	Generar reporte del nomenclador general de ubicación.
Actores:	Especialista
Resumen:	El caso de uso inicia cuando el actor accede a la opción de Reportes estadísticos en el menú Inicio y presiona el icono que indica Reporte del nomenclador general de ubicación, el sistema genera el reporte. El actor puede consultarlo o guardarlo según sus necesidades, el caso de uso termina.
Referencias	RF 1, RF 2

Tabla # 6: CUS. Generar reporte del nomenclador general de ubicación.

Caso de Uso:	Generar reporte del inventario de equipos médicos.
Actores:	Especialista
Resumen:	El caso de uso inicia cuando el actor accede a la opción de Reportes estadísticos en el menú Inicio y presiona el icono que indica Reporte del inventario de equipos médicos, el sistema genera el reporte. El actor puede consultarlo o guardarlo según sus necesidades, el caso de uso termina.
Referencias	RF 13, RF 14, RF 15, RF 16, RF 17, RF 30.

Tabla # 7: CUS. Generar reporte del inventario de equipos médicos

.Caso de Uso:	Generar reporte del inventario de equipos médicos afectados.
Actores:	Especialista
Resumen:	El caso de uso inicia cuando el actor accede a la opción de Reportes estadísticos en el menú Inicio y presiona el icono que indica Reporte del inventario de equipos

	médicos afectados, el sistema genera el reporte. El actor puede consultarlo o guardarlo según sus necesidades, el caso de uso termina.
Referencias	RF 18, RF 19, RF 20, RF 21, RF 30.

Tabla # 8: CUS. Generar reporte del inventario de equipos médicos afectados.

Mediante la descripción del modelado del negocio se han presentado el actor y el trabajador del negocio, definiéndose el caso de uso del negocio y su relación con los mismos. Además, del diagrama de casos de uso del sistema que engloba los requisitos funcionales definidos a raíz de previa conciliación con el cliente, así como una breve descripción de los casos de usos definidos.

CAPÍTULO III: ANÁLISIS Y DISEÑO DEL SISTEMA

En el presente capítulo se exponen el modelo del análisis y el modelo del diseño. Mostrándose los diagramas correspondientes a clases persistentes, el modelo de datos con la descripción de las tablas que lo componen y de despliegue del sistema. Se ofrece además una panorámica de la arquitectura y los patrones de diseño utilizados.

3.1 Modelo de análisis

Este modelo brinda una comprensión detallada de los requisitos que fueron identificados durante la disciplina de gestión de requisitos, razón por la cual impone una estructura del sistema que debe ser mantenida lo más fielmente posible durante la implementación del sistema. Es un modelo conceptual que se encuentra centrado en la abstracción del sistema a desarrollar y posibilita el entendimiento de varios aspectos de la implementación. No tiene en cuenta aquellos elementos relacionados con el diseño pues no toma en consideración las tecnologías a utilizar por el grupo de desarrollo del software; se puede afirmar que es un prototipo de las clases que posteriormente servirán para dar vida al mismo.

Describe las características y comportamiento comunes de un conjunto de cosas que existen en el sistema. Durante su concepción se presentan tres estereotipos conceptuales, que se encargan de identificar a las clases del análisis, estos son:

- × **Interfaz:** Se encargan de la modelación de toda la interacción que puede existir entre los actores y el sistema; constituyen las fronteras del mismo.
- × **Control:** Representan la coordinación, secuenciación, transacciones y a veces la lógica del negocio; se emplean a menudo para encapsular el control referido a un CU.
- × **Entidad:** representa la información de larga duración y a menudo persistente que se maneja en el sistema.¹⁶

¹⁶ Ingeniería de Software 2 (2010-2011) **Disciplina de Análisis y Diseño**. [Consultado: 03/02/2011] [En línea] http://eva.uci.cu/file.php/259/Curso_2009-2010/Conferencia_1/Materiales_basicos/Conferencia_1.doc

Diagramas de clases del análisis

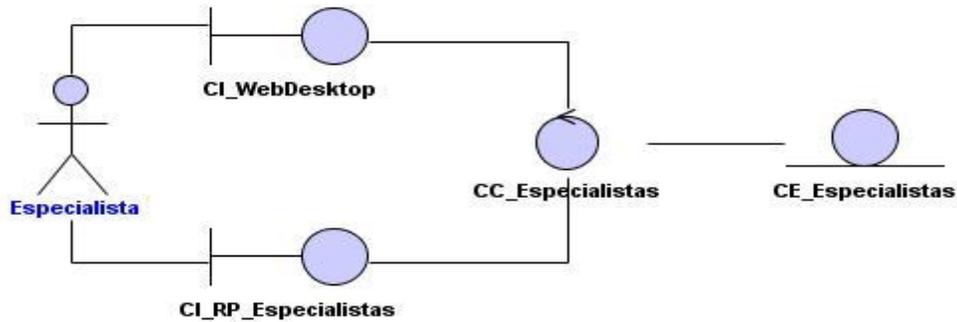


Figura # 10: Diagrama de clases del análisis. Caso de uso: Generar reporte personalizado de especialistas.

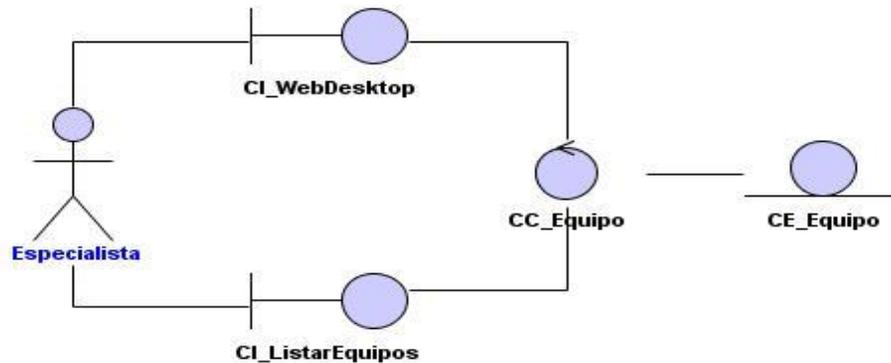


Figura # 11: Diagrama de clases del análisis. Caso de uso: Listar equipos médicos.



Figura # 12: Diagrama de clases del análisis. Caso de uso: Generar reporte del nomenclador general de ubicación.



Figura # 13: Diagrama de clases del análisis. Caso de uso: Generar reporte del inventario de equipos médicos por países.



Figura # 14: Diagrama de clases del análisis. Caso de uso: Generar reporte del inventario de equipos médicos afectados por países.

Diagramas de interacción. Comunicación



Figura # 15: Diagrama de comunicación. Caso de uso: Generar reporte personalizado de especialistas.

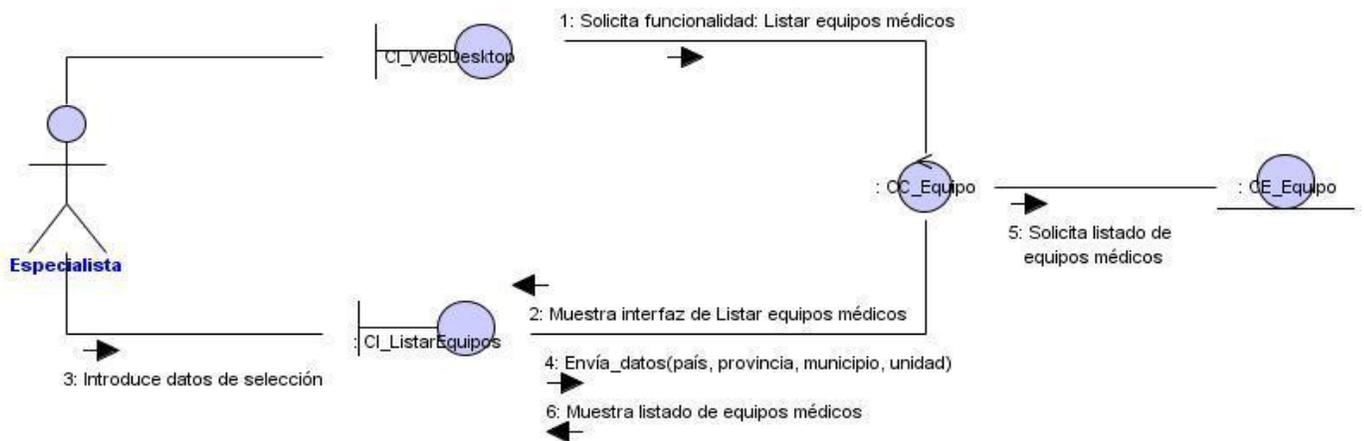


Figura # 16: Diagrama de comunicación. Caso de uso: Listar equipos médicos.

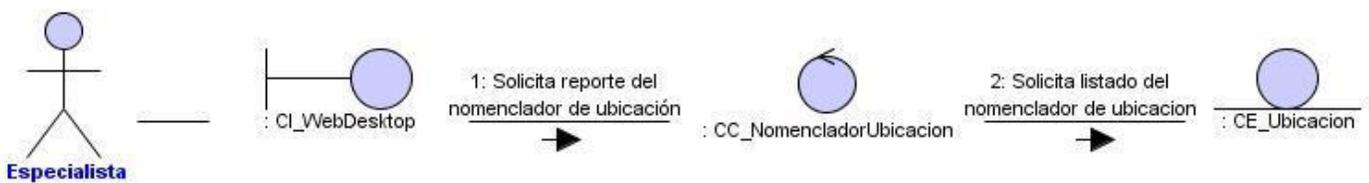


Figura # 17: Diagrama de comunicación. Caso de uso: Generar reporte del nomenclador general de ubicación.

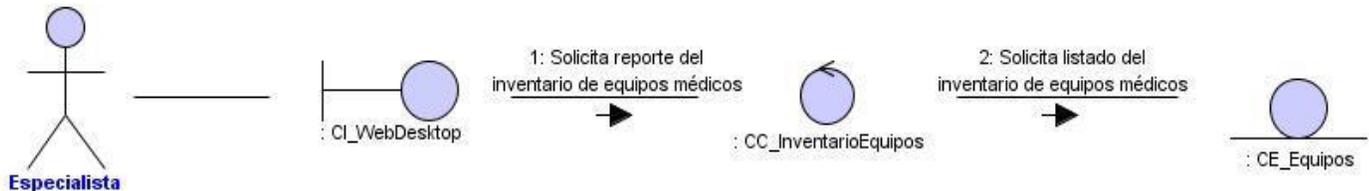


Figura # 18: Diagrama de comunicación. Caso de uso: Generar reporte del inventario de equipos médicos por países.



Figura # 19: Diagrama de comunicación. Caso de uso: Generar reporte del inventario de equipos médicos afectados por países.

3.2 Modelo de diseño

El diseño toma en cuenta los requisitos funcionales y no funcionales para crear un plano del modelo de implementación; para la consolidación de una arquitectura sólida y estable, suficiente para lograr que el sistema sea implementado sin errores. Este modelo es específico para cada implementación, por lo que no se puede realizar de manera genérica para distintos software. Propone una estructura que debe ser conservada lo más exacto posible cuando se de forma a la aplicación, este sirve de entrada al flujo de trabajo de implementación. En general se puede plantear que:

“El modelo de diseño es una abstracción de la implementación del sistema. Se utiliza para concebir y para documentar el diseño del sistema de software. Es un producto de trabajo integral y compuesto que abarca todas las clases de diseño, subsistemas, paquetes, colaboraciones y las relaciones entre ellos”.¹⁷

Para la representación de elementos específicos del diseño de aplicaciones web son utilizados los estereotipos web. La extensión UML para web establece como elementos significativos, tres clases estereotipadas de la siguiente manera:

- × **Página servidora (Server page):** encargada de la representación de la página que contiene el código que es ejecutado en el servidor. Esta lleva a cabo las acciones correspondientes a construir y generar el resultado HTML, además de realizar peticiones a la capa inferior.
- × **Página cliente (Client Page):** presenta el formato XHTML, mezcla datos, presentación y lógica, interpretados por el navegador.
- × **Formulario (Form):** son una colección de elementos de entrada que son parte de una página cliente. Se relacionan directamente con la etiqueta de igual nombre del HTML. Sus atributos son los elementos de entrada del formulario.

¹⁷ Rational Unified Process. **Extended Help for RUP**. 2003.

En busca de concebir un diagrama de clases del diseño con poca complejidad de interpretación y mayor organización; estos fueron estructurados de forma tal que los elementos del diseño se encuentren localizados en la capa a la que pertenece según se define en el patrón Modelo-Vista-Controlador. Los elementos relacionados con el modelo, la vista y el controlador fueron separados.

Tomándose en cuenta que, se hace uso de Symfony, es necesario delimitar que elementos del diseño corresponden al framework y que a la implementación propia del caso de uso. En aras de lograr dicha separación se engloban de manera general en un subsistema denominado “Componentes de Symfony”, todos aquellos componentes propios de Symfony que actúan o intervienen. Estos al ser propios de este trabajan de manera transparente al programador.

Situación semejante a la descrita anteriormente sucede con los componentes propios que han sido implementados para el SIGICEM. Los desarrolladores de este sistema han definido varias funcionalidades que aminoran la complejidad de integración de Symfony con ExtJS, además de algunas que permiten la importación y exportación de documentos en formatos XLS y PDF, entre otras funcionalidades. Debido a esto se concibe la creación del subsistema “Componentes de SIGICEM”, en el que se aglomeren estas funcionalidades. Ambos subsistemas se encontraran representados dentro del subsistema “Componentes”, del cual se consumirán las prestaciones deseadas.

En la capa correspondiente a la vista, se podrán encontrar los paquetes “Componentes CSS” y “Componentes JS”. Estos simbolizan las Hojas de Estilo en Cascada y JavaScript, de las que se hacen uso, para definir y obtener mejoras de los atributos visuales propios de la aplicación en desarrollo. Mientras que en la capa que representa al modelo se presentan las entidades t_Reporte y t_ReportePeer, que representan de manera genérica todas las entidades del modelo de datos de SIGICEM que interactúan con cada uno de los casos de usos. Se ha decidido realizar esta generalización, producto al alto número de entidades que deberían ser modeladas, pues en caso de hacerse se entorpecería la claridad del diagrama.

Diagramas de clases del diseño

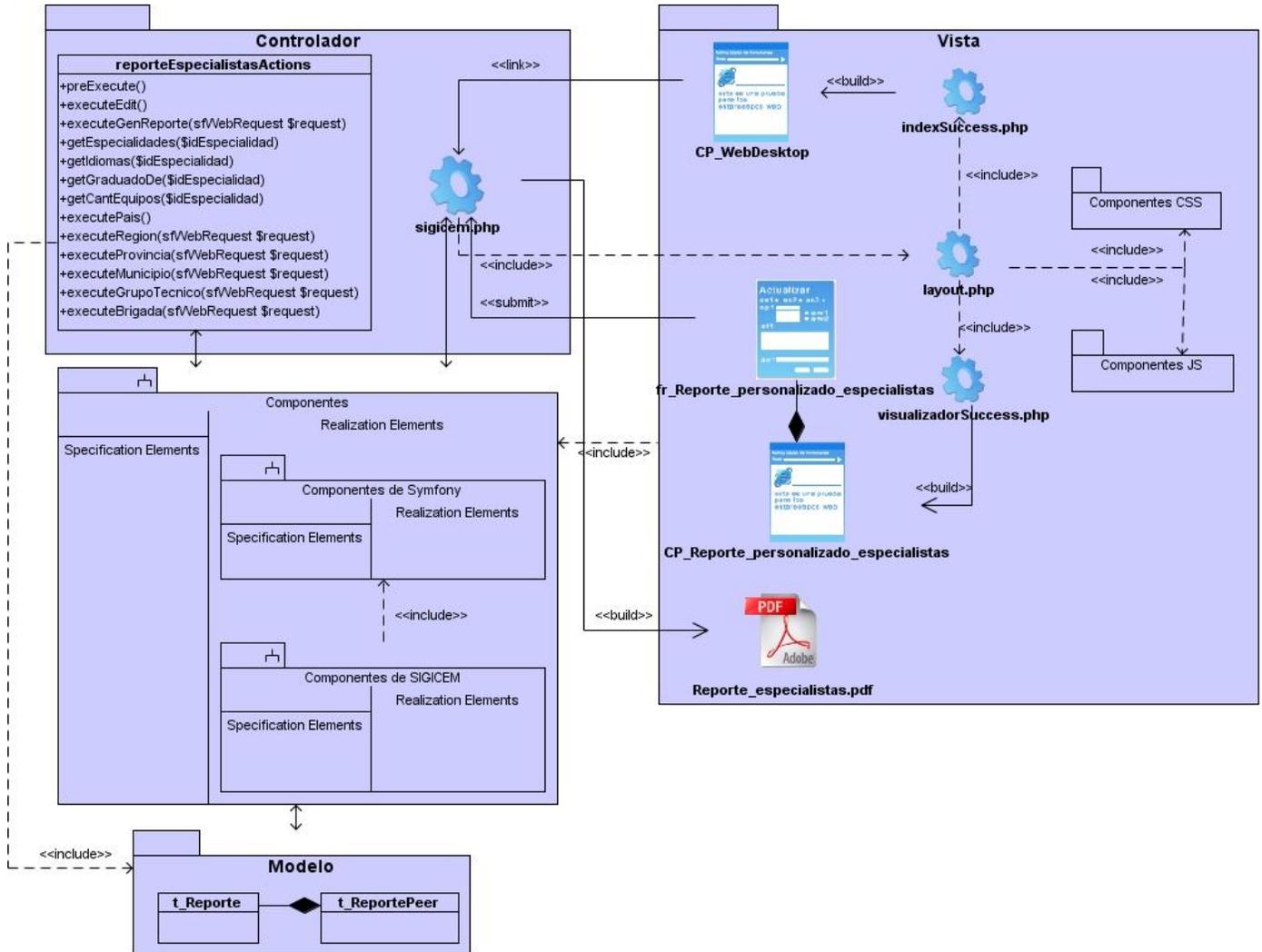


Figura # 20: Diagrama de clases del diseño. Caso de uso: Generar reporte personalizado de especialistas.

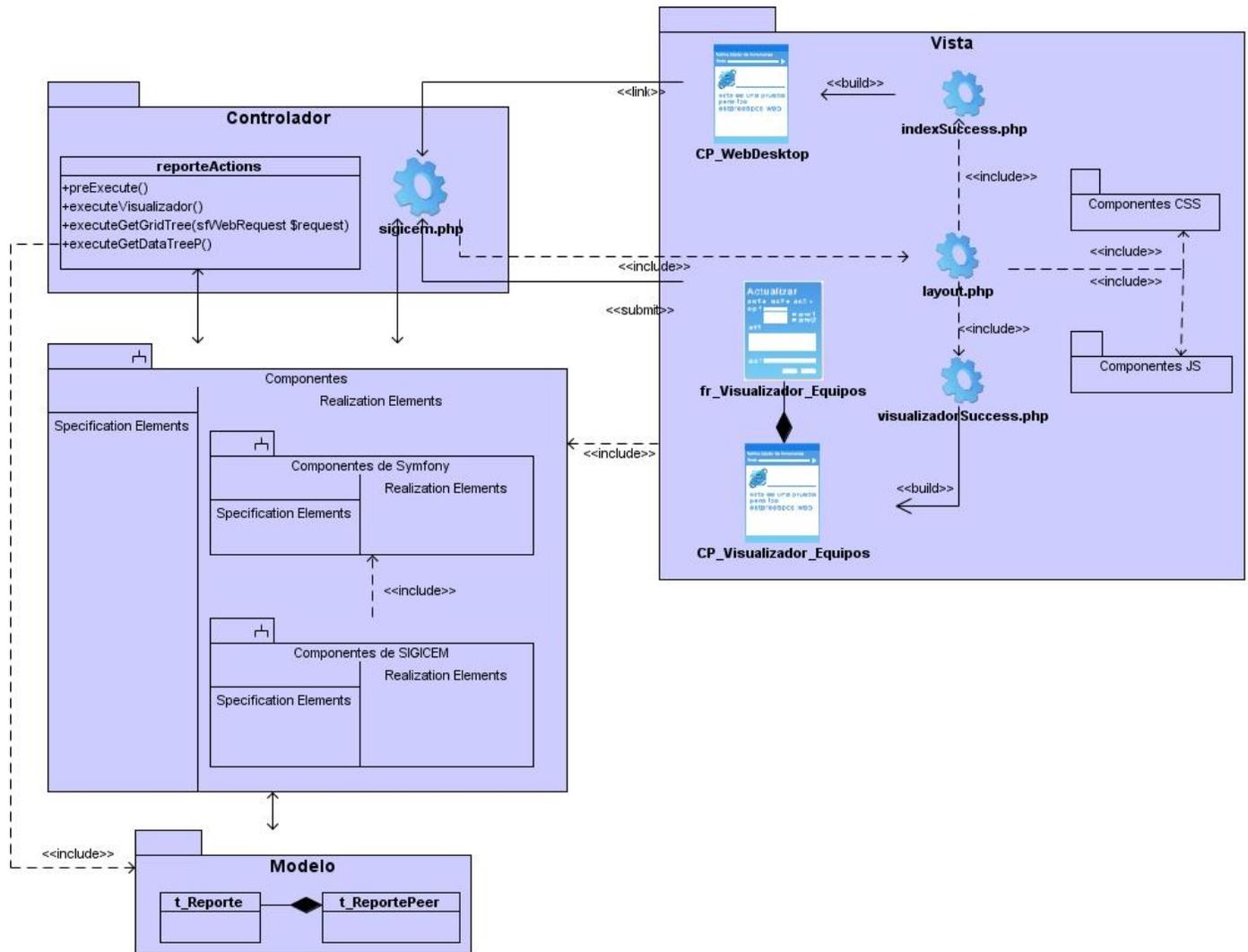


Figura # 21: Diagrama de clases del diseño. Caso de uso: Listar equipos médicos.

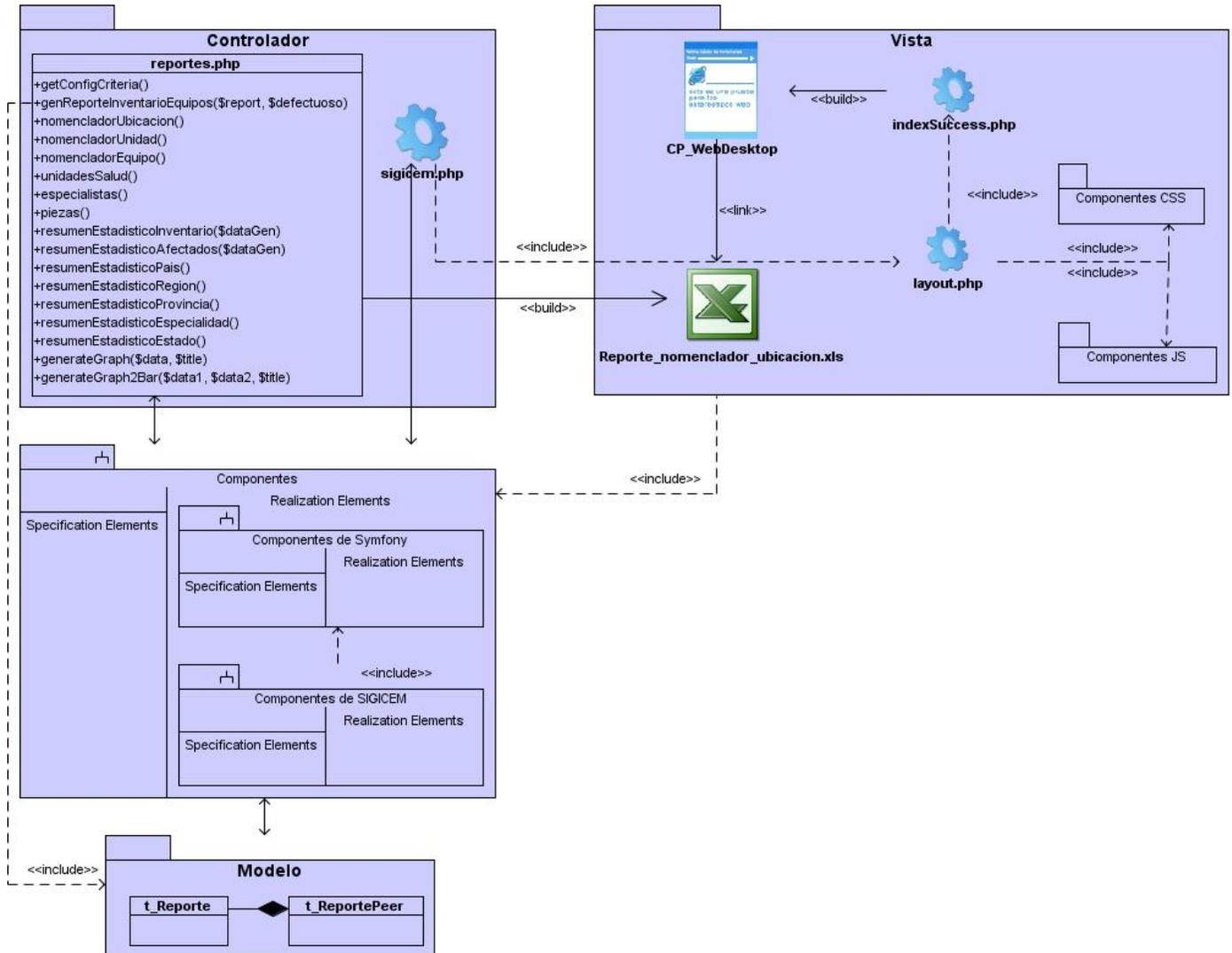


Figura # 22: Diagrama de clases del diseño. Caso de uso: Generar reporte del nomenclador general de ubicación.

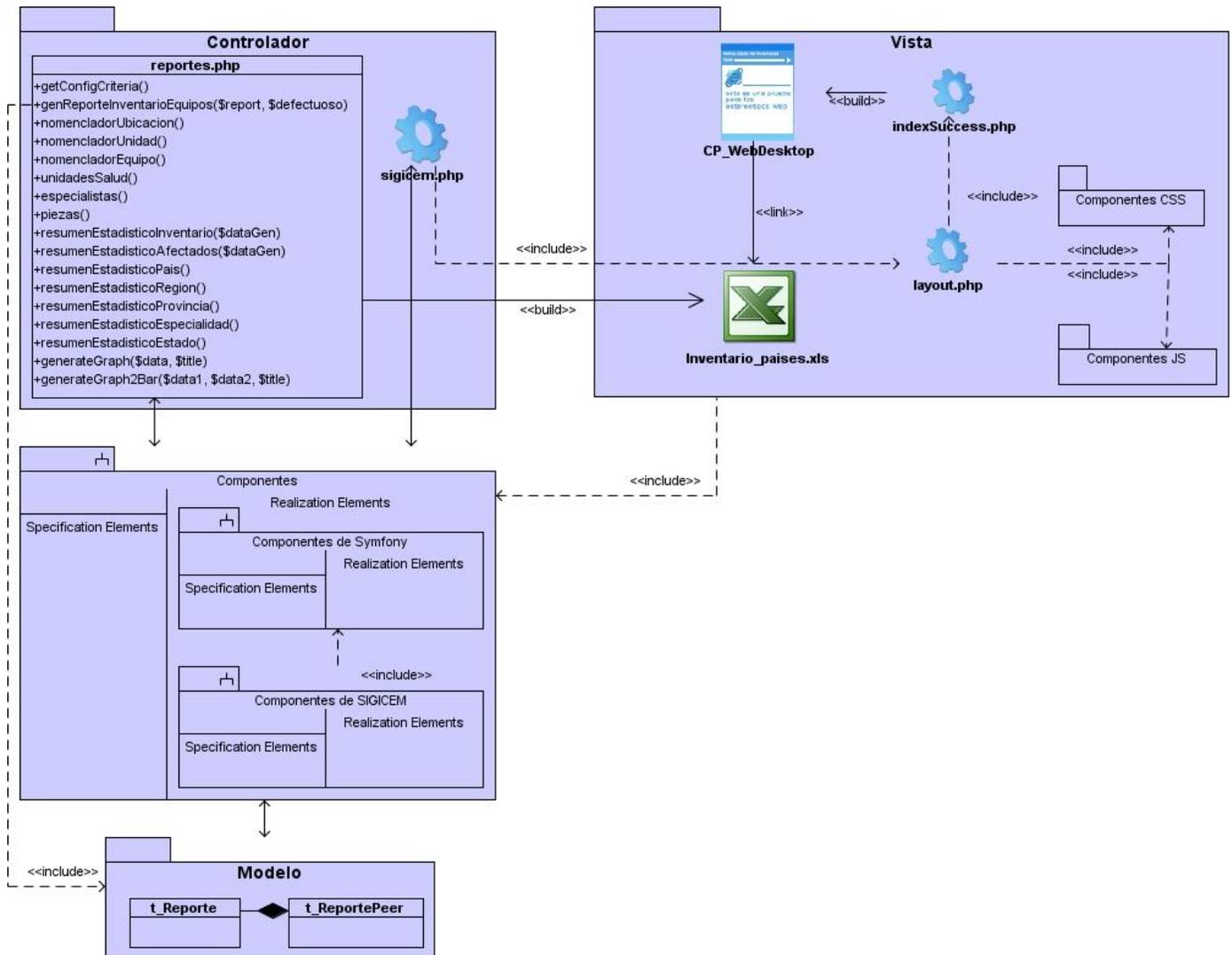


Figura # 23: Diagrama de clases del diseño. Caso de uso: Generar reporte del inventario de equipos médicos por países.

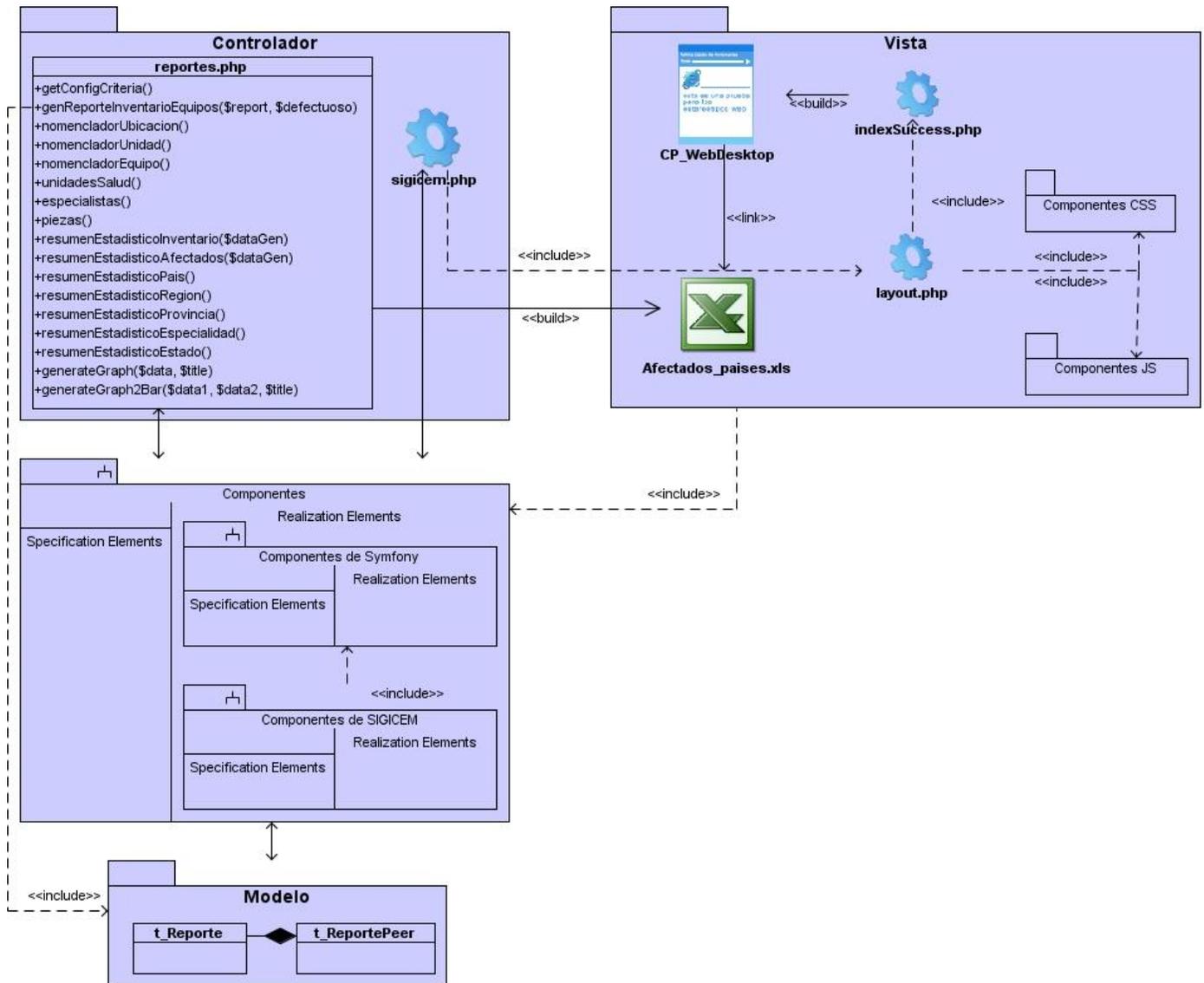


Figura # 24: Diagrama de clases del diseño. Caso de uso: Generar reporte del inventario de equipos médicos afectados por países.

Descripción de las clases del diseño

Nombre de la Clase:	<i>sigicem.</i>
Descripción general:	
<p>La página servidora sigicem es el punto de entrada único de toda la aplicación, denominada comúnmente controlador frontal. Todas las peticiones realizadas son gestionadas mediante esta clase. Hace uso del sistema de enrutamiento para realizar la asociación del nombre de una acción y el nombre de un módulo mediante la URL definida por el usuario.</p>	

Nombre de la Clase:	<i>layout.</i>
Descripción general:	
La página servidora layout también conocida como plantilla global, posee en su interior todo el código HTML común a todas las páginas de la aplicación. De esta forma es posible evitar repetirlo en cada una de las páginas que se tengan. Se encuentra ubicada en la capa Vista propuesta por el patrón MVC.	
Nombre de la Clase:	<i>indexSuccess.</i>
Descripción general:	
La página servidora indexSuccess presenta el código HTML, CSS y JavaScript común a todas las páginas de la aplicación para evitar repetirlo en cada una. Es utilizada para definir la interfaz principal de la aplicación. Se encuentra ubicada en la capa Vista propuesta por el patrón MVC.	
Nombre de la Clase:	<i>editSuccess.</i>
Descripción general:	
La página editSuccess contiene el código de presentación, es el resultado de una acción a la que previamente esta asociada por el nombre seguido del resultado. Se encuentra ubicada en la capa Vista propuesta por el patrón MVC.	
Nombre de la Clase:	<i>CP_WebDesktop.</i>
Descripción general:	
La página cliente CP_WebDesktop representa el acceso del usuario al sistema, esta se encarga de atender todas las selecciones y peticiones que pueda realizar un determinado usuario. Se encuentra ubicada en la capa Vista propuesta por el patrón MVC.	
Nombre de la Clase:	<i>CP_Reporte_personalizado_equipo</i>
Descripción general:	
La página cliente CP_Reporte_personalizado_equipo se encarga de atender todas las selecciones y peticiones que pueda realizar un determinado usuario. Se encuentra ubicada en la capa Vista propuesta por el patrón MVC. Esta representa el acceso del usuario a la interfaz visual del CUS 16. Generar reporte personalizado de equipos médicos.	
Nombre de la Clase:	<i>fr_Reporte_personalizado_equipo</i>
Descripción general:	
El formulario fr_Reporte_personalizado_equipo representa la entrada y salida de datos en el sistema, a través de esta el usuario podrá introducir los datos necesarios para la posterior generación del reporte personalizado de equipos médicos. Se encuentra ubicada en la capa Vista propuesta por el patrón MVC. Representa el acceso del usuario a la interfaz visual del CUS 16. Generar reporte personalizado de equipos médicos.	
Nombre de la Clase:	<i>reporteEquipoActions.</i>
Descripción general:	
La clase servidora reporteEquipoActions contiene parte de la lógica de la aplicación del módulo reporteEquipo, la misma se encuentra compuesta por varios métodos y acciones que a su vez utilizan el modelo y define las variables para la vista. Es utilizada para dar respuesta al: CUS 16. Generar reporte personalizado de equipos médicos.	
Nombre de la Clase:	<i>reportePiezasActions.</i>
Descripción general:	
La clase servidora reportePiezasActions contiene parte de la lógica de la aplicación del módulo reporteEquipo, la misma se encuentra compuesta por varios métodos y acciones que a su vez utilizan el modelo y define las	

variables para la vista. Es utilizada para dar respuesta al: <ul style="list-style-type: none"> • CUS 17. Generar reporte personalizado de piezas. 	
Nombre de la Clase:	<i>reporteEspecialistaActions.</i>
Descripción general:	
La clase servidora <i>reporteEspecialistaActions</i> contiene parte de la lógica de la aplicación del módulo <i>reporteEquipo</i> , la misma se encuentra compuesta por varios métodos y acciones que a su vez utilizan el modelo y define las variables para la vista. Es utilizada para dar respuesta al: <ul style="list-style-type: none"> • CUS 18. Generar reporte personalizado de especialistas. 	
Nombre de la Clase:	<i>reporteMantenimientoActions.</i>
Descripción general:	
La clase servidora <i>reporteMantenimientoActions</i> contiene parte de la lógica de la aplicación del módulo <i>reporteEquipo</i> , la misma se encuentra compuesta por varios métodos y acciones que a su vez utilizan el modelo y define las variables para la vista. Es utilizada para dar respuesta al: <ul style="list-style-type: none"> • CUS 19. Generar reporte personalizado de mantenimiento. 	
Nombre de la Clase:	<i>reporteOsActions.</i>
Descripción general:	
La clase servidora <i>reporteOsActions</i> contiene parte de la lógica de la aplicación del módulo <i>reporteEquipo</i> , la misma se encuentra compuesta por varios métodos y acciones que a su vez utilizan el modelo y define las variables para la vista. Es utilizada para dar respuesta al: <ul style="list-style-type: none"> • CUS 20. Generar reporte personalizado de órdenes de servicio. 	
Nombre de la Clase:	<i>reporte.</i>
Descripción general:	
La clase servidora <i>reporte</i> contiene la lógica de la aplicación del módulo <i>reporteEquipo</i> , la misma encuentra compuesta por varios métodos y acciones que a su vez utilizan el modelo y define las variables para la vista. Es utilizada para dar respuesta a los siguientes casos de uso: <ul style="list-style-type: none"> • CUS 1. Generar reporte del nomenclador general de ubicación. • CUS 2. Generar reporte del nomenclador general de equipos médicos • CUS 3. Generar reporte del nomenclador de unidad de salud • CUS 4. Generar reporte de unidades de salud existentes • CUS 5. Generar reporte de especialistas • CUS 6. Generar reporte de piezas • CUS 7. Generar reporte del inventario de equipos médicos por países. • CUS 8. Generar reporte del inventario de equipos médicos por regiones. • CUS 9. Generar reporte del inventario de equipos médicos por provincias. • CUS 10. Generar reporte del inventario de equipos médicos por especialidad. • CUS 11. Generar reporte del inventario de equipos médicos por estado. • CUS 12. Generar reporte del inventario de equipos médicos afectados por países. • CUS 13. Generar reporte del inventario de equipos médicos afectados por regiones. • CUS 14. Generar reporte del inventario de equipos médicos afectados por provincias. • CUS 15. Generar reporte del inventario de equipos médicos afectados por especialidad. • CUS 23. Generar resumen de equipos médicos en inventario general. 	

- CUS 24. Generar resumen de equipos médicos afectados en inventario general.
- CUS 25. Generar resumen de equipos médicos por países.
- CUS 26. Generar resumen de equipos médicos por regiones.
- CUS 27. Generar resumen de equipos médicos por provincias.
- CUS 28. Generar resumen de equipos médicos por estado.
- CUS 29. Generar resumen de equipos médicos por especialidades.

Tabla # 9: Descripción de las clases empleadas en el los diagramas de clases del diseño.

3.3 Patrón arquitectónico

El Modelo Vista Controlador (MVC) es un patrón de arquitectura de software, este separa los datos de la aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos: Modelo, Vista y Controlador. Este modelo especifica un conjunto de subsistemas con sus responsabilidades y una serie de consideraciones a tener en cuenta para organizar los distintos componentes. Su popularidad se debe a que permite organizar aplicaciones grandes de manera mucho más fácil. Con su implementación se logra obtener una separación entre los distintos componentes que permite lograr mayor sencillez para crear distintas representaciones de los mismos datos. Incrementa en gran medida la reutilización y flexibilidad, así como que las soluciones desarrolladas suelen ser más escalables.

En la solución propuesta es el MVC el patrón arquitectónico empleado, como se puede deducir de su nombre, consiste en separar de la mejor forma posible las capas de:

- × **Modelo:** contiene la lógica de negocio, la base de datos se encuentra almacenada en esta capa, en la que se engloban los datos y las funcionalidades. Se comporta de manera independiente a cualquier representación de salida y comportamiento de entrada, lo que asegura la integridad de los datos y permite derivar nuevos.
- × **Vista:** es la parte que utilizan los usuarios para interactuar con la aplicación, simplemente la página HTML que se muestra al usuario. Esta capa se centra en presentar al modelo en un formato adecuado para el intercambio con el usuario final. Los gestores de plantillas se incluyen dentro de esta.
- × **Controlador:** se enfoca en realizar las llamadas necesarias al modelo para capturar los datos y devolverlos a la vista para que en esta sean mostrados al usuario. Este responde a eventos,

usualmente acciones del usuario e invoca cambios en el modelo. En resumen, se encarga de asimilar las peticiones del usuario y realizar los cambios apropiados en el modelo o en la vista.

En la siguiente figura se observa la interacción que se efectúa entre las distintas capas, específicamente en el framework Symfony que es el que se emplea en la presente investigación:

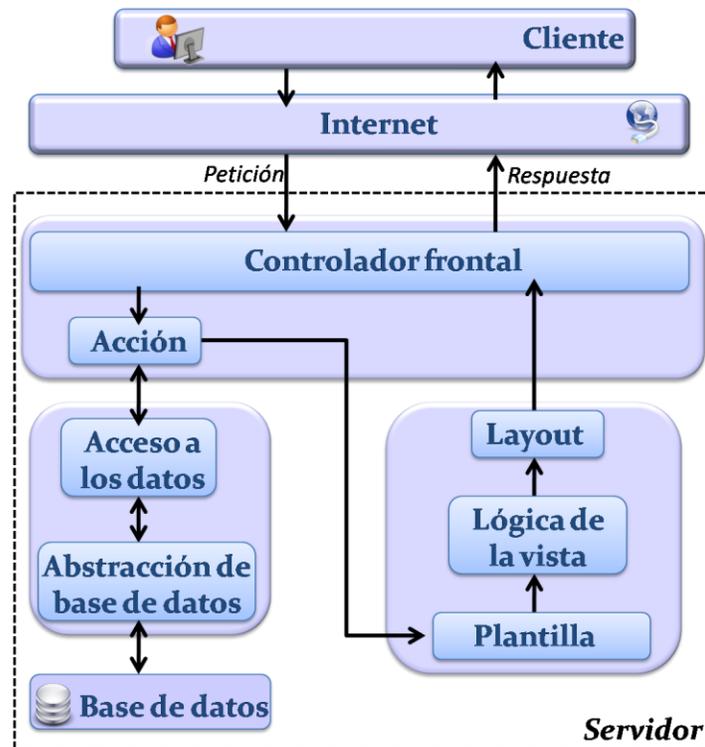


Figura # 25: Comportamiento del MVC en Symfony.

Las capas del MVC que implementa Symfony, se encuentran compuestas por siete scripts fundamentalmente, los cuales se distribuyen de la siguiente forma: en el modelo se encuentra la abstracción de la base de datos y el acceso a los datos; en la vista se podrá encontrar la vista, las plantillas y los layouts correspondientes; mientras que en el controlador se cuenta con el controlador frontal y las acciones.

Propel, como ORM, se encarga de generar de forma automática las clases de la capa del modelo en dependencia de la estructura de datos de la aplicación, crea métodos especiales para acceder y modificar los datos. La abstracción de la base de datos se gestiona de manera transparente para el programador,

con la utilización Objetos de Datos de PHP (PDO, PHP Data Objects), razón por la cual es posible migrar de sistema gestor de bases de datos. Por su parte la lógica de la vista se puede transformar en un archivo de configuración sencillo, sin necesidad de que esto implique grandes cambios, debido a que generalmente, el layout es global en toda la aplicación o en un grupo de páginas. La plantilla sólo se encarga de dar salida a las variables definidas en el controlador.

El controlador tiene como principal objetivo el manejo de las peticiones del usuario, de la seguridad, cargar la configuración de la aplicación y otras tareas. Este se genera de forma automática, dividiéndose, normalmente en un controlador frontal, que es único para cada aplicación, y las acciones, que incluyen el código específico del controlador de cada página. Este ofrece un punto de entrada único para toda la aplicación, por lo que en caso de ser necesario impedir el acceso a la aplicación, solamente se debe editar el script correspondiente al mismo. El controlador frontal y el layout, generalmente son comunes para todas las acciones de la aplicación.

3.4 Patrones de diseño

Varios son los patrones arquitectónicos, de análisis y diseño existentes actualmente, durante la definición de la arquitectura se precisan aquellos que serán utilizados. El arquitecto encargado de este proceso no se centra en la concepción de nuevos patrones, sino que, reutiliza patrones para definir la estructura y el comportamiento interno y externo de un sistema. Los patrones de diseño conforman una estructura para optimizar el desarrollo de subsistemas o componentes. Estos permiten dar solución a problemas de diseño generales en una situación particular. Su utilización hace posible identificar: clases, instancias, roles, colaboraciones y la distribución de responsabilidades, además de ayudar a construir clases y a estructurar sistemas de clases.

Durante el transcurso del capítulo 1 de la presente investigación se definió el uso del framework Symfony, este se encuentra concebido de manera tal que puede afirmarse que obliga al uso de varios patrones. Se puede hacer referencia a dos grupos esenciales, los Patrones Generales de Software para Asignar Responsabilidades (GRASP) y los Patrones Gof (Gang of four o Banda de los cuatros). Dentro de estos se encuentran diversas variantes que pueden ser aplicadas según las necesidades reales que se tengan, seguidamente se señalan y ejemplifican aquellas que han sido consideradas a utilizar:

- × **Experto:** Al incluir Propel como ORM, este genera las clases para la gestión de las entidades con las responsabilidades debidamente asignadas. Esta es precisamente la solución que propone el patrón Experto, pues cada una de estas clases cuenta con un conjunto de funcionalidades relacionadas directamente con la entidad que representan.
- × **Bajo acoplamiento:** En la capa del Modelo se encuentran las clases que implementan la lógica de negocio y de acceso a datos, estas clases tienen pocas asociaciones con otras de la Vista o el Controlador por lo que la dependencia en este caso es baja, poniéndose de manifiesto el patrón Bajo Acoplamiento.¹⁸
- × **Controlador:** Este se basa en la existencia de un intermediario entre las páginas clientes y el algoritmo que responde a las peticiones realizadas por estas. La existencia del controlador frontal, es el ejemplo básico que evidencia de forma clara su utilización en Symfony. Este maneja las peticiones del usuario, la seguridad, carga la configuración de la aplicación y otras tareas, es único para cada aplicación. En busca de aminorar un poco la carga que este posee se cuentan con las acciones que contienen las especificaciones de cada página.
- × **Decorador:** Symfony presenta el denominado archivo layout.php o también conocido como plantilla global, en la que convergen todos los elementos comunes a cada una de las páginas del sistema en construcción. Este fichero se complementa con las plantillas, decorándolas y obteniéndose la interfaz final que será mostrada al usuario.

Aplicar estos patrones posibilita la reutilización del diseño y logra identificar aspectos claves de la estructura de un diseño que puede ser aplicado en varias situaciones. De esta manera se reducen los esfuerzos de desarrollo y mantenimiento, mejora la seguridad, eficiencia y consistencia del diseño. Además de elevar la flexibilidad, modularidad y extensibilidad del software que se encuentre en desarrollo.

3.5 Modelo de despliegue

En el modelo de despliegue se encuentra definida la ubicación física con la que contará la aplicación, se indican cuáles son los enlaces de comunicación existentes entre los distintos componentes en tiempo de

¹⁸ Machado Díaz, Maidelys, Coca Ribas, Yader Luis. **SACCEM: Módulo de gestión de la información del Departamento de Supervisión del CCEEM versión 1.0.** Universidad de las Ciencias Informáticas, 2009. [Consultado: 09/03/2011.]

ejecución. Este modelo tiene como objetivo recolectar toda la configuración de los elementos de proceso y las conexiones entre estos elementos. Varios de los integrantes del equipo de trabajo durante la construcción del software, utilizan el modelo de despliegue. Ejemplo de esto se puede mencionar el arquitecto de software, para capturar y entender el entorno de ejecución física del sistema y su distribución; los diseñadores de software y de bases de datos, para asimilar la distribución del proceso y los datos en el sistema y el gestor de proyectos, en la estimación de costes, y otras actividades.

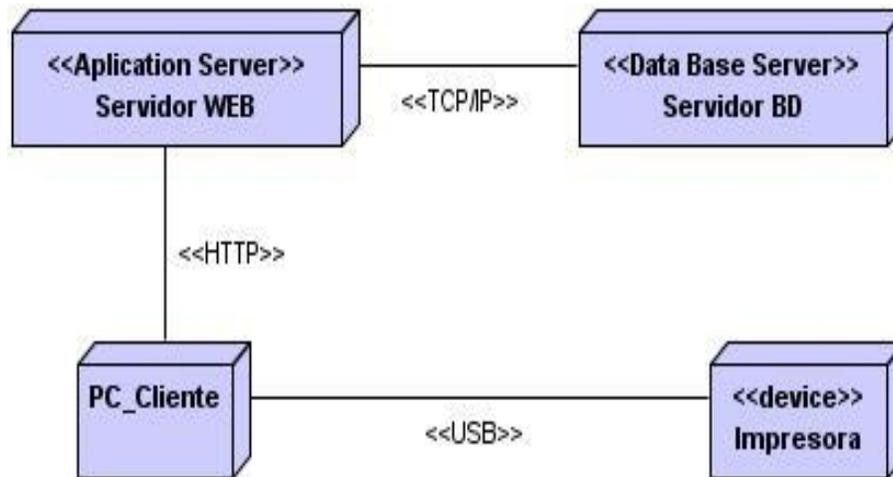


Figura # 26: Diagrama de despliegue del Módulo de Reportes Estadísticos.

En este capítulo se ilustraron los elementos más significativos del modelado del análisis y el diseño del sistema en desarrollo, a partir de los casos de uso especificados en el capítulo # 2. Se representaron los diagramas de clases del análisis, diagramas de comunicación; especificándose las clases del diseño. Se identificaron los patrones de diseño y las características propias de la arquitectura a emplear.

CAPÍTULO IV: IMPLEMENTACIÓN Y PRUEBAS DEL SISTEMA

En este capítulo serán abarcados algunas de las especificaciones de las Disciplinas de Implementación y Prueba, mediante la descripción de sus principales artefactos. Se confeccionará el modelo de implementación con los diagramas de componentes, en correspondencia a los casos de uso definidos anteriormente. Se definen los estándares de codificación a utilizar y algunos de los elementos del tratamiento de errores para sentar las bases de la implementación del sistema. Se elaborara el Modelo de Pruebas necesario para la posterior realización de estas al Módulo de Reportes Estadísticos.

4.1 Modelo de datos

Este se enfoca en la representación lógica y física de los datos persistentes de los cuales se hará uso para el desarrollo y funcionamiento del módulo propuesto. Se utiliza para definir la correlación entre las clases de diseño y las estructuras de datos persistentes. Específicamente, para la investigación que se encuentra en desarrollo no se realizará la confección de este artefacto, producto a que se adopta de forma íntegra el modelo de datos previamente definido para el SIGICEM. Esta afirmación se hace en consecuencia de que para la implementación del módulo de Reportes Estadísticos no es preciso añadir entidad alguna. A partir de las previamente definidas se obtienen los datos adecuados para dar respuestas a las necesidades del usuario final.

El modelo de datos existente gestiona toda la información referente a los distintos procesos que han sido informatizados en el sistema. Actualmente, se encuentra integrado por 105 entidades, que dan soporte a la gestión de nomencladores, de ubicación de los equipos médicos, instituciones de salud y especialistas. Además se presentan informaciones relacionadas con grupos técnicos, la captación de datos propios de los equipos médicos, los reportes de piezas, las órdenes de despacho, los principales datos de los especialistas, las solicitudes de compra, así como la relación de mantenimientos planificados y brindados en las distintas unidades de salud existentes al equipamiento médico presente en estas, así como otras informaciones.

4.2 Modelo de implementación

El modelo de implementación representa la composición física de la implementación en términos de subsistemas de implementación, y elementos de implementación (directorios y archivos, incluyendo código fuente, datos y archivos ejecutables).¹⁹

Mediante la confección de este se identifican los componentes físicos de la implementación para que de esta manera puedan comprenderse y gestionarse de una mejor forma. Se definen además, las principales unidades de integración, así como las unidades que podrán ser versionadas, desplegadas y reemplazadas por separado.

Diagrama de componentes

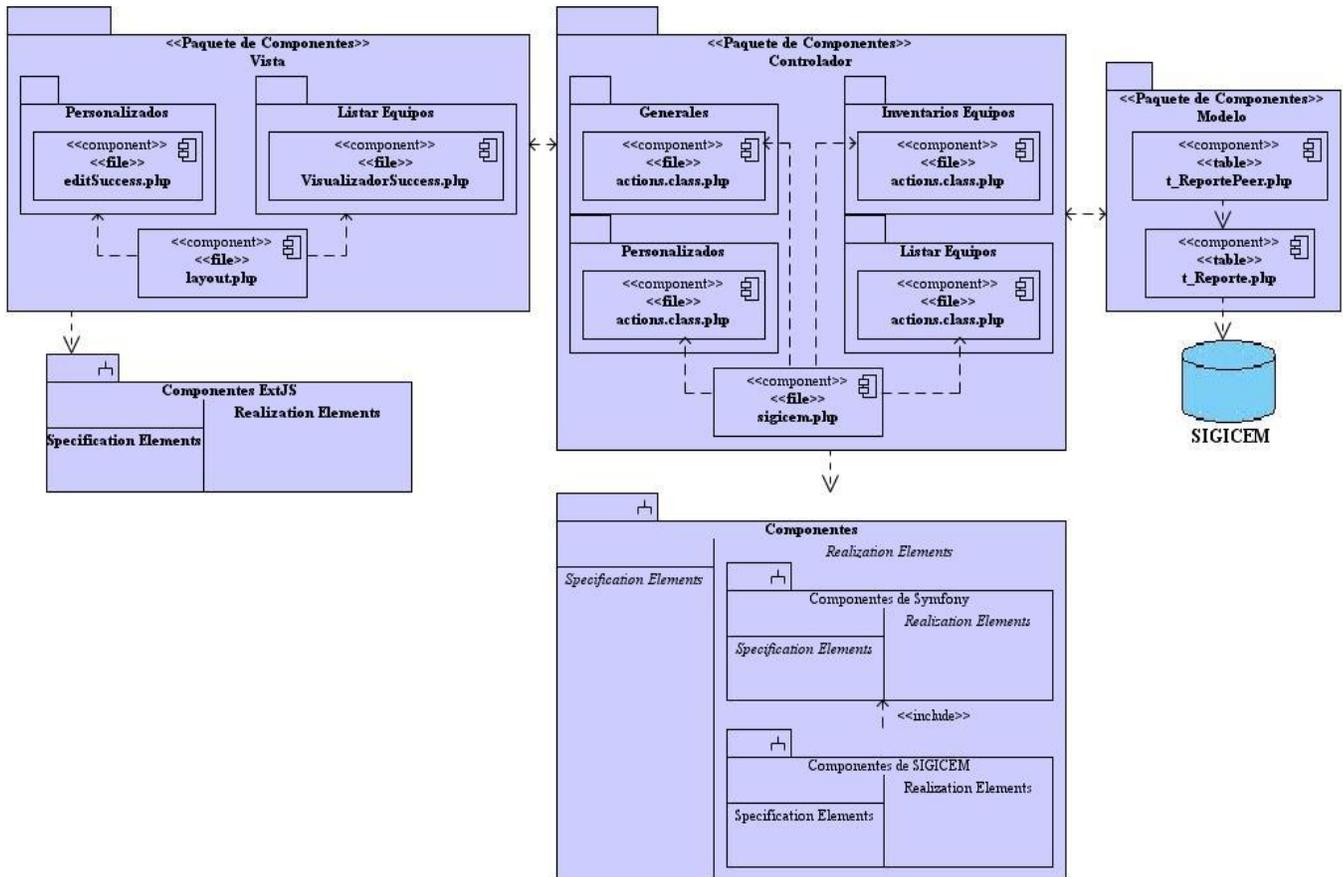


Figura # 27: Diagrama de componentes Módulo de Reportes Estadísticos.

¹⁹ Ídem 10.

4.3 Tratamiento de excepciones

Con el fin de obtener mayor robustez y fiabilidad se lleva a cabo durante el ciclo de vida del software el tratamiento de errores necesario; es decir, se investigan e identifican soluciones que den respuesta durante el tiempo de ejecución de un sistema a diferentes rutinas que pueden fracasar. Con la utilización del tratamiento de excepciones se hace posible restaurar la aplicación informática a un estado en el que la rutina pueda seguir su ejecución.

Realizar este proceso permite evitar que estas situaciones indeseadas se conviertan en problemas mayores que afecten la funcionalidad de la aplicación. Con la validación de los datos enviados por los usuarios al servidor se evitan muchos de estos errores; la validación de formularios se puede realizar en el lado del servidor y en el lado del cliente. En el servidor es obligatoria para no corromper la base de datos con datos incorrectos y en el lado del cliente es opcional, pero mejora la experiencia de usuario.

En el caso del presente trabajo se ha realizado el tratamiento de errores tanto en el cliente como en el servidor. Se puede señalar por ejemplo que se cuenta con cuadros de opción y menú de selección que facilitan la entrada de datos por parte del usuario al sistema.

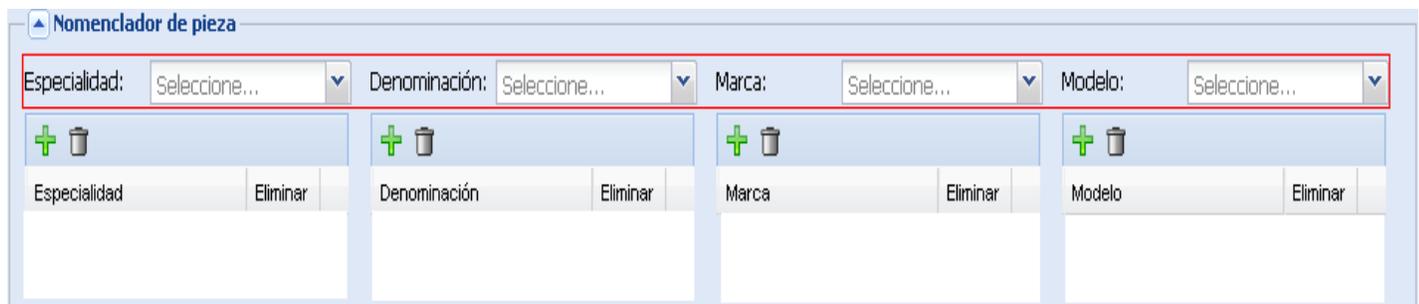


Figura # 28: Ejemplo de cuadros de selección utilizados.



Figura # 29: Ejemplo de menú de selección utilizado.

Intentar predecir los posibles usos incorrectos del programa y hacer uso del tratamiento de excepciones para corregirlos, es esencial para el correcto funcionamiento del software.

4.4 Estándares y estilos de codificación

Los estándares y estilos de codificación permiten que el código fuente de la solución que se esté en desarrollo posea mayor calidad, y que realizar el mantenimiento de esta se torne menos complejo.

Contar con un estilo de programación único permite que todos los involucrados puedan entender en un menor tiempo y que sea más fácil de ofrecer mantenimiento al código. Luego de analizar lo anteriormente planteado, y de efectuar un estudio de los estilos y estándares definidos por el CESIM y el departamento de SAS (Ver Documento_de_Arquitectura_de_Software_v1.3.2.pdf) se han identificado aquellos que serán utilizados para la implementación del módulo de Reportes Estadísticos propuesto.

Se tiene como idioma a utilizar el español, además que se debe tener en cuenta que las palabras no se acentuarán; ni se utilizará la letra “ñ” para evitar interpretaciones erróneas por parte de las PC, pues se encuentra en dependencia de las configuraciones que posea la misma. Además de estos aspectos se han asimilado estándares y estilos propuestos por las organizaciones antes mencionadas.

Identación		
Inicio y fin de bloque		Se recomienda dejar dos espacios en blanco desde la instrucción anterior para el inicio y fin de bloque <code>{}</code> . Lo mismo sucede para el caso de las instrucciones <code>if</code> , <code>else</code> , <code>for</code> , <code>while</code> , <code>do while</code> , <code>switch</code> , <code>foreach</code> .
Aspectos Generales	El indentado debe ser de dos espacios por bloque de código. No se debe usar el tabulador; porque este puede variar según la PC o la configuración de dicha tecla. Los inicios (<code>{</code>) y cierre (<code>}</code>) de ámbito deber estar alineados debajo de la declaración a la que pertenecen y deben evitarse si hay sólo una instrucción. Nunca colocar <code>{en</code> la línea de un código cualquiera, esto requiere una línea propia.	
Comentarios, separadores, líneas, espacios en blanco y márgenes.		
Ubicación de comentarios	Al inicio de cada clase o función y al final de cada bloque de código.	Se recomienda comentar al inicio de la clase o función especificando el objetivo de la misma.
Líneas en blanco	Se emplean antes y después de métodos, clases y estructuras.	Se recomienda dejar una línea en blanco antes y después de la declaración de una clase o de una estructura y de la implementación de una función.
Espacios en blanco	Entre operadores	Se recomienda usar espacios en blanco entre estos

	lógicos y aritméticos.	operadores para una mayor legibilidad en el código. Ejemplo: producto = nomproducto
Aspectos generales	Sobre el comentario	Se debe evitar comentar cada línea de código. Cuando el comentario se aplica a un grupo de instrucciones debe estar seguido de una línea en blanco. En caso de que se necesite comentar una sola instrucción se suprime la línea en blanco o se escribe a continuación de la instrucción.
	Sobre los espacios en blanco	No se debe usar espacio en blanco: Después del corchete abierto y antes del cerrado de un arreglo. Después del paréntesis abierto y antes del cerrado. Antes de un punto y coma.
Variables y constantes		
Aspectos generales	El nombre empleado, debe permitir que con sólo leerlo se conozca el propósito de la misma.	
Clases y Objeto		
Apariencia de clases y objetos	Primera letra en mayúscula.	Los nombres de las clases deben comenzar con la primera letra en mayúscula y el resto en minúscula, en caso de que sea un nombre compuesto se empleará notación PascalCasing. Ejemplo: MiClase ().
Apariencia de atributos	Primera letra en minúscula	El nombre que se le da a los atributos de las clases debe comenzar con la primera letra en minúscula, en caso de que sea un nombre compuesto se empleará notación CamellCasing.
Apariencia de las funciones	Primera letra en mayúscula	Para nombrar las funciones se debe tratar de utilizar verbos que denoten la acción que hace la función. Se empleará notación PascalCasing. Ejemplo: function BuscarUnidad (). Si son funciones que obtienen un dato se emplea el prefijo get y si fijan algún valor se emplea el prefijo set.
Aspectos generales	El nombre empleado para las clases, objetos, atributos y funciones debe permitir que con solo leerlo se conozca el propósito de los mismos.	

Tabla # 10: Estándares y estilos de codificación.²⁰

4.5 Modelo de pruebas

Verificar y determinar la calidad del software que se desarrolla constituye un eslabón de suma importancia durante el ciclo de construcción del mismo. RUP, dentro de sus disciplinas o flujos de trabajo propone la

²⁰ Fernández, Karel. *Documento de arquitectura de software*. CESIM, 2010. [Consultado: 05/03/2011]

realización de pruebas, que son utilizadas para determinar aquellos elementos candidatos a posibles errores de implementación, calidad, o usabilidad. *“Las pruebas son una actividad en la cual un sistema o componente es ejecutado bajo unas condiciones o requerimientos especificados, los resultados son observados y registrados, y una evaluación es hecha de algún aspecto del sistema o componente”*.²¹

Este flujo se centra en la detección y descripción de los defectos de la calidad del software, además de verificar que las funciones del producto se encuentren según lo diseñado y que los requisitos sean implementados correctamente. El proceso inicia desde los primeros momentos del desarrollo del software hasta la finalización del mismo. Se verifica el correcto funcionamiento del producto, basándose para ello en las especificaciones y peticiones hechas por los clientes y futuros usuarios.

El Módulo de Reportes Estadístico, obtenido a través de la ejecución de la presente investigación, en unión a los módulos de Nomencladores, Equipos Médicos, Captación de Datos, Recepción y Gestión Tecnológica, fue sometido a pruebas de funcionales. Estas fueron desarrolladas por parte de especialistas del Centro de Calidad para Soluciones Integrales (Calisoft), centro de certificación de software, reconocido y acreditado a nivel nacional.

Es común referirse a este tipo de pruebas también como pruebas de comportamiento o pruebas de caja negra. Estas se basan en la ejecución, revisión y retroalimentación de las funcionalidades previamente diseñadas para el software. Para el desarrollo de estas pruebas fueron diseñados los casos de pruebas correspondientes, que no son más que la representación de los datos que serán utilizados para ejecutar el software que se desea probar. Estableciéndose entradas, condiciones de ejecución y resultados esperados para la obtención de un objetivo específico.

Se diseñaron los casos de pruebas correspondientes a los casos de uso: Generar reporte personalizado de piezas, Generar reporte personalizado de órdenes de servicio, Generar reporte personalizado de mantenimientos, Generar reporte personalizado de especialistas, Generar reporte personalizado de equipos médicos y Listar equipos médicos. En las tablas que se muestran a continuación se podrán apreciar los escenarios que fueron definidos para la realización de las pruebas, no limita esto la posibilidad

²¹ *Ingeniería de Software 2 Curso: 2010-2011. Sobre la disciplinas de pruebas. [Consultado: 23/02/2011] [En línea]*

http://eva.uci.cu/file.php/259/Curso_2010-2011/Semana_9/Conferencia_7/Materiales_Basicos/Sobre_la_disciplina_de_Prueba.pdf

de que el especialista pueda incorporar nuevos escenarios durante el desarrollo de los previamente identificados.

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad	Flujo central
SC 1: Generar reporte personalizado (GRP) de piezas.	EC1.1: GRP de piezas con todos los campos vacíos.	El usuario accede a la interfaz correspondiente a reporte personalizado de piezas y presiona el botón Generar Reportes sin introducir ningún dato.	Inicio / Estadísticas y reportes / Reportes personalizados / Reporte personalizado de piezas
	EC1.2: GRP de piezas con la selección solamente los campos a visualizar.	El usuario accede a la interfaz correspondiente a reporte personalizado de piezas, selecciona la información a mostrar y presiona el botón Generar Reportes.	Inicio / Estadísticas y reportes / Reportes personalizados / Reporte personalizado de piezas
	EC1.3: GRP de piezas con la selección de criterios de búsqueda pero sin identificar los campos a visualizar.	El usuario accede a la interfaz correspondiente a reporte personalizado de piezas, selecciona criterios de búsqueda pero no la información a mostrar y presiona el botón Generar Reportes.	Inicio / Estadísticas y reportes / Reportes personalizados / Reporte personalizado de piezas
	EC1.4: GRP de piezas con la selección de los criterios de búsqueda y los campos a visualizar.	El usuario accede a la interfaz correspondiente a reporte personalizado de piezas, selecciona criterios de búsqueda, la información a mostrar y presiona el botón Generar Reportes.	Inicio / Estadísticas y reportes / Reportes personalizados / Reporte personalizado de piezas

Tabla # 11: Sección a probar: CUS-Generar reporte personalizado de piezas.

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad	Flujo central
SC 1: Generar reporte personalizado (GRP) de órdenes de	EC1.1: GRP de órdenes de servicio con todos los campos vacíos.	El usuario accede a la interfaz correspondiente a reporte personalizado de órdenes de servicio y presiona el botón Generar Reportes sin introducir ningún dato.	Inicio / Estadísticas y reportes / Reportes personalizados / Reporte personalizado de orden de servicio

servicio.	EC1.2: GRP de órdenes de servicio con la selección solamente los campos a visualizar.	El usuario accede a la interfaz correspondiente a reporte personalizado de órdenes de servicio, selecciona la información a mostrar y presiona el botón Generar Reportes.	Inicio / Estadísticas y reportes / Reportes personalizados / Reporte personalizado de orden de servicio
	EC1.3: GRP de órdenes de servicio con la selección de criterios de búsqueda pero sin identificar los campos a visualizar.	El usuario accede a la interfaz correspondiente a reporte personalizado de órdenes de servicio, selecciona criterios de búsqueda pero no la información a mostrar y presiona el botón Generar Reportes.	Inicio / Estadísticas y reportes / Reportes personalizados / Reporte personalizado de orden de servicio
	EC1.4: GRP de órdenes de servicio con la selección de los criterios de búsqueda y los campos a visualizar.	El usuario accede a la interfaz correspondiente a reporte personalizado de órdenes de servicio, selecciona criterios de búsqueda, la información a mostrar y presiona el botón Generar Reportes.	Inicio / Estadísticas y reportes / Reportes personalizados / Reporte personalizado de orden de servicio

Tabla # 12: Sección a probar: CUS-Generar reporte personalizado de órdenes de servicio.

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad	Flujo central
SC 1: Generar reporte personalizado (GRP) de mantenimiento.	EC1.1: GRP de mantenimiento con todos los campos vacíos.	El usuario accede a la interfaz correspondiente a reporte personalizado de mantenimiento y presiona el botón Generar Reportes sin introducir ningún dato.	Inicio / Estadísticas y reportes / Reportes personalizados / Reporte personalizado de mantenimiento
	EC1.2: GRP de mantenimiento con la selección solamente los campos a visualizar.	El usuario accede a la interfaz correspondiente a reporte personalizado de mantenimiento, selecciona la información a mostrar y presiona el botón Generar Reportes.	Inicio / Estadísticas y reportes / Reportes personalizados / Reporte personalizado de mantenimiento
	EC1.3: GRP de mantenimiento con la selección de criterios de búsqueda pero sin identificar los campos a visualizar.	El usuario accede a la interfaz correspondiente a reporte personalizado de mantenimiento, selecciona criterios de búsqueda pero no la información a mostrar y presiona el botón Generar Reportes.	Inicio / Estadísticas y reportes / Reportes personalizados / Reporte personalizado de mantenimiento

	EC1.4: GRP de mantenimiento con la selección de los criterios de búsqueda y los campos a visualizar.	El usuario accede a la interfaz correspondiente a reporte personalizado de mantenimiento, selecciona criterios de búsqueda, la información a mostrar y presiona el botón Generar Reportes.	Inicio / Estadísticas y reportes / Reportes personalizados / Reporte personalizado de mantenimiento
--	--	--	---

Tabla #13: Sección a probar: CUS-Generar reporte personalizado de mantenimiento.

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad	Flujo central
SC 1: Generar reporte personalizado (GRP) de especialistas.	EC1.1: GRP de especialistas con todos los campos vacíos.	El usuario accede a la interfaz correspondiente a reporte personalizado de especialistas y presiona el botón Generar Reportes sin introducir ningún dato.	Inicio / Estadísticas y reportes / Reportes personalizados / Reporte personalizado de especialistas
	EC1.2: GRP de especialistas con la selección solamente los campos a visualizar.	El usuario accede a la interfaz correspondiente a reporte personalizado de especialistas, selecciona la información a mostrar y presiona el botón Generar Reportes.	Inicio / Estadísticas y reportes / Reportes personalizados / Reporte personalizado de especialistas
	EC1.3: GRP de especialistas con la selección de criterios de búsqueda pero sin identificar los campos a visualizar.	El usuario accede a la interfaz correspondiente a reporte personalizado de especialistas, selecciona criterios de búsqueda pero no la información a mostrar y presiona el botón Generar Reportes.	Inicio / Estadísticas y reportes / Reportes personalizados / Reporte personalizado de especialistas
	EC1.4: GRP de especialistas con la selección de los criterios de búsqueda y los campos a visualizar.	El usuario accede a la interfaz correspondiente a reporte personalizado de especialistas, selecciona criterios de búsqueda, la información a mostrar y presiona el botón Generar Reportes.	Inicio / Estadísticas y reportes / Reportes personalizados / Reporte personalizado de especialistas

Tabla # 14: Sección a probar: CUS-Generar reporte personalizado de especialistas.

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad	Flujo central
SC 1: Generar reporte personalizado (GRP)	EC1.1: GRP de equipos médicos con todos los campos vacíos.	El usuario accede a la interfaz correspondiente a reporte personalizado de equipos y presiona el	Inicio / Estadísticas y reportes / Reportes personalizados /

de equipos médicos.		botón Generar Reportes sin introducir ningún dato.	Reporte personalizado de equipos médicos
	EC1.2: GRP de equipos médicos con la selección solamente los campos a visualizar.	El usuario accede a la interfaz correspondiente a reporte personalizado de equipos, selecciona la información a mostrar y presiona el botón Generar Reportes.	Inicio / Estadísticas y reportes / Reportes personalizados / Reporte personalizado de equipos médicos
	EC1.3: GRP de equipos médicos con la selección de criterios de búsqueda pero sin identificar los campos a visualizar.	El usuario accede a la interfaz correspondiente a reporte personalizado de equipos, selecciona criterios de búsqueda pero no la información a mostrar y presiona el botón Generar Reportes.	Inicio / Estadísticas y reportes / Reportes personalizados / Reporte personalizado de equipos médicos
	EC1.4: GRP de equipos médicos con la selección los criterios de búsqueda y los campos a visualizar.	El usuario accede a la interfaz correspondiente a reporte personalizado de equipos, selecciona criterios de búsqueda, la información a mostrar y presiona el botón Generar Reportes.	Inicio / Estadísticas y reportes / Reportes personalizados / Reporte personalizado de equipos médicos

Tabla # 15: Sección a probar: CUS-Generar reporte personalizado de equipos médicos.

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad	Flujo central
SC1: Listar equipos médicos.	EC1.1: Listar equipos médicos satisfactoriamente.	El sistema debe Listar equipos médicos satisfactoriamente.	Inicio / Estadísticas y reportes / Visualizador de equipos médicos
	EC1.2: No existen equipos médicos a listar.	El sistema debe devolver que no existen equipos médicos a mostrar.	Inicio / Estadísticas y reportes / Visualizador de equipos médicos

Tabla # 16: Sección a probar: CUS-Listar equipos médicos.

Los casos de pruebas, fueron puestos a disposición de los especialistas de Calisoft, que ejecutaron cada uno en busca de todas las posibles fallas que pudiesen ser detectadas. Luego del proceso de ejecución de estas, se obtuvo el Acta de Liberación de Productos de Software (Anexo # 1), emitida por este centro. Actualmente se alistan los preparativos necesarios para emprender las Pruebas de Aceptación de los módulos anteriormente mencionados. Estas se hacen antes del despliegue del sistema, para constatar

que el software cumple con las necesidades reales establecidas en la etapa inicial del ciclo de vida del software y por tanto puede ser usado por usuarios finales.

Durante la confección del capítulo que recién concluye se abordaron cuestiones pertenecientes a la etapa de implementación. Obteniéndose el modelo de implementación, con los diagramas de componentes correspondientes, se detallaron además las organizaciones y dependencias lógicas existentes entre dichos componentes. Fueron identificados los estándares y estilos de codificación a utilizar para la implementación de los componentes definidos y las opciones a efectuar para realizar el tratamiento de errores. Se diseñaron los casos de pruebas correspondientes y fueron puestos a disposición de los especialistas de Calisoft, obteniéndose el Acta de Liberación de Productos de Software (Anexos # 1) emitido por dicho centro.

CONCLUSIONES GENERALES

- ✦ La realización de un estudio previo de sistemas utilizados para la generación de reportes estadísticos a nivel nacional e internacional, resaltó la necesidad de llevar a cabo el desarrollo del Módulo Reportes Estadísticos del Sistema de Gestión para la Ingeniería Clínica y Electromedicina.
- ✦ El análisis de los procesos asociados a la gestión de reportes estadísticos en el Centro de Ingeniería Clínica y Electromedicina, permitió obtener una guía para el posterior desarrollo de la solución propuesta, mediante la aplicación de las herramientas y tecnologías definidas por el Centro de Informática Médica y el Departamento Sistemas de Apoyo a la Salud.
- ✦ Mediante la confección de los artefactos pertenecientes a las Disciplinas de Trabajo definidas y guiados por el Proceso Unificado de Desarrollo, se logró establecer la documentación necesaria de la investigación.
- ✦ Se implementaron los componentes del Módulo de Reportes Estadísticos integrándose a los existentes en el Sistema de Gestión para la Ingeniería Clínica y Electromedicina.
- ✦ Se realizaron las pruebas de calidad a la solución propuesta, obteniéndose el Acta de Liberación de Productos de Software por parte del Centro de Calidad para Soluciones Tecnológicas (Calisoft).

RECOMENDACIONES

- × Hacer uso del Módulo de Reportes Estadísticos en las distintas Organizaciones Básicas de Ingeniería Clínica y Electromedicina, para su utilización y validación por parte de los usuarios finales.
- × Analizar una posible integración del Módulo de Reportes Estadísticos con el Generador Dinámico de Reportes versión 2.0, actualmente en desarrollo.

REFERENCIAS BIBLIOGRÁFICAS

1. INFOMED. Portal de Salud de Cuba. **Aspectos Generales del Sistema Nacional de Salud**. [Consultado: 15/10/2010.] [En línea]: http://www.sld.cu/sistema_de_salud/aspectos.html.
2. Ídem 1.
3. Sitio oficial del MIC. **Tecnología de la información**. [Consultado: 15/10/2010.] [En línea:] <http://www.mic.gov.cu/HThemEmp.aspx?1>
4. Hernández, Rolando Alfredo y Coello, Zayda. **El paradigma cuantitativo de la investigación científica**. Editorial Universitaria, 2002.
5. Definición de. **Concepto de información**. [Consultado: 2/11/2010.] [En línea:] <http://definicion.de/informacion>
6. Definición de. **Definición de reporte**. [Consultado: 2/11/2010.] [En línea:] <http://definicion.de/reporte>
7. Moreno Tamayo, Yoanki. **Análisis y diseño del Módulo Generador de Reportes del proyecto ONE**. Universidad de las Ciencias Informáticas, 2008. [Consultado: 08/03/2011.]
8. Sánchez Hernández, Yanet, Matos Arias, José Manuel. **Reportes Estadísticos del Registro de Partos y Nacimientos**. Universidad de las Ciencias Informáticas, 2009. [Consultado: 08/03/2011.]
9. Ídem 7.
10. Jacobson, Ivar, Booch, Grady y Rumbaugh, James. **El Proceso Unificado de Desarrollo de Software**. España: Adison-Wesley.
11. Sitio oficial. **NetBeans ya incluye soporte para Symfony**. [Consultado: 17/11/2010.] [En línea] <http://www.symfony.es/2009/10/05/netbeans-ya-incluye-soporte-para-symfony/>
12. Potencier, Fabien y Zaninotto, François. **Symfony 1.2, la guía definitiva**. 2008.
13. Ídem 12.
14. Miranda Levis, Carlos. **LAMP: La Plataforma Más Popular para Software Códigos Libres**. [Consultado: 06/03/2011.] [En línea:] <http://www.socinfo.com/codigolibre/lamp>
15. Ídem 4.
16. Ingeniería de Software 2 (2010-2011). **Disciplina de Análisis y Diseño**. [Consultado: 03/ 02/2011] [En línea] http://eva.uci.cu/file.php/259/Curso_2009-2010/Conferencia_1/Materiales_basicos/Conferencia_1.doc
17. Rational Unified Process. **Extended Help for RUP**. 2003. [Consultado: 03/ 02/2011]

18. Machado Díaz, Maidelys, Coca Ribas, Yader Luis. **SACCEM: Módulo de gestión de la información del Departamento de Supervisión del CCEEM versión 1.0.** Universidad de las Ciencias Informáticas, 2009. [Consultado: 09/03/2011.]
19. Ídem 10.
20. Fernández, Karel. **Documento de arquitectura de software.** CESIM, 2010. [Consultado: 05/03/2011]
21. Ingeniería de Software 2 Curso: 2010-2011. **Sobre la disciplinas de pruebas.** [Consultado: 23/02/2011] [En línea] http://eva.uci.cu/file.php/259/Curso_2010-2011/Semana_9/Conferencia_7/Materiales_Basicos/Sobre_la_disciplina_de_Prueba.pdf

BIBLIOGRAFÍA

1. Arias, Manuel. **Carmen. Estándares de codificación**. [Consultado: 11/01/2011.]
[En línea] <http://www.cisiad.uned.es/carmen/estilo-codificacion.pdf>
2. Besembel, Isabel y Chacón, Edgar. **Objetos y reglas de negocios en la integración y automatización de procesos de producción continua**.
3. Brito, Karenny. **“Selección de metodologías de desarrollo para aplicaciones web en la Facultad de Informática de la Universidad de Cienfuegos”**. Cienfuegos, Cuba, 2009.
[Consultado: 28/ 01/2011]
[En línea] http://www.eumed.net/libros/2009c/584/RUP_Modelo_del_sistema.htm
4. Camallea, Noel Luis y Coutin Abalo, Ronald. **Diccionario de Informática**. Editorial Científico-Técnica, 2005.
5. de la Osa, José A. **La semilla del desarrollo de la salud pública en Cuba**. [Consultado: 20/10/2010.] [En línea] <http://www.granma.cubaweb.cu/marti-moncada/pro07.html>
6. Definición ABC, una guía única en la red. **Definición de reporte**. [Consultado: 2/11/2010.]
[En línea] <http://www.definicionabc.com/comunicacion/reporte.php>
7. Definición de. **Concepto de información**. [Consultado: 2/11/2010.]
[En línea:] <http://definicion.de/informacion>
8. Definición de. **Definición de reporte**. [Consultado: 2/11/2010.]
[En línea:] <http://definicion.de/reporte>
9. Delgado, Gregorio. **Etapas del desarrollo de la salud pública revolucionaria cubana**.
[Consultado: 18/10/2010.] [En línea] http://bvs.sld.cu/revistas/spu/vol22_1_96/spu10196.htm
10. Diccionario de términos económicos y financieros. [Consultado: 11/10/2010.]
[En línea] http://portal.lacaixa.es/docs/diccionario/E_es.html#ESTADISTICA
11. **El Patrón MVC (Modelo Vista Controlador)**. [Consultado: 15/12/2010.]
[En línea] <http://prestashop5estrellas.wordpress.com/2010/03/29/el-patron-mvc-modelo-vista-controlador/>
12. Ext Management Team. Ext JS- A foundation you can build on. **Introducing ExtJS 2.0**.
[Consultado: 10/11/2010.] [En línea] <http://www.extjs.com/>
13. fabFORCE.net. **The DBDesigner 4 Successor**. [Consultado: 10/11/2010.]

- [En línea] <http://www.fabforce.net/dbdesigner4/>.
14. Fernández, Karel. **Documento de arquitectura de software**. CESIM, 2010. [Consultado: 05/03/2011.]
 15. Gisbert, Jaime. **La importancia de las estadísticas**. [Consultado: 11/10/2010.]
[En línea] <http://www.MundodelMarketing.com>
 16. Hernández Sampieri, Roberto, Fernández Collado, Carlos y Baptista Lucio, Pilar. **Metodología de la investigación científica**. Segunda edición, 1998.
 17. Hernández, Rolando Alfredo y Coello, Zayda. **El paradigma cuantitativo de la investigación científica**. Editorial Universitaria, 2002.
 18. Hidalgo, Mauricio. **Reporting Services**. [Consultado: 4/11/2010.]
[En línea] http://www.sqlmax.com/reportin_services1.asp
 19. INFOMED. **Aspectos Generales del Sistema Nacional de Salud**. Infomed. Portal de Salud de Cuba. [Consultado: 15/10/2010.] [En línea]: http://www.sld.cu/sistema_de_salud/aspectos.html.
 20. Ingeniería de Software 1 Curso: 2010-2011. **Introducción a la Disciplina de Requisitos de RUP**. [Consultado: 24/ 01/2011] [En línea] http://eva.uci.cu/file.php/102/Curso_2010-2011/Clases/Semana_06/Conf_7/Materiales_complementarios/Introduccion_a_la_Disciplina_de_Requisitos.pdf
 21. Ingeniería de Software 2 Curso: 2010-2011. **Disciplina de Análisis y Diseño**. [Consultado: 03/ 02/2011] [En línea] http://eva.uci.cu/file.php/259/Curso_2009-2010/Conferencia_1/Materiales_basicos/Conferencia_1.doc
 22. Ingeniería de Software 2 Curso: 2010-2011. **Documentación sobre pruebas**. [Consultado: 23/ 02/2011] [En línea] http://eva.uci.cu/file.php/259/Curso_2010-2011/Semana_9/Conferencia_7/Materiales_Basicos/Documentacion_sobre_Pruebas.pdf
 23. Ingeniería de Software 2 Curso: 2010-2011. **Evaluación de software**. [Consultado: 23/ 02/2011] [En línea] http://eva.uci.cu/file.php/259/Curso_2010-2011/Semana_9/Conferencia_7/Materiales_Complementarios/Evaluacion_de_Software.pdf
 24. Ingeniería de Software 2 Curso: 2010-2011. **Sobre la disciplinas de pruebas**. [Consultado: 23/ 02/2011] [En línea] http://eva.uci.cu/file.php/259/Curso_2010-2011/Semana_9/Conferencia_7/Materiales_Basicos/Sobre_la_disciplina_de_Prueba.pdf
 25. Intrepidlbex. **Ubuntu**. [Consultado: 10/11/2010.] [En línea] <https://wiki.ubuntu.com/Intrepidlbex>

26. Jacobson, Ivar, Booch, Grady y Rumbaugh, James. **El Proceso Unificado de Desarrollo de Software**. España : Adison-Wesley
27. Machado Díaz, Maidelys, Coca Ribas, Yader Luis. **SACCEM: Módulo de gestión de la información del Departamento de Supervisión del CCEEM versión 1.0.** Universidad de las Ciencias Informáticas, 2009. [Consultado: 09/03/2011.]
28. Mestras, Juan Pavón. **Estructura de las Aplicaciones Orientadas a Objetos. El patrón Modelo-Vista-Controlador**. [Consultado: 15/12/2010.]
[En línea] <http://www.fdi.ucm.es/profesor/jpavon/poo/2.14.MVC.pdf>
29. Miranda Levis, Carlos. **LAMP: La Plataforma Más Popular para Software Códigos Libres**. [Consultado: 06/03/2011.] [En línea:] <http://www.socinfo.com/codigolibre/lamp>
30. Moreno Tamayo, Yoanki. **Análisis y diseño del Módulo Generador de Reportes del proyecto ONE**. Universidad de las Ciencias Informáticas, 2008. [Consultado: 08/03/2011.]
31. Moreno, Carlos Miguel. **Proyecto moReport v1.0**. [Consultado: 4/11/2010.]
[En línea] <http://bo.unsa.edu.ar/docacad/morep/>
32. Potencier, Fabien y Zaninotto, François. **Symfony 1.2, la guía definitiva**. 2008.
33. Pressman, Roger. **Ingeniería del Software: Un enfoque práctico**. Quinta Edición. Madrid, 2002.
34. Rational Unified Process. **Extended Help for RUP**. 2003.
35. **Rational Unified Process**. [Consultado: 20/11/2010.]
[En línea] <http://www.rational.com.ar/herramientas/rup.html>
36. **Razones por las cuales usar Netbeans para proyectos PHP**. [Consultado: 17/11/2010.] [En línea] <http://playa-soluciones.cl/blog/7-investigacion-y-desarrollo/31-razones-por-las-cuales-usar-netbeans-para-proyectos-php>
37. **Reutilización del software. Patrones de diseño**. [Consultado: 13/12/2010.]
[En línea] <http://siul02.si.ehu.es/alfredo/iso/06Patrones.pdf>
38. **RUP - Rational Unified Process**. [Consultado: 20/11/2010.]
[En línea] <http://www.aesist.com/metodologias/rup---rational-unified-process>
39. Saavedra, Esteban. **Symfony Framework para el desarrollo de aplicaciones web**. [Consultado: 20/11/2010.] [En línea] <http://www.softwarelibre.org.bo/esteban/files/86/203/symfony.pdf>
40. Sánchez Hernández, Yanet, Matos Arias, José Manuel. **Reportes Estadísticos del Registro de Partos y Nacimientos**. Universidad de las Ciencias Informáticas, 2009. [Consultado: 08/03/2011.]

41. Sitio oficial. **NetBeans ya incluye soporte para Symfony.** [Consultado: 17/11/2010.]
[En línea] <http://www.symfony.es/2009/10/05/netbeans-ya-incluye-soporte-para-symfony/>
42. Sitio oficial del MIC. **Tecnología de la información.** [Consultado: 15/10/2010.]
[En línea:] <http://www.mic.gov.cu/HThemEmp.aspx?1>
43. Tedeschi, Nicolás. **¿Qué es un Patrón de Diseño?** [Consultado: 13/12/2010.]
[En línea] <http://msdn.microsoft.com/es-es/library/bb972240.aspx#XSLTsection123121120120>
44. The PHP Group. **News Archive - 2007. PHP.** [Consultado: 11/11/2010.]
[En línea] <http://www.php.net/archive/2007.php>.
45. Torossi, Gustavo. **El proceso unificado de software.**
46. Videla, Álvaro. **Symfony Design Patterns.** [Consultado: 13/12/2010.]
[En línea] <http://obvioushints.blogspot.com/2008/12/symfony-design-patterns.html>

ANEXOS



Acta de Liberación de Productos Software

Fecha de liberación: 25 de marzo de 2011.

Emitida a favor de: Electromedicina.

1. Datos del producto

Artefacto	Versión	Estado final	Cantidad Iteraciones	Tipos de pruebas realizadas
Módulo Nomencladores	1.0	0 NC	3	Pruebas Funcionales
Módulo Captación de Datos	1.0	0 NC	2	Pruebas Funcionales
Módulo Equipos Médicos	1.0	0 NC	2	Pruebas Funcionales
Módulo Recepción	1.0	0 NC	2	Pruebas Funcionales
Módulo Gestión Tecnológica	1.0	0 NC	3	Pruebas Funcionales
Módulo de Reportes Estadísticos	1.0	0 NC	2	Pruebas Funcionales
SIGICEM (Integración de los Módulos)	1.0	0 NC	2	Pruebas Funcionales

Yoanis Costilla Camejo

Nombre y Apellidos

Responsable Calisoft

Yoanis
Costilla
Camejo

Firmado digitalmente por Yoanis Costilla Camejo
Nombre de reconocimiento (DN):
0.9.2342.19200300.100.1.1=ycostilla, cn=Yoanis Costilla Camejo, serialNumber=15727, givenName=Yoanis, sn=Costilla Camejo, c=CU
Fecha: 2011.03.28 10:21:06 -04'00'

Dennys Javier Hernández

Nombre y Apellidos

Responsable Proyecto

Tayché
Capote
García

Firmado digitalmente por Tayché Capote García
Nombre de reconocimiento (DN):
0.9.2342.19200300.100.1.1=tcapoto, cn=Tayché Capote García, serialNumber=14320, givenName=Tayché, sn=Capote García, c=CU
Fecha: 2011.03.28 11:22:25 -04'00'

Anexo # 1: Acta de liberación de productos de software.

GLOSARIO DE TÉRMINOS

CESIM: Centro de Informática Médica.

CICEM: Centro de Ingeniería Clínica y Electromedicina.

CNE: Centro Nacional de Electromedicina.

CPE: Centro Provincial de Electromedicina.

CUN: caso de uso del negocio.

CUS: caso de uso del sistema.

Estadísticas: ciencia que trata la recolección, clasificación y presentación de los hechos sujetos a una apreciación numérica como base a la explicación, descripción y comparación de los fenómenos.

Framework: Se conoce como marco de trabajo y constituye un conjunto de conceptos, metodologías y herramientas de administración y diseño para el desarrollo de forma estandarizada de una aplicación.

GADE: Grupo de Automatización y Desarrollo de Electromedicina.

Instituciones de salud: Establecimientos del Sistema Nacional de Salud destinados a brindar atención médica y servicios de salud a la población.

MIC: Ministerio de la Informática y las Comunicaciones.

MINSAP: Ministerio de Salud Pública.

MVC: Modelo Vista Controlador, patrón de arquitectura de software.

Notación PascalCasing: Los identificadores y nombres de variables, métodos y funciones están compuestos por múltiples palabras juntas, cada palabra con letra mayúscula.

OBICEM: Organización Básica de Ingeniería Clínica y Electromedicina.

PHP: (Hypertext PreProcessor) Lenguaje de programación para el desarrollo de web dinámicas. Originalmente se conocía como Personal Home Page tools, herramientas para páginas personales (en Internet).

Reportes: informes que organizan y exhiben la información contenida en una base de datos, cuya función es aplicar un formato determinado a los datos para mostrarlos.

RF: Requerimientos Funcionales.

RNF: Requerimientos No Funcionales.

RUP: Proceso Unificado de Desarrollo, del inglés, Rational Unified Process; proceso de desarrollo de software, es decir, conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema de software.

SAS: Sistemas de Apoyo a la Salud.

SIGEM: Sistema de Gestión para la Electromedicina.

SIGICEM: Sistema de Gestión para la Ingeniería Clínica y Electromedicina.

SNE: Sistema Nacional de Electromedicina.

SNS: Sistema Nacional de Salud.

Symfony: framework desarrollado en PHP 5 y está enfocado al desarrollo de aplicaciones web en el mismo lenguaje de programación.

TIC: Tecnologías de la Informática y las Comunicaciones.

UCI: Universidad de Ciencias Informáticas.

UML: Lenguaje Unificado de Modelado, del inglés, Unified Modeling Language.