



Universidad de las Ciencias Informáticas

Facultad 7

Trabajo de diploma para optar por el Título de Ingeniero en Ciencias
Informáticas

**Sistema para el levantamiento de inventario de equipos
médicos por los especialistas del Centro de Ingeniería
Clínica y Electromedicina.**

Autores: Zuleidy Rodríguez Gómez

Hanssel Mirabal Cantillo

Tutores: Ing. Dennys Javier Hernández Peña

Ing. Yuneisi Castillo Tamayo

La Habana, junio 2013

“Año 55 de la Revolución”

DATOS DE CONTACTO

Ing. Dennys Hernández Peña.

Correo Electrónico: dhernandez@uci.cu

Ingeniero Informático graduado del curso 2004-2005. CUJAE. Profesor Asistente. Líder del grupo de desarrollo para la gestión de equipos médicos (GDEM) durante 4 años. Ha impartido las asignaturas de Seguridad Informática, Sistema Operativo e Ingeniería de Software. Ha sido tribunal de tesis durante 7 años y tutor de más de 20 trabajos de tesis de grado. Actualmente arquitecto del proyecto SIGICEM del Dpto. Sistemas Especializados del Centro de Informática Médica y profesor de la Facultad 7.

Ing. Yuneisi Castillo Tamayo.

Correo electrónico: yctamayo@uci.cu

Ingeniera en Ciencias Informáticas, graduada en el curso 2008-2009, en la Universidad de Ciencias Informáticas. Especialista del departamento de Sistemas Especializados del Centro de Informática Médica (CESIM). Cotutora de 1 tesis de grado en el curso 2011-2012 en la Facultad 7 de la Universidad de Ciencias Informáticas (UCI).

Agradecimientos

AGRADECIMIENTOS

A la fuerza que me impulsó a seguir el camino: mi mamá.

Al apoyo y consentimiento de mi papá.

Al ejemplo de mi hermana Dreyemi y alegría de mis sobrinos en mis momentos más tristes.

A los consejos inolvidables de mi tía Luisa.

Al amor y cariño del amor de mi vida: Reysel.

A mis viejos amigos Rey, Darianna, Daneily y Yanet, que después de tanto tiempo sigue formando parte de mi vida.

A los momentos vividos con mis amigos(as): Yeni, Clau, Arle, Yasnay, Glory, Odelys, Abel, el flaco en fin el grupo en general, que de una forma u otra hicieron que mis deseos por llegar a donde estoy hoy fueran más grandes durante estos 5 años.

A Yisel por su gran ayuda.

Al apoyo y ayuda de mi compañero de tesis, sin el yo no estaría cumpliendo mi sueño hoy.

A mis tutores Denis y Yuneisi, y a todo el equipo de trabajo que nos guió realizar este trabajo

Zuleidy

Agradecimientos

Antes que todo mi mayor agradecimiento es para la persona que creo que sin su ayuda esto no hubiera sido posible: Nelson, que siempre estuvo ahí para lo que fuera necesario. También quiero agradecerles a mis padres, ya que todo este esfuerzo más que nada es por ellos.

A Yisel porque desde que la conocí me ha ayudado incondicionalmente y ha estado siempre para lo que me haga falta.

A todos los de mi grupo por soportarme todos estos años que se que no es fácil pero en especial a mi compañera de tesis que desde primer año está cargando conmigo en trabajos, seminarios y finalmente en la tesis.

A Leo, Rey, La Nane, Alexis, David, Andres, Daymer, Gloria, Carlos, Jose Joel, en fin a todos.

A la Dagma y Maryis por soportarme y aguantarme.

A mis tutores, a Angelica, a Idayana, a Jorgito y a todos los que de una forma u otra contribuyeron en esta tesis.

Y en especial a mi hermano que esto le sirva de ejemplo.

Hanssel

DEDICATORIA

Dedico este Trabajo de Diploma de manera especial a nuestro querido Comandante en Jefe Fidel Castro Ruz, por hacer realidad el gran sueño que es la Universidad de las Ciencias Informáticas y por brindarnos a los jóvenes su luz y su guía.

RESUMEN

El Centro de Ingeniería Clínica y Electromedicina (CICEM), órgano rector y metodológico para la política de la tecnología médica en nuestro país, posee un sistema llamado Sistema de Gestión para Ingeniería Clínica y Electromedicina (SIGICEM) que permite gestionar toda la información electromédica existente en las unidades de salud. Dentro de sus funcionalidades, cuenta con un módulo que facilita el levantamiento de inventario de los equipos médicos del Sistema Nacional de Salud (SNS).

Actualmente por problemas de conectividad en algunas de las unidades de salud del país, el trabajo de los especialistas que laboran en el levantamiento de inventario de equipos médicos se ve afectado. Es por ello que el presente trabajo se propone desarrollar una aplicación para la gestión de los inventarios de tecnología médica desde dichas instituciones de salud. Para el desarrollo del sistema fue necesario el estudio, profundización y utilización de metodologías, tecnologías y herramientas entre las que se encuentran fundamentalmente: el Proceso Unificado de Desarrollo (RUP), el Lenguaje Unificado de Modelado (UML), el lenguaje de programación Java, la librería de interfaz gráfica Swing, el mecanismo de serialización de java, la herramienta de modelado Visual Paradigm, el Entorno Integrado de Desarrollo Netbeans y el sistema operativo Ubuntu.

El desarrollo de este sistema va a permitir realizar el inventario de equipos médicos en las unidades de salud que no posean una infraestructura de red que permita conectarse a SIGICEM, garantizando con esto el correcto funcionamiento del proceso de levantamiento de inventario y una integridad de la información relacionada con el equipamiento médico, ya que el sistema brindará la posibilidad de seleccionar los datos de entrada.

Palabras clave: conectividad, equipo médico, inventario.

ÍNDICE

INTRODUCCIÓN	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA	5
1.1. Sistemas automatizados existentes vinculados a la gestión de inventario.....	5
1.2. Metodología de Desarrollo de Software.....	10
1.3. Herramientas y tecnologías.....	11
CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA	16
2.1. Descripción del problema	16
2.2. Propuesta del sistema	16
2.3. Modelo del negocio	17
2.4. Especificación de los requisitos del software.....	20
2.5. Definición de los casos de uso del sistema	23
CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA	28
3.1. Descripción de la arquitectura. Fundamentación.....	28
3.2. Análisis de posibles implementaciones o componentes existentes que puedan ser rehusados. 30	
3.3. Análisis.....	30
3.4. Diseño.....	36
CAPÍTULO 4: IMPLEMENTACIÓN	45
4.1. Modelo de Implementación.....	45
4.2. Tratamiento de errores	47
4.3. Seguridad e Integridad de la información.....	48
4.4. Estrategias de codificación. Estándares y estilos a utilizar.....	49
4.5. Solución Implementada	50
CONCLUSIONES	55
RECOMENDACIÓN.....	56
REFERENCIAS BIBLIOGRÁFICAS.....	57
BIBLIOGRAFÍA.....	60
GLOSARIO DE TÉRMINOS.....	64

INTRODUCCIÓN

Las Tecnologías de la Información y las Comunicaciones (TIC) constituyen uno de los recursos más importantes de la sociedad, trayendo como consecuencia una explosión exponencial en la transmisión e intercambio de datos, información y conocimientos, a los cuales se puede acceder sin tener en cuenta barreras geográficas o limitaciones de tiempo. Es por ello que en los últimos años la mayoría de los países han establecido e implementado proyectos, políticas y estrategias para promover el uso de las TIC y aprovechar los beneficios y los aportes que estas ofrecen.

En Cuba se han derivado programas sociales, impulsados por la Revolución, donde se han introducido paulatinamente el uso de las TIC, tal es el caso de la educación y la salud. En este último ha existido un incremento vertiginoso de la tecnología médica, surgiendo la necesidad de crear el Centro Nacional de Electromedicina (CNE), orientado a gestionar y controlar los equipos de todas las unidades de salud del país. Desde sus inicios el centro cuenta con un sistema automatizado que permite realizar las gestiones específicas de las actividades electromédicas e integrar las informaciones de carácter general del SNS, conocido como: Sistema Integral de Gestión para Electromedicina (SIGEM).

Posteriormente se decide vincular la ingeniería clínica y la electromedicina, dando lugar a la actualización de dicho centro quedando como CICEM, institución que se encarga actualmente de brindar servicios técnicos y gestionar la tecnología médica existente en el SNS. Esto trajo consigo la necesidad de crear un nuevo sistema que respondiera a las necesidades y políticas del nuevo centro, por lo que en la actualidad se encuentra en desarrollo el SIGICEM, por parte de un equipo de trabajo del Departamento Sistemas Especializados en Salud (SES), el cual conjuntamente a otros departamentos integran el Centro de Informática Médica (CESIM), subordinando a la facultad 7 de la Universidad de las Ciencias Informáticas (UCI).

El nuevo sistema permite cubrir todo el proceso de gestión tecnológica llevado a cabo por los especialistas del CICEM, contribuyendo a la fiabilidad y actualización de la información referente al equipamiento médico instalado en el país, y a elevar la calidad y eficiencia de los servicios técnicos dentro del SNS mediante la organización, planificación, control y análisis de las acciones que se ejecuten.

La versión actual de SIGICEM obliga a los especialistas a trabajar en línea con el sistema, pero al no contar con una infraestructura de red desde cada una de las unidades de salud del país, la gestión de la información que se debe recoger preliminarmente sobre el estado, ubicación y los datos técnicos de los equipos médicos se hace engorrosa, teniendo que efectuar el proceso de forma manual, trayendo consigo:

- La duplicación de esfuerzo cada vez que los especialistas realicen la impresión de modelos para soporte de esta actividad, tomen los detalles en papel de los equipos médicos y posteriormente reescriban la información dentro del sistema al conectarse al mismo.
- Un alto costo económico en función de la compra de materiales de oficina y piezas innecesarias para equipos médicos.
- Retrasos e introducción de errores en esta etapa tan importante del proceso de gestión tecnológica, la cual garantiza la continuidad y correcta ejecución de los demás procesos relacionados con el equipamiento médico.

De acuerdo a la situación descrita anteriormente surge como **problema a resolver**: ¿Cómo gestionar el inventario de los equipos médicos desde las unidades de salud cuando no exista conexión con el SIGICEM?, dando lugar al **objeto de estudio**: el proceso de gestión de los inventarios de la tecnología médica, el cual se enmarca en el **campo de acción**: la gestión de inventarios de la tecnología médica en el Centro de Ingeniería Clínica y Electromedicina. Por lo que se propone como **objetivo general**: Desarrollar una aplicación para la gestión de los inventarios de equipos médicos desde las instituciones de salud.

Para dar cumplimiento al objetivo planteado para esta investigación se han definido las siguientes **tareas de investigación**:

1. Realización de un estudio de los procesos para la gestión de equipos implementados dentro del Sistema de Gestión para Ingeniería Clínica y Electromedicina.
2. Asimilar las herramientas, tecnologías y arquitectura definidas por el proyecto para el desarrollo de soluciones de escritorio.

3. Diseño de los artefactos del flujo de trabajo Modelamiento del Negocio, Requisitos, Análisis y Diseño e implementación del sistema para el levantamiento de inventario de equipos médicos por los especialistas del Centro de Ingeniería Clínica y Electromedicina.
4. Selección del sistema de soporte de datos a utilizar y diseñar la estructura de soporte de los datos para la gestión de la información utilizada.
5. Implementación de las funcionalidades especificadas para el sistema.

Una vez desarrollada la aplicación se espera obtener los siguientes **resultados**:

1. Los entregables establecidos y los artefactos generados a partir de las Políticas establecidas para el desarrollo de software en la universidad.
2. Versión funcional estable del Sistema para el levantamiento de inventario de equipos médicos para los especialistas del Centro de Ingeniería Clínica y Electromedicina.
3. Optimización de tiempo y de recursos al realizar el levantamiento de inventarios de equipos médicos para la puesta en marcha de la gestión tecnológica.
4. Integridad de la información referente al equipamiento médico existente en las unidades de salud del país.

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

Capítulo 1: Fundamentación teórica. Se describen los conceptos fundamentales asociados al dominio del problema, exponiéndose el estado del arte del tema a abordar, a nivel internacional y nacional. Se realiza además un análisis crítico entre las soluciones ya existentes y la propuesta de solución, para posteriormente seleccionar las metodologías, tecnologías y herramientas en las que se apoya la solución al problema.

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 el proceso de inventario. 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 arquitectónicos utilizados.

Capítulo 4: Implementación. Se expone una descripción del desarrollo de la solución inicialmente propuesta, mediante la representación del Modelo de Implementación; así como, se elabora Modelo de Despliegue que será utilizado para la puesta en marcha del sistema. Se presentan algunos de los criterios que se tuvieron en cuenta para llevar a cabo el tratamiento de errores en el sistema. Se describen los estándares y estilos de codificación a utilizar.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

El proceso de gestión de los inventarios del equipamiento médico en cada una de las instituciones de salud es de vital importancia para la racionalidad de los recursos, ya que este permite llevar un estricto control de los equipos médicos existentes en las instituciones de salud. Debido a esto se han desarrollado varias soluciones informáticas encaminadas a automatizar los procesos relacionados con la gestión de inventarios, para un mejor control del mismo y la correcta toma de decisiones, las cuales están soportadas sobre diferentes tecnologías y herramientas. Estos sistemas serán capaces de almacenar y centralizar grandes volúmenes de información, obtener resultados de forma rápida, encontrar información específica fácilmente, procesar los datos, ahorrar tiempo y dinero.

1.1. Sistemas automatizados existentes vinculados a la gestión de inventario.

1.1.1. Sistema precedente

En la presente investigación fue considerado como sistema precedente el SIGICEM, el cual rige todos los procesos que se llevan a cabo por los especialistas del Sistema Nacional de Electromedicina (SNE). Entre los once módulos que posee, se encuentra el Módulo de Gestión de Inventario de los equipos médicos gestionados por dicha entidad en las unidades de salud del país. Debido a su naturaleza, el sistema necesita la presencia de una infraestructura de red para su funcionamiento, surgiendo la necesidad de realizar un estudio de sistemas a nivel nacional e internacional que permitan la gestión de inventarios y que posean similitud con algunas funcionalidades de dicho módulo, garantizando la integración de ambos sistemas.

1.1.2. Sistemas Internacionales

Moyín 1.4: es un pequeño y sencillo programa portable para realizar el control de un inventario. Con Moyín puedes registrar, modificar y eliminar los artículos, así como también imprimir reportes ordenados por código, descripción y cantidad. El programa permite realizar búsquedas por código o descripción y también filtrar la información para ver sólo los artículos que cumplan algunos requisitos de cantidad,

descripción o código. Es posible realizar copias de seguridad de una manera muy sencilla y restaurar la información guardada fácilmente¹. Está soportado solamente para la plataforma Windows.

OK Inventario 2.1.5: es una aplicación diseñada para proveer soluciones de contabilidad, control de almacén e inventario, así como la gestión empresarial, de ventas y de documentación para pequeñas y medianas empresas comerciales y de fabricación. Se subdivide en los módulos:

- Gestión de ventas y suministros.
- Ventas en línea.
- Finanzas.
- Gestión de almacenes e inventario.
- Informes.
- Empleados y Gestión de usuarios.

Desde dichos módulos se podrá, de forma intuitiva, llevar el control de las mercancías, servicios, productos, facturas, reservas, ventas, pedidos y precios, efectuar la catalogación, generar informes y listados, publicar inventarios en la web, realizar la gestión de pagos/costes, de la información de cuentas bancarias, del registro de la caja, de las transferencias de fondos. Además permite llevar el control sobre los suministros, los beneficios, los representantes de los productos, entre otros². Este sistema solo puede ser utilizado en el sistema operativo Windows.

Sistema Administrativo para el Taller (SAT): es un sistema que gracias a su diseño estructural es fácilmente adaptable para llevar a cabo la administración de cualquier tipo de taller. Es una aplicación con fines comerciales por lo que para hacer uso de la misma es necesario pagar la licencia. Los sistemas operativos soportados son Windows® ME, XP, 2000, 2003, Windows® Vista, Windows® 7 y Windows® 8³. Dentro de las principales características que posee el SAT se encuentran:

- Gestión integral para el taller.

¹ ABCdatos. [En línea] 17 de 07 de 2009. [Citado el: 25 de 11 de 2012.] <http://www.abcdatos.com/programa/inventarios-gestion.html>.

² Programas-gratis.net. [En línea] Mediaprogramas , 2013. [Citado el: 2013 de enero de 28.] <http://ok-inventario.programas-gratis.net/>.

³ satnetwork Sistemas Informaticos. [En línea] 2011. [Citado el: 6 de 12 de 2012.] <http://www.satnetwork.com.ar>.

- Búsqueda y modificación de las órdenes de trabajo.
- Control de inventario.
- Campos personalizables.
- Copia de seguridad automática.
- Rastreo de artículos mediante código de barra.
- Editor de reportes de impresión.
- Gráfico de cantidad de reparaciones por cliente.
- Gráfico de tiempo de la reparación.

1.1.3. Sistemas Nacionales

Rodas XXI: sistema creado por CITMATEL para la automatización de la gestión empresarial. El módulo de Inventario de Rodas XXI le permite tener un control detallado de los inventarios de su entidad, realizando en el mismo momento que se registra un movimiento, su contabilización. Se pueden realizar todo tipo de operaciones de entradas y salidas de los almacenes con facilidad en el momento que se desee, generando el documento asociado al movimiento. Es posible trabajar con varios almacenes trabajando cada uno de forma independiente⁴. Este sistema está soportado solamente por la plataforma Windows.

Versat Sarasola: es un paquete integrado para la gestión económica financiera que permite enviar información eficaz, de forma inmediata, desde lugares apartados, a la vez que ofrece mayor organización, control y disciplina en cada gestión. Se distingue por ser el primer sistema de contabilidad cubano certificado, según las nuevas normativas establecidas tanto por el Ministerio de Finanzas y Precios, como por el de la Informática y las Comunicaciones, para este tipo de software. Este sistema cuenta con un conjunto de 10 módulos entre los que se encuentra: Control de activos fijos, que recoge las operaciones normales que en esta actividad se realizan (altas, bajas y modificaciones de los medios). También se le ha incorporado otras opciones novedosas, dentro de las cuales se encuentran la posibilidad de amortizar los

⁴Rodas XXI Sistema Integral Económico Administrativo. [En línea] 2002. <http://www.rodasxxi.cu/inventario.php>.

Capítulo 1: Fundamentación Teórica

activos por los 4 métodos más conocidos internacionalmente. Posee un asistente (*wizard*) para la configuración del subsistema, la tarea más compleja de este módulo⁵.

Características:

- Es una aplicación de escritorio.
- Implementado en Delphi.
- Trabaja solamente sobre el sistema operativo Windows.
- Soporte para base de datos SQL Server 2000.

SISCONT5: sistema cubano creado por la empresa Tecnomática en el año 2007, el cual se aviene a las definiciones y conceptos del Ministerio de la Industria Básica (MINBAS), aunque por las acciones contables financieras que permite puede ser utilizado en otras entidades nacionales. Está formado por varios módulos⁶:

- Efectivo en Caja y Bancos.
- Inventarios.
- Cobros y Pagos.
- Facturación.
- Activos Fijos Tangibles.
- Nóminas.
- Contabilidad de Costos

CONDOR: sistema automatizado de alta complejidad y seguridad que abarca todos los aspectos del proceso contable de una entidad, tales como la dualidad de moneda y el pago por resultados. Entre sus módulos se encuentra el de Gestión de Inventarios, el cual permite entre otras funcionalidades:

- Control de los esquemas de depósitos y almacenes.
- Realizar inventarios rotativos.

⁵ DISAIC Casa Consultora. [En línea] 2008. [Citado el: 4 de 12 de 2012.]
<http://www.disaic.cu/modules.php?name=content&pa=showpage&pid=818>.

⁶ SISCONT. (s.a). Tu software contable financiero. [En línea] [Citado el: 14 de 11 de 2012.]
<http://www.siscont.com/DESCRIPTIVO%20SISCONT.pdf>.

Capítulo 1: Fundamentación Teórica

- Serializar los productos.
- Generar transacciones de costeo.

Posee una interfaz gráfica, brinda rapidez, comodidad y seguridad para el cliente. Actualmente este software se encuentra implementado solamente para el Sistema Operativo Windows en cualquiera de sus versiones, lo que imposibilita su utilización como solución al problema planteado.

SISCOMIP: es un Sistema Contable Integral diseñado para las empresas del Ministerio de la Industria Pesquera. Su arquitectura no es modular. Está concebido para llevar a cabo el registro contable de la actividad económica, a nivel de documento primario, en una entidad cualquiera compuesta por unidades, de forma compartimentada o no, y está en explotación actualmente en entidades pertenecientes al Ministerio de la Industria Pesquera⁷. Está formado por varios subsistemas:

- General.
- Inventarios.
- Finanzas.
- Comercial.
- Nóminas.
- Activos Fijos.

Los sistemas anteriormente expuestos no constituyen una solución viable para darle respuesta a la presente investigación, porque además de las características que se muestran en la tabla # 1, no son configurables en cuanto a los componentes que presentan, no son lo suficientemente adaptables para asimilar el proceso de levantamiento de inventario del CICEM y no poseen la capacidad de integración con el SIGICEM.

⁷ Miniet, Y. V. *Definición del ciclo de vida del proyecto ERP*. Habana: s.n., 2009.

Sistemas	Características		
	Multiplataforma	Dependencia de un Sistema Gestor de Base de Datos	Integración con SIGICEM
Moyín	No	No	No
OK Inventario	No	No	No
Sat	No	Si	No
Rodas	No	Si	No
Versat Sarasola	No	Si	No
SISCONT5	Si	Si	No
CONDOR	No	Si	No
SISCOMIP	No	Si	No

Tabla 1: Comparación de los sistemas

1.2. Metodología de Desarrollo de Software

Durante el desarrollo de software, muchas tareas y actividades se complejizan, provocando que el proceso de desarrollo de software sea riesgoso y difícil de controlar, obteniendo con esto clientes insatisfechos con el resultado final. La forma para solucionar o tratar de llevar a cabo un eficiente desarrollo es aplicando una metodología de desarrollo de software, la cual es un conjunto de pasos y procedimientos que deben seguirse para desarrollar software, indicando quién debe hacer cada actividad, cuándo hacerla y qué debe hacer.

Proceso Unificado de Desarrollo (RUP)

El Proceso Unificado de Desarrollo (*RUP*, por sus siglas en inglés) es un proceso de desarrollo de software, es decir, un “conjunto de actividades necesarias para transformar los requisitos del usuario en un sistema software.”⁸ RUP fue seleccionada por ser uno de los procesos más generales de los existentes actualmente, ya que está pensado para adaptarse a cualquier proyecto y para conseguir sus objetivos se basa fundamentalmente en el orden y la documentación. Esta metodología se caracteriza por ser dirigido

⁸Jacobson, Ivar, Booch, Grady y Rumbaugh, James. *El proceso Unificado de desarrollo de Software*. Madrid: Addison Weasley, 2000.

por casos de uso, centrado en la arquitectura y ser iterativo e incremental, donde cada fase se desarrolla en iteraciones, de forma tal que se pueda dividir en pequeños proyectos mejorando su comprensión y desarrollo.

En el proceso de desarrollo de software el modelado ejerce un papel fundamental. RUP utiliza como lenguaje de modelado el Lenguaje Unificado de Modelado (UML) que con la utilización de diagramas y gráficos brindan una mejor perspectiva de lo que se quiere.

1.3. Herramientas y tecnologías.

Durante la presente investigación fueron definidas diferentes herramientas y tecnologías que serán utilizadas para apoyar el proceso de desarrollo del Sistema para el levantamiento de inventario de equipos médicos por los especialistas de CICEM, guiado por una solución de escritorio desarrollada en el año 2012 para el manejo de órdenes de servicio del SIGICEM. Para la implementación de esta pequeña aplicación se utilizó el lenguaje de programación Java, la librería Swing, el mecanismo de serialización de Java y Netbeans 6.9 sobre el sistema operativo Ubuntu 10.04. Con el objetivo de seguir el mismo patrón en el proyecto para el desarrollo de soluciones de escritorio se considera importante mantener estas herramientas y tecnologías en la nueva propuesta.

Lenguaje Unificado de Modelado (UML 2.1)

El Lenguaje Unificado de Modelado (UML, por sus siglas en inglés), es un lenguaje de modelado visual que permite visualizar, especificar, construir y documentar los artefactos que se generan en el proceso de modelado y construcción de un software. Las funciones principales de UML son las siguientes:

Visualizar: UML permite expresar de una forma gráfica un sistema de forma que otra persona lo pueda entender.

Especificar: UML permite especificar cuáles son las características de un sistema antes de su construcción.

Construir: A partir de los modelos especificados se pueden construir los sistemas diseñados.

Documentar: Los propios elementos gráficos sirven como documentación del sistema desarrollado que pueden servir para su futura revisión⁹.

Básicamente, UML permite a los desarrolladores visualizar los resultados de su trabajo en esquemas o diagramas estandarizados.

Visual Paradigm for UML 6.4

Las herramientas Ingeniería de Software Asistida por Ordenador (CASE, por sus siglas en inglés), son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el coste de las mismas, en términos de tiempo y de dinero.

Visual Paradigm es una herramienta CASE para el modelado UML que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. Esta herramienta es independiente de la plataforma y dota de una buena cantidad de productos o módulos para facilitar el trabajo durante la confección de un software, así como garantizar la calidad del producto final. Además se integra fácilmente con varios Entornos de Desarrollo Integrado (IDE, por sus siglas en inglés).

Lenguaje de Programación Java

A diferencia de los lenguajes convencionales, Java es compilado a un código intermedio (bytecode), el cual es interpretado por una máquina virtual de Java. La máquina virtual hace posible que una aplicación que haya sido implementada en Java se ejecute en cualquier sistema operativo con soporte para la máquina virtual. También proporciona un entorno de ejecución que convierte el código neutro de Java al código nativo del ambiente en que está siendo ejecutada. Entre sus características se encuentran:

Portabilidad: Permite desarrollar con un ambiente y desplegar en otro.

Seguridad: Permite que cualquier error por parte de la aplicación no pueda comprometer la seguridad ni la estabilidad del sistema en su totalidad. Todos los errores son registrados de alguna manera por las herramientas utilizadas, de forma que la información puede ser utilizada para corregir el error.

⁹ James Rumbaugh, Ivar Jacobson y Grady Booch. *El Lenguaje Unificado de Modelado. Manual de Referencia.* Madrid : Addison Weasley, 2000.

Manejo de memoria: Java es un lenguaje administrado, lo que significa que la memoria utilizada por las aplicaciones no es manejada por ellas sino por la máquina virtual subyacente.

Java es un lenguaje de programación multipropósito que tiene todas las características de un ambiente orientado a objetos: es sencillo, robusto, portable, seguro, dinámico y de alto rendimiento¹⁰. Es distribuido bajo la licencia (GPL) lo que lo hace gratis y sin restricciones en su uso, existe una gran cantidad de documentación sobre él, es fácil de usar y es multiplataforma.

Java Platform, Standard Edition (Java SE 6)

Es una plataforma de programación para desarrollar y ejecutar software de aplicaciones en lenguaje de programación Java. Los dos productos principales de la plataforma Java SE son: *Java Development Kit* (JDK) y *Java SE Runtime Environment* (JRE). El JDK es un súper conjunto de la JRE y contiene todo lo que está en el JRE, además de herramientas tales como los compiladores y depuradores necesarios para el desarrollo de aplicaciones en Java. El JRE proporciona las bibliotecas, la máquina virtual de Java, y otros componentes para ejecutar aplicaciones escritas en Java¹¹.

Esta plataforma ha sido seleccionada porque permite el desarrollar aplicaciones de escritorio que cumplan con las expectativas de la universidad basadas en las políticas de utilización de software libre que se están aplicando a lo largo de todo el país.

Mecanismo de serialización de Java

La serialización es un proceso por el que un objeto cualquiera se puede convertir en una secuencia de bytes con la que más tarde se podrá reconstruir dicho objeto manteniendo el valor de sus variables¹².

Dentro de las principales características de esta técnica se encuentran que:

- Permite guardar una gran cantidad de información en un fichero garantizando con ello la persistencia de los datos y la portabilidad de los mismos.

¹⁰ Enterate en línea. [En línea] 27 de 11 de 2008. [Citado el: 8 de 12 de 2012.]

<http://www.enterate.unam.mx/Articulos/2003/junio/j2ee.htm>.

¹¹ Oracle Corporation. Oracle Technology Network. [En línea] 2011. [Citado el: 8 de 12 de 2012.]

<http://download.oracle.com/javase/6/docs/>.

¹² Vélez Serrano, José F, y otros. *Diseñar y programar, todo es empezar: Una introducción a la Programación Orientada a Objetos usando UML y Java*. Madrid : DYKINSON, S.L. Méndez Valdés, 61 - 28015 Madrid, 2011. pág. 128. 978-84-9982-345-4.

- Brinda la posibilidad de que la información pueda ser leída y escrita en el fichero serializado sin importar el sistema operativo.

Se empleará el mecanismo de serialización que provee Java ya que la información que se gestiona en el sistema se almacenará en un fichero con el objetivo de lograr la portabilidad de la información. No se consideró necesario la creación de una base de datos porque requiere la instalación y configuración de un Sistema Gestor de Base de Datos (SGBD) en cada una de las estaciones donde se vaya a utilizar la aplicación, requiriendo esto de mayores prestaciones.

Swing

Conjunto de librerías que son parte de la *Java Foundation Classes* (JFC) que implementan una serie de componentes utilizados en la construcción de interfaces gráficas¹³. Entre sus principales características se encuentran: su arquitectura está basada en patrón Modelo Vista Controlador (MVC), lo que proporciona un alto grado de extensibilidad y de personalización de los componentes visuales de la librería. Permite establecer estilos de interfaz llamados *look and feels* que modifican la forma en que se muestra y se comporta toda la interfaz de usuario. Es más dependiente de la plataforma Java y no del sistema operativo, facilitándole con ello a los programadores mayor variedad de componentes gráficos a la hora de diseñar las interfaces. Se empleará Swing para diseñar la interfaz gráfica del sistema.

iText 2.1.7

Es una librería que permite crear y manipular documentos con extensión .pdf. Además se distribuye bajo los principios del software libre haciendo uso de la Licencia Pública de Mozilla (MPL, por sus siglas en inglés)¹⁴. Se empleará la librería iText para generar el reporte en formato pdf que brinda el sistema.

Netbeans 6.9

Como Entorno Integrado de Desarrollo (IDE, por sus siglas en inglés) fue utilizado el NetBeans, siendo este una herramienta para que los programadores puedan escribir, compilar, depurar y ejecutar programas. Está escrito en Java pero puede servir para cualquier otro lenguaje de programación. Existe

¹³García Peñalvo, Francisco José. zarza.usa. [En línea] 2008. [Citado el: 5 de 12 de 2012.] http://zarza.usal.es/~fgarcia/doc/tuto2/IV_3.htm.

¹⁴Lowagie, Bruno. *iText in Action*. New York : Manning Publications Co, 2007. 1932394796.

además un número importante de módulos para extenderlo. Además es un producto libre y gratuito sin restricciones de uso¹⁵. Este IDE permite flexibilizar la programación gráfica en java debido a los componentes visuales que tiene predefinido.

Sistema Operativo

Fue seleccionado Ubuntu en su versión 10.04 para el desarrollo de la aplicación por ser el sistema operativo definido en la arquitectura del proyecto SIGICEM y del departamento SES. Ubuntu es una distribución Linux que ofrece un sistema operativo enfocado a ordenadores de escritorio aunque también proporciona soporte para servidores. Basada en Debian GNU/Linux, concentra su objetivo en la libertad de uso y los lanzamientos regulares (cada 6 meses)¹⁶. Ubuntu está totalmente basado en los principios del Software Libre.

Luego de analizados los sistemas a nivel nacional e internacional que permiten la gestión de inventarios, se ratificó la necesidad de desarrollar un sistema que cumpla con las características del sistema precedente. Fueron definidas la metodología de desarrollo, tecnologías y herramientas, seleccionadas por el equipo de trabajo para este tipo de aplicaciones complementarias con el SIGICEM, que soportarán el sistema a desarrollar. Debido a la característica de portabilidad que posee el sistema, fue seleccionado para el almacenamiento de los datos que se gestionarán el fichero serializado, utilizando el mecanismo de serialización de Java.

¹⁵ Oracle Corporation. *netbeans.org*. [En línea] 2012 de 12 de 5. http://netbeans.org/index_es.html.

¹⁶ Canonical Ltd. *ubuntu-es*. [En línea] *ubuntu-es*, 4 de 6 de 2011. http://doc.ubuntu-es.org/Sobre_Ubuntu.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

La importancia del flujo de trabajo Modelamiento del negocio radica en que sin el panorama completo del alcance del negocio y sin el entendimiento de sus procesos no podrán identificarse las necesidades inmediatas de mejora y continuidad relativa a las actividades relacionadas con los sistemas informáticos, que son el producto final del desarrollo.

2.1. Descripción del problema

El CICEM cuenta con un sistema informático para gestionar la información referente a la tecnología médica existente en el país y los recursos asociados a estas nombrado SIGICEM, el cual posee un Módulo de Inventario que permite llevar el control de todos los equipos médicos existentes solo en algunas unidades de salud, ya que no todas poseen una infraestructura de red debido a que el país, a pesar del esfuerzo realizado, no ha logrado informatizar completamente el sector de la salud.

En estas unidades de salud que no tienen acceso a las facilidades que brinda dicho módulo, el proceso de inventario se realiza de forma manual por los especialistas del CICEM. Esto trae consigo la introducción de una gran cantidad de errores relacionados con los datos del equipo y que no exista una integridad de la información que se maneja, repercutiendo esto considerablemente en la calidad del proceso. Teniendo en cuenta estos problemas surge la necesidad de implementar un sistema de escritorio que permita llevar a cabo el inventario de equipos médicos en estas unidades de salud que no poseen conectividad.

2.2. Propuesta del sistema

Sobre la base del análisis realizado a los problemas existentes en el CICEM, el sistema a desarrollar permitirá al usuario autenticarse antes de efectuar el levantamiento de inventario de los equipos médicos de las unidades de salud que no poseen conectividad con la base de datos de SIGICEM. La información que se maneja será guardada en un fichero serializado usando el mecanismo de serialización de Java, el cual almacena la información encriptada para imposibilitar su lectura. Además se podrá generar reportes en formato .pdf resumiendo los datos de los inventarios, con el objetivo de dejar una constancia física del inventario realizado en la unidad.

Capítulo 2: Características del sistema

El sistema brindará la posibilidad de trabajar conectado a la base de datos de SIGICEM desde los centros de Electromedicina y algunas unidades de salud donde se pueda establecer la conexión, con el objetivo de descargar del servidor los datos necesarios para realizar el levantamiento de inventario de los equipos médicos y actualizar en el mismo los datos de dicho inventario, permitiendo a los usuarios:

- Autenticarse por primera vez, verificando que el usuario sea un especialista registrado en la base de datos y guardar los datos del mismo en el fichero, garantizando que el usuario pueda acceder al sistema cuando no exista conexión.
- Seleccionar una o varias unidades de salud.
- Descargar de la base de datos la información necesaria para el correcto funcionamiento del sistema y los equipos médicos pertenecientes a la(s) unidad(es) de salud seleccionada(s), los cuales se guardarán en el fichero serializado con estado viejo.
- Sincronizar en la base de datos las modificaciones efectuadas en el proceso de inventario, actualizando solo los equipos médicos guardados en el fichero que posean estado nuevo o modificado. Luego de la sincronización el usuario tendrá la opción de borrar o no los datos en la aplicación, en este último caso los equipos modificarán su estado a viejo, garantizándose que no se duplique la información.

2.3. Modelo del negocio

Comprender la estructura y la dinámica de la organización en la cual se va a implantar un sistema; así como sus problemas actuales e identificar las mejoras potenciales, derivar los requerimientos del sistema y asegurar que los consumidores, usuarios finales y desarrolladores tengan un entendimiento común de lo que se desea lograr, son los objetivos fundamentales del modelo del negocio. Para lograr esos propósitos, el proceso de modelado permite obtener una visión de la organización que permita definir sus procesos, roles y responsabilidades en los modelos de casos de uso del negocio y de objetos.

En el problema descrito anteriormente interviene un actor y un trabajador del negocio, los cuales desarrollan determinadas responsabilidades en función de un objetivo común: realizar el inventario de equipos médicos (Ver tabla # 1 y 2).

Capítulo 2: Características del sistema

Actor	Descripción
Jefe de Brigada	El Jefe de Brigada es el encargado de orientar la realización del inventario.

Tabla 2: Descripción del actor del negocio.

Trabajador	Descripción
Especialista	El especialista es el encargado de defectar y revisar el estado técnico de los equipos médicos pertenecientes al área de salud designada, además de darle de alta a los mismos en el inventario. También crea la ficha técnica con los datos iniciales de cada equipo, la cual puede o no contener una propuesta de mantenimiento anual.

Tabla 3: Descripción del trabajador del negocio.

Antes de dar inicio a las actividades necesarias para el levantamiento de inventario de equipos médicos es preciso tomar en consideración que el Jefe de Brigada es el máximo responsable de la realización del inventario (Ver figura # 1).

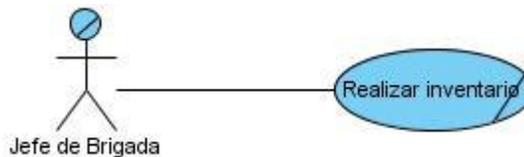


Figura #1: Diagrama de casos de uso del negocio.

El Jefe de brigada y los especialistas tendrán acceso únicamente a la información dentro del área a la que pertenecen. Además se debe verificar que los datos de los equipos sean correctos para no incurrir en errores posteriores.

La inicialización de las tareas se encuentra a cargo del Jefe de brigada, actor del negocio, ya que es el responsable de orientar al especialista la realización del inventario de los equipos médicos pertenecientes a un área de salud determinada. Este último, trabajador del negocio, debe primeramente llevar a cabo la defectación de cada equipo médico que se encuentre dentro del área especificada, con el objetivo de conocer todos los datos técnicos del mismo. En caso de que el equipo sea complejo, es decir, este formado por varios equipos, el especialista debe definir las partes que lo conforman y defectar cada una de dichas partes.

Capítulo 2: Características del sistema

Posteriormente debe definir el estado técnico actual de los equipos, teniendo en cuenta que el estado funcionando se le asigna a todo aquel que funcione en óptimas condiciones, el estado defectuoso al que no ofrezca la máxima calidad de sus servicios o no pueda realizar ciertas y determinadas tareas, el estado roto al que no funcione en su totalidad. Luego de estas acciones el especialista posee todos los datos necesarios para darle alta al equipo en el libro de inventario y para crear la ficha técnica con los datos iniciales del mismo, la cual puede o no contener un mantenimiento anual propuesto por el especialista, concluyendo de esta forma el flujo de actividades (Ver figura # 2).

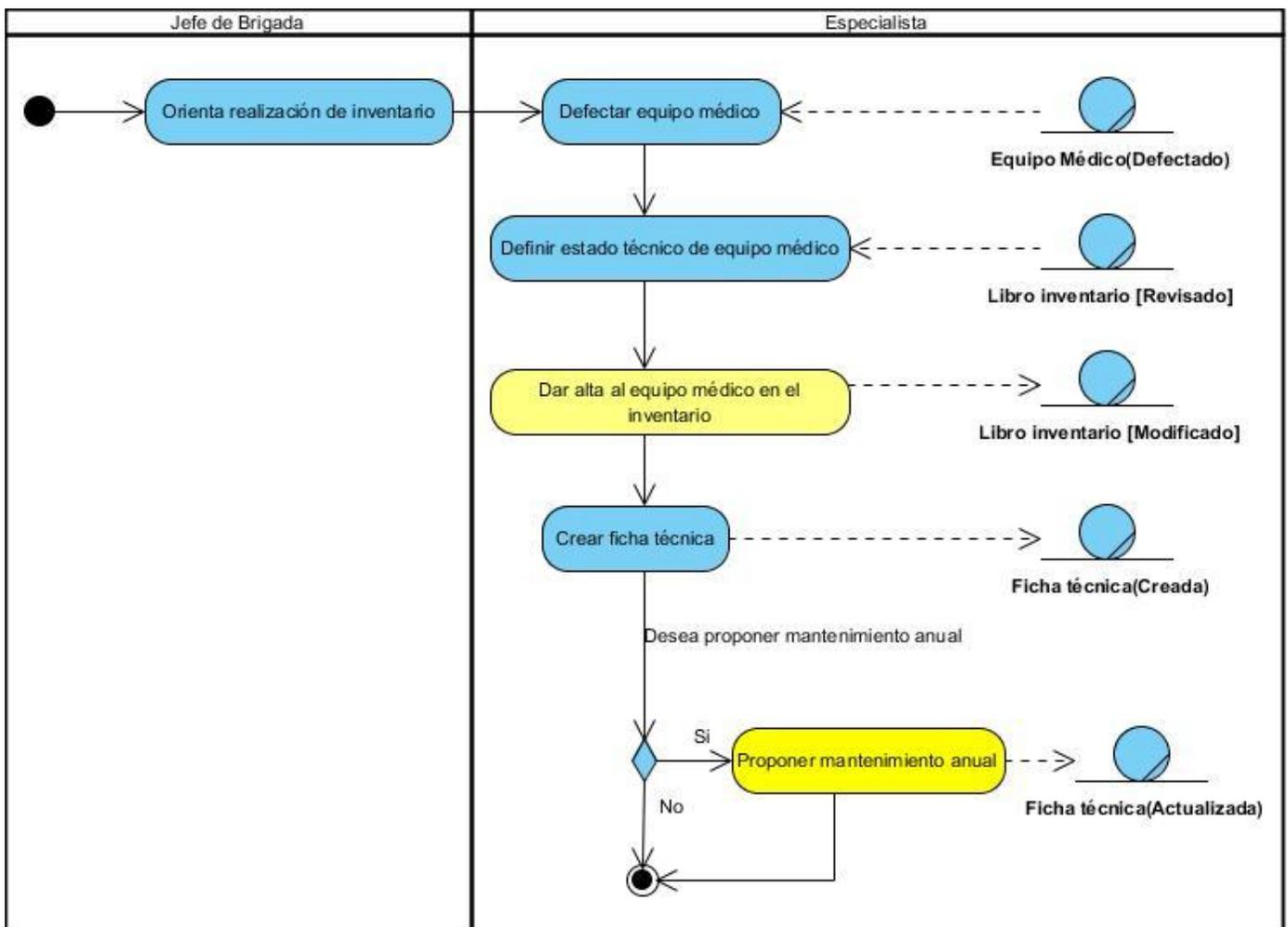


Figura # 2: Diagrama de actividades del caso de uso del negocio Realizar inventario.

Capítulo 2: Características del sistema

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 inventario, hace referencia al Equipo Médico (Ver figura # 3).

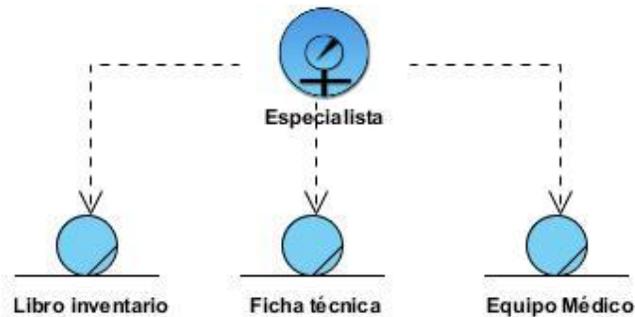


Figura # 3: Modelo de objetos.

Este término se encuentra representado de forma genérica, ya que estos pueden ser: equipos de rayos x, equipos de ultrasonido, complejo hematológico, equipo conteo de células, somatón, entre otros.

2.4. Especificación de los requisitos del software

En cualquier proyecto software los requisitos son las necesidades del producto que se debe desarrollar. Por ello, en la fase de análisis de requisitos se deben identificar claramente estas necesidades y documentarlas. Como resultado de esta fase se debe producir un documento de especificación de requisitos en el que se describa lo que el futuro sistema debe hacer.

Una buena especificación de requisitos software ofrece una serie de ventajas entre las que se destacan el contrato entre cliente y desarrolladores, la reducción del esfuerzo en el desarrollo, una buena base para la estimación de costes y planificación, un punto de referencia para procesos de verificación y validación, y una base para la identificación de posibles mejoras en los procesos analizados¹⁷. Los requisitos se pueden clasificar en funcionales y no funcionales.

¹⁷Agut, Raúl Monferrer. *Especificación de Requisitos Software según el estándar de IEEE 830*. Departament d'Informàtica, Universitat Jaume I. 2000-2001.

Requisitos Funcionales (RF)

Son capacidades o condiciones que el sistema debe cumplir. Estos describen los servicios que se espera que el sistema cumpla para satisfacer las necesidades del usuario y define el comportamiento del sistema. Se definen los siguientes requisitos después de haber hecho un análisis del negocio.

- RF1:** Insertar equipo médico.
- RF2:** Modificar equipo médico.
- RF3:** Eliminar equipo médico.
- RF4:** Buscar equipo médico
- RF5:** Listar equipos médicos.
- RF6:** Generar reporte de equipo médico.
- RF7:** Autenticar
- RF8:** Cargar usuario.
- RF9:** Configurar la conexión.
- RF10:** Actualizar datos en la aplicación.
- RF11:** Seleccionar unidad de salud.
- RF12:** Eliminar unidad de salud.
- RF13:** Actualizar datos en el servidor.
- RF14:** Cargar equipo médico.
- RF15:** Cargar unidad de salud.
- RF16:** Cargar nomenclador de ubicación.
- RF17:** Cargar programa de salud.
- RF18:** Cargar nomenclador de equipo médico.
- RF19:** Cargar especialista.
- RF 20:** Proponer mantenimiento anual.

Requisitos no Funcionales (RnF)

Son propiedades o cualidades que el producto debe tener. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable. Estos requerimientos forman una parte significativa de la especificación, ya que son importantes para que clientes y usuarios puedan

Capítulo 2: Características del sistema

valorar las características del producto y marcar la diferencia entre un producto bien aceptado y uno con poca aceptación. El sistema a desarrollar debe poseer las siguientes cualidades:

Usabilidad

RnF 1.1. Los usuarios del sistema deben tener un tiempo de entrenamiento no mayor de un mes para que sean productivos operando en el mismo.

RnF 1.2. La aplicación de escritorio deberá facilitar la interacción usuario–sistema con el objetivo de evitar rechazo en el uso de la misma, y guiará mediante mensajes al usuario en las diferentes acciones que realice.

RnF 1.3. La finalidad de la aplicación es realizar el levantamiento de inventario de todas las instituciones de salud donde se encuentra desplegado.

Confiabilidad

RnF 2.1. Al ocurrir un error el sistema muestra un mensaje indicando dónde se ha producido el mismo para que pueda ser corregido por el usuario.

Soporte

RnF 3.1. La implementación del sistema se regirá por las especificaciones del estándar de codificación del lenguaje Java: Java Code Conventions.

RnF 3.2. Se utilizarán como normas para el nombrado la notación CamellCasing para identificar las variables, parámetros y métodos.

Restricciones de diseño

RnF 4.1. El sistema estará desarrollado en base a las políticas del software libre, que fueron ajustadas al sistema nacional de salud como: uso de servidores GNU/Linux.

RnF 4.2. Se utilizará para el desarrollo la plataforma Java SE y como lenguaje de programación Java.

RnF 4.3. Se utilizará Netbeans como IDE de desarrollo.

RnF 4.4. Para el diseño de las interfaces se utilizará la librería swing de java.

RnF 4.5. Se utilizará Visual Paradigm como herramienta para el modelado de los artefactos generados por la metodología RUP.

Requisitos para la documentación de usuarios en línea y ayuda del sistema.

RnF 5.1. La aplicación de escritorio contará con una ayuda donde el usuario podrá suplir las dudas que se le puedan presentar durante la utilización de la misma. El usuario del sistema deberá recibir un adiestramiento previo en la utilización del sistema con el fin de que pueda explotar las prestaciones del sistema sin contratiempos ocasionados por la falta de preparación técnica.

Interfaz

RnF 6.1. La aplicación será sencilla e intuitiva, de fácil navegación por parte del usuario. Estará diseñada para una óptima visualización siendo adaptable a cualquier resolución.

RnF 6.2. La entrada de datos incorrecta será detectada claramente e informada al usuario.

Estándares aplicables

RnF 7.1. La aplicación se desarrollará siguiendo las normas de Integración de Modelos de Madurez de Capacidades o *Capability Maturity Model Integration (CMMI)*, un modelo para la mejora y evaluación de procesos para el desarrollo, mantenimiento y operación de sistemas de software.

Hardware

RnF 8.1. Para que el sistema se ejecute correctamente en la computadora del cliente la misma debe contar con un procesador Pentium o AMD a 800 MHz o superior, 256 MB RAM o superior y 4 GB de espacio libre en el disco duro como mínimo.

2.5. Definición de los casos de uso del sistema

Definición de los actores

Un actor es un agente o sistema externo que interactúa con el sistema. Entra, manipula o recibe información del sistema, pero es externo al mismo. Puede ser una persona o una aplicación externa que intercambia datos con el sistema. Son candidatos a actor del sistema los trabajadores del negocio que tienen actividades a automatizar, así como los actores del negocio que van a interactuar con el sistema.

Capítulo 2: Características del sistema

Actor	Objetivo
Especialista	Es el actor del sistema encargado de la gestión de inventario de equipos médicos.

Tabla 4: Descripción de actores del sistema.

Definición de caso de uso del sistema (CUS)

Son artefactos narrativos que describen, bajo la forma de acciones y reacciones, el comportamiento del sistema desde el punto de vista del usuario. Por lo tanto, establece un acuerdo entre clientes y desarrolladores sobre las condiciones y posibilidades (requisitos) que debe cumplir el sistema. En la presente investigación fueron definidos los siguientes CUS:

- CUS1:** Insertar equipo médico.
- CUS 2:** Modificar equipo médico.
- CUS 3:** Eliminar equipo médico.
- CUS 4:** Buscar equipo médico
- CUS 5:** Listar equipos médicos.
- CUS 6:** Generar reporte de equipo médico.
- CUS 7:** Autenticar
- CUS 8:** Cargar usuario.
- CUS 9:** Configurar la conexión.
- CUS 10:** Actualizar datos en la aplicación.
- CUS 11:** Seleccionar unidad de salud.
- CUS 12:** Eliminar unidad de salud.
- CUS 13:** Actualizar datos en el servidor.
- CUS 14:** Cargar equipo médico.
- CUS 15:** Cargar unidad de salud.
- CUS 16:** Cargar nomenclador de ubicación.
- CUS 17:** Cargar programa de salud.
- CUS 18:** Cargar nomenclador de equipo médico.
- CUS 19:** Cargar especialista.
- CUS 20:** Proponer mantenimiento anual

Diagrama de casos de uso del sistema

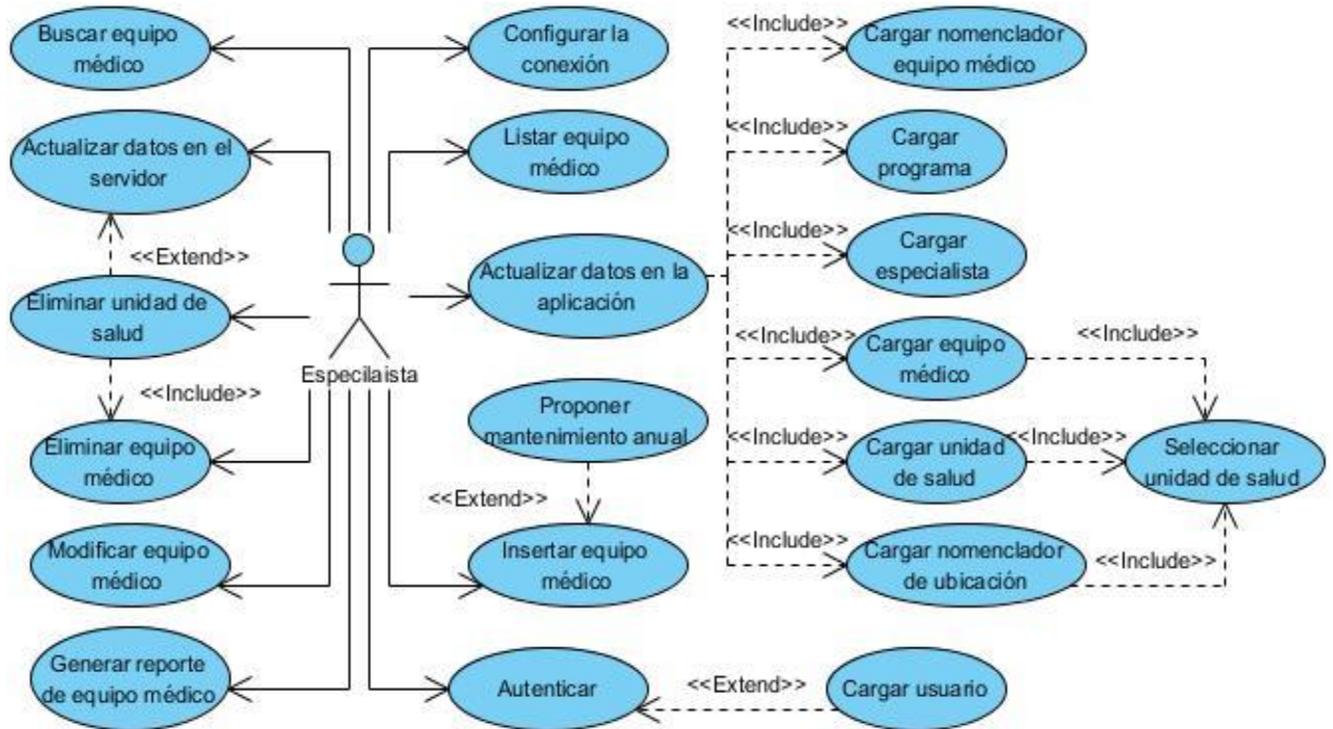


Figura # 4: Diagrama de casos de uso del sistema.

Descripción de los casos de uso del sistema

Antes de insertar un equipo médico se deben definir los parámetros de configuración para posteriormente establecer la conexión con la base de datos de SIGICEM. Luego es necesario autenticarse para poder acceder al sistema, con el objetivo de actualizar los datos en la aplicación y cargar lo equipos pertenecientes a la(s) unidad(es) de salud previamente seleccionadas. Finalmente se podrá insertar un nuevo equipo médico si necesidad de estar conectados, los cuales serán guardados en un fichero serializado.

Para un mejor entendimiento se describen los casos de usos más importantes del sistema, teniendo en cuenta el flujo de actividades explicado, los cuales serán abordados en el resto de las fases del desarrollo del sistema de los próximos capítulos.

Capítulo 2: Características del sistema

Objetivo	Insertar un equipo médico.
Actores	Especialista
Resumen	El caso de uso inicia cuando el actor accede a la opción Adicionar equipo de la interfaz Inicio, el sistema brinda la posibilidad de introducir y/o seleccionar los datos para insertar un nuevo equipo médico, el actor introduce los datos necesarios, el sistema crea un nuevo equipo médico, el caso de uso termina.
Referencia	RF 1

Tabla 5: CUS Insertar equipo médico.

Objetivo	Configurar la conexión
Actores	Especialista.
Resumen	El caso de uso inicia cuando el actor accede a la opción Configurar del menú Conexión, el sistema brinda la posibilidad de introducir los datos de configuración, el actor introduce los datos necesarios, el sistema guarda los datos, el caso de uso termina.
Referencia	RF 9

Tabla 6: CUS Configurar la conexión.

Objetivo	Actualizar datos en la aplicación.
Actores	Especialista.
Resumen	El caso de uso inicia cuando el actor accede a la opción Actualizar datos en la aplicación, el sistema guarda temporalmente los nomencladores de ubicación y los nombres de las unidades de salud, posteriormente brinda la posibilidad de seleccionar una o varias unidades de salud, el actor selecciona la o las unidades de salud que desee y el sistema carga además de todos los datos de dichas unidades, sus equipos médicos y nomencladores de ubicación correspondientes. El sistema carga también los programas, los especialistas y los nomencladores de equipo médico, el caso de uso termina.
Referencia	RF 10

Tabla 7: CUS Actualizar datos en la aplicación.

Capítulo 2: Características del sistema

.Objetivo	Seleccionar unidad de salud.
Actores	Especialista.
Resumen	El caso de uso inicia cuando el actor accede a la opción Actualizar datos en la aplicación del menú Base de Datos, el sistema brinda la posibilidad de seleccionar varias unidades de salud, el actor selecciona la o las unidades de salud donde trabajará, el caso de uso termina.
Referencia	RF 11

Tabla 8: CUS Seleccionar unidad de salud.

Objetivo	Cargar equipo médico
Actores	Especialista
Resumen	El caso de uso inicia cuando el actor accede a la opción Actualizar datos en la aplicación del menú Base Datos, el sistema carga los equipos existentes de la(s) unidad(es) de salud seleccionada(s), termina el caso de uso.
Referencia	RF 14

Tabla 9: CUS Cargar equipo médico.

Mediante la descripción del modelado del negocio se obtuvieron las principales características que debe tener el Sistema para el levantamiento de inventario de equipos médicos por los especialistas del CICEM, teniendo en consideración la funcionalidad insertar equipo médico del Módulo de Inventario de SIGICEM, concretando una propuesta de solución con la especificación de sus funcionalidades, las que se describieron en términos de casos de usos para una mayor comprensión de sus usuarios finales. Partiendo de esta propuesta se diseñará el mismo.

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

El objetivo del flujo de trabajo Análisis y Diseño es traducir los requisitos a una especificación que describe cómo implementar el sistema, desarrollar una arquitectura para el sistema y adaptar el diseño para que sea consistente con el entorno de implementación. El análisis consiste en obtener una visión del sistema que se preocupa de ver qué hace, de modo que sólo se interesa por los requisitos funcionales. Por otro lado el diseño es un refinamiento del análisis que tiene en cuenta los requisitos no funcionales, en definitiva cómo cumple el sistema sus objetivos.

3.1. Descripción de la arquitectura. Fundamentación.

Una de las acciones que contribuyen a un buen desarrollo de software, es saber escoger sabiamente cual será la arquitectura de software que regirá el ambiente de trabajo, definiéndose esta como “las estructuras de un sistema, compuestas de elementos con propiedades visibles de forma externa y las relaciones que existen entre ellos”¹⁸. Por este motivo el sistema a desarrollar, teniendo en cuenta la arquitectura definida por el proyecto, será implementado siguiendo el patrón de arquitectura de software Modelo-Vista-Controlador (MVC), el cual separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos.¹⁹

Modelo: contiene ficheros serializados donde se guardarán los datos utilizados por la aplicación y una clase AccesoDatos que permite obtener la información de la base de datos.

Vista: es la interfaz gráfica de usuario, se encarga de manejar la visualización de la información.

Controlador: Es el encargado de capturar todas las peticiones provenientes de la interfaz de usuario y de controlar el flujo de datos entre la vista y el modelo.

Se hará uso del patrón MVC por las ventajas siguientes:

- Clara separación entre el modelo, la vista y el controlador.
- Facilidad para la realización de pruebas unitarias de los componentes.

¹⁸L. Bass, P. Clements, R. Kazman. *Software Architecture in Practice. 2nd Edition. s.l. : Addison Wesley, 2003.*

¹⁹ComuSOFT. [En línea] 2011. [Citado el: 22 de abril de 2013.] <http://www.comusoft.com/modelo-vista-controlador-definicion-y-caracteristicas>.

Capítulo 3: Análisis y Diseño del sistema

- Reutilización de los componentes.
- Simplicidad en el mantenimiento de los sistemas.
- Facilidad para desarrollar prototipos rápidos.
- Los desarrollos suelen ser más escalables.

Como se explicaba en el capítulo 1 el conjunto de librerías gráficas Swing se basa en el patrón MVC, pero este a su vez presenta algunas diferencias en su implementación:

- El controlador y la vista están implementados como un único elemento denominado delegado de interfaz de usuario (*UI delegate* o *UI object*).
- Cada componente está asociado a un modelo del dominio a través de un modelo de aplicación y del *UI delegate* por tanto cada componente tiene asociado un delegado de interfaz que a su vez, está incorporado a un modelo de aplicación que proporciona toda la información que el delegado requiere y que notifica al componente si algún cambio se ha dado en el modelo del dominio.

En la figura 5 se observa también la existencia de un elemento denominado administrador de interfaz de usuario (*UI manager*); el cual es el encargado de realizar los cambios en el modelo de dominio permitiendo conectar un *look and feel* específico a cada componente visualizado, brindando la posibilidad de modificarle a los mismos diversos atributos como son el color de fondo por defecto, tipo de letra, tipos de bordes entre otros. Todos los componentes de Swing están manejados por el mismo administrador de interfaz de usuario.

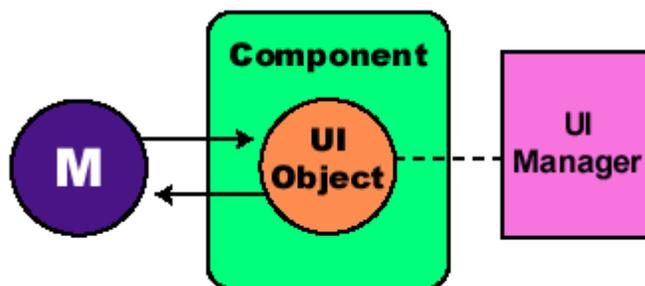


Figura # 5: Funcionamiento arquitectura Swing

3.2. Análisis de posibles implementaciones o componentes existentes que puedan ser rehusados.

La reutilización de código no es más que una técnica o mecanismo de programación que garantiza que una parte o totalidad de una aplicación existente pueda emplearse en la construcción de otra. De esta forma, se elimina la repetición de código, economiza el tiempo y el costo de desarrollo. Dado que el sistema a implementar es de escritorio y SIGICEM es una aplicación web es imposible reutilizar algunos componentes de este último debido a que la tecnología para la que están soportados ambos es diferente.

3.3. Análisis

Modelo de Análisis

El modelo de análisis ofrece una especificación más precisa de los requisitos y estructura los mismos de un modo que nos facilita su comprensión, preparación, modificación y su mantenimiento en general. Dicho modelo se describe utilizando el lenguaje de los desarrolladores, lo cual permite introducir un mayor formalismo y ser utilizado para razonar sobre los funcionamientos internos del sistema.

Ventajas del modelo de análisis:

- Suaviza la transición al diseño.
- Apoya el cambio a otra plataforma de programación.
- Sirve para tener una visión general de la propuesta del sistema.
- Sirve para planificar y dividir el Diseño e Implementación en pequeños módulos, lo cual facilita el trabajo con diferentes equipos de desarrolladores.
- Apoya la aplicación de reingeniería a aplicaciones existentes, ya que el lenguaje es menos técnico y se puede entender mucho más fácil la propuesta de solución.

Este modelo está compuesto por varios artefactos entre ellos se encuentran las clases de análisis, las cuales se centran en los requisitos funcionales y son evidentes en el dominio del problema porque representan conceptos y relaciones del dominio. RUP propone clasificarlas en:

- **Interfaz:** Modelan la interacción entre el sistema y sus actores.

- **Control:** Coordinan la realización de uno o unos pocos casos de uso coordinando las actividades de los objetos que implementan la funcionalidad del caso de uso.
- **Entidad:** Modelan información que posee larga vida y que es a menudo persistente.

Diagramas de clases de análisis

Los diagramas de clases de análisis de los casos de uso descritos en el capítulo anterior reflejan la interacción entre el usuario y el sistema mediante una interfaz, encargada de la comunicación con la clase controladora, que procesa los datos y devuelve la información por la misma interfaz.

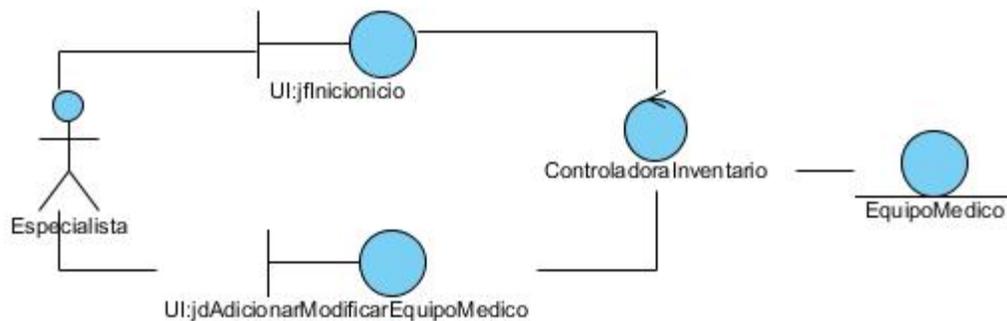


Figura # 6: Diagrama de clases de análisis. Caso de uso: Insertar Equipo Médico.

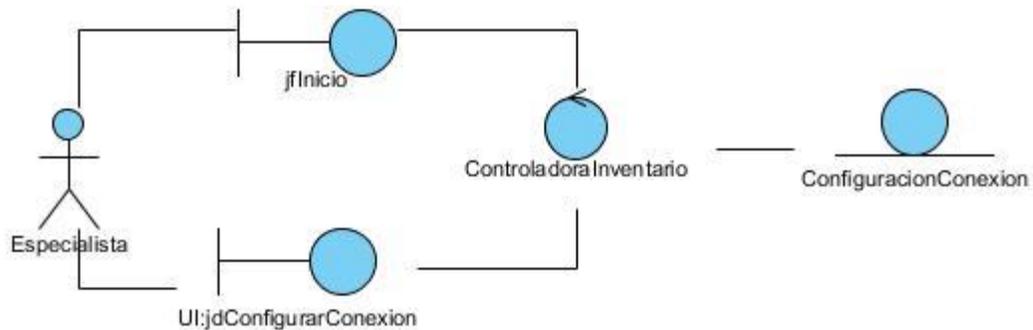


Figura # 7: Diagrama de clases de análisis. Caso de uso: Configurar Conexión.

Capítulo 3: Análisis y Diseño del sistema

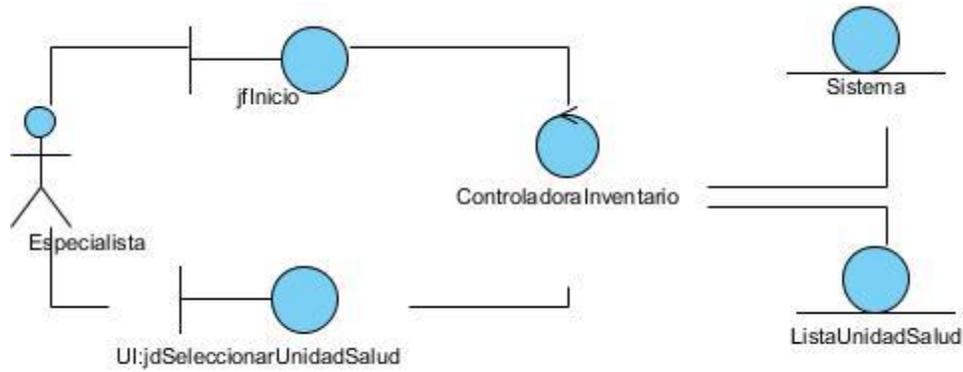


Figura # 8: Diagrama de clases de análisis. Caso de uso: Seleccionar unidad de salud.

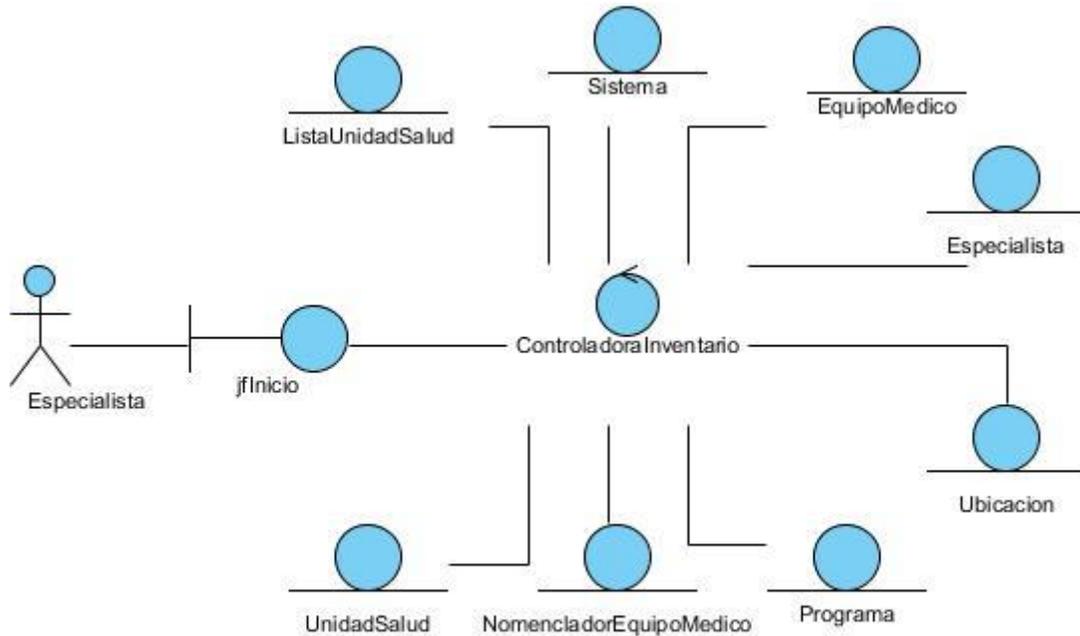


Figura # 9: Diagrama de clases de análisis. Caso de uso: Actualizar datos en la aplicación.

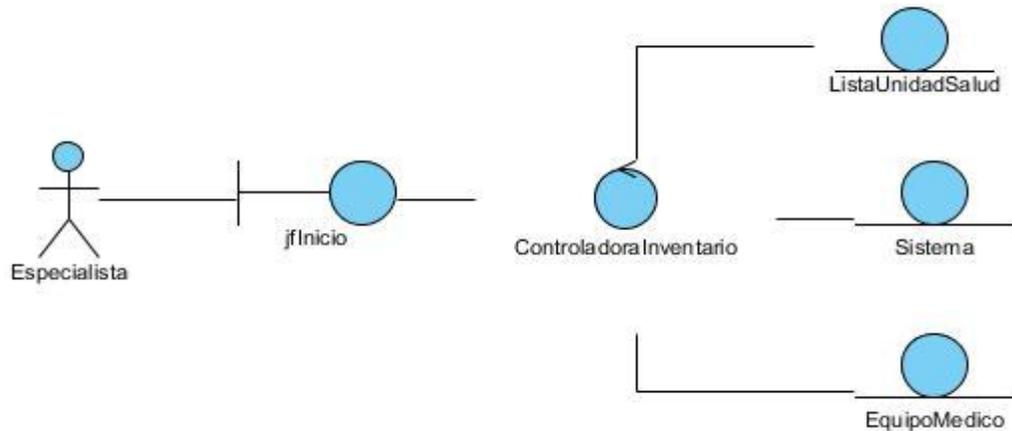


Figura # 10: Diagrama de clases de análisis. Caso de uso: Cargar equipo médico.

Diagramas de interacción. Comunicación

Un diagrama de interacción consiste en un conjunto de objetos y sus relaciones, incluyendo los mensajes que se pueden enviar entre ellos. Estos diagramas se pueden clasificar de dos tipos: de secuencia y de comunicación. Los de secuencia enfatizan el ordenamiento temporal de los mensajes mientras que los de comunicación muestran la organización estructural de los objetos que envían y reciben mensajes. Debido a que ambos diagramas son semánticamente equivalentes solamente fueron elaborados los diagramas de comunicación de los casos de uso descritos.

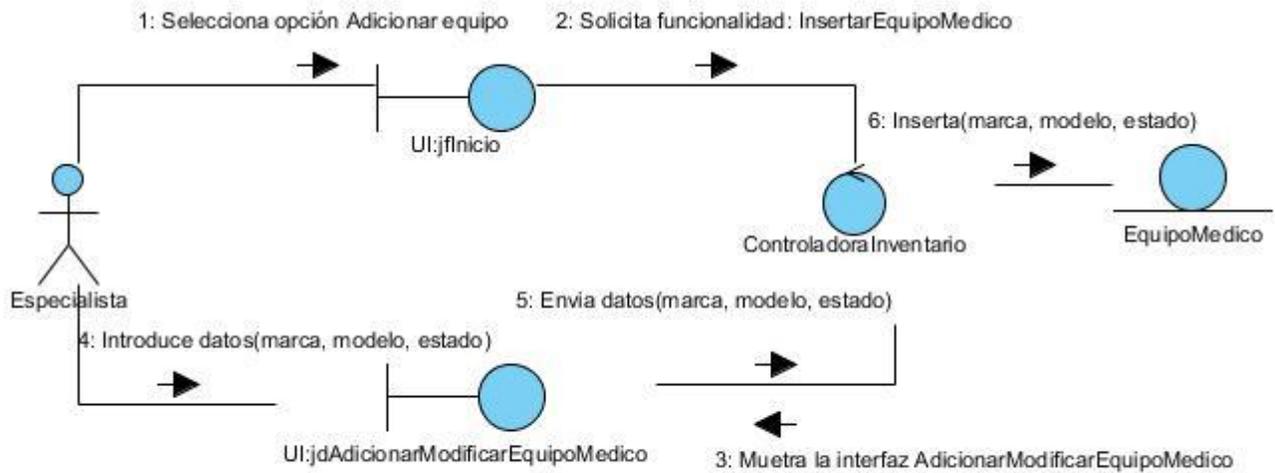


Figura # 11: Diagrama de comunicación Caso de uso: Insertar equipo médico.

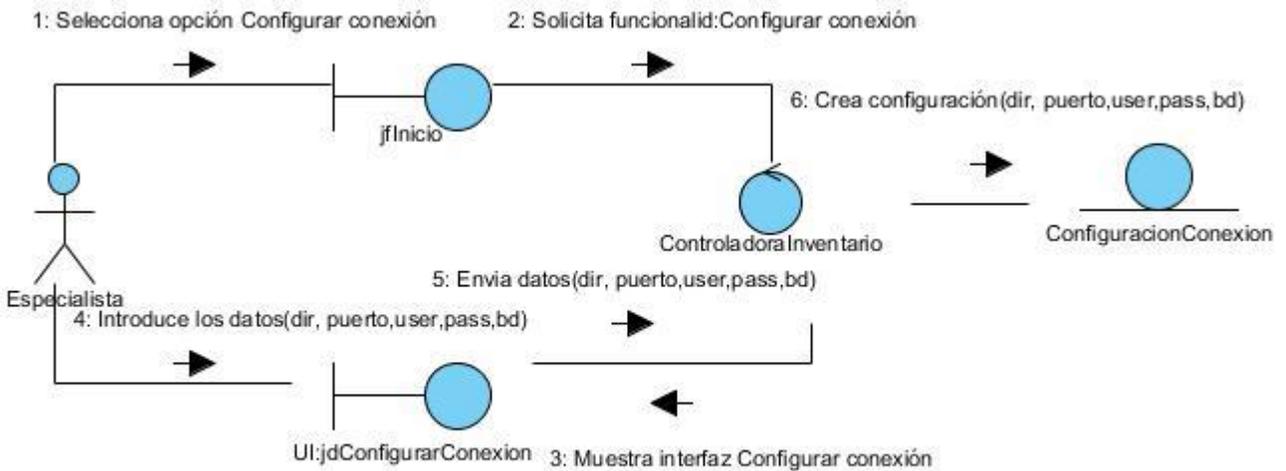


Figura # 12: Diagrama de comunicación. Caso de uso: Configurar la conexión.

Capítulo 3: Análisis y Diseño del sistema

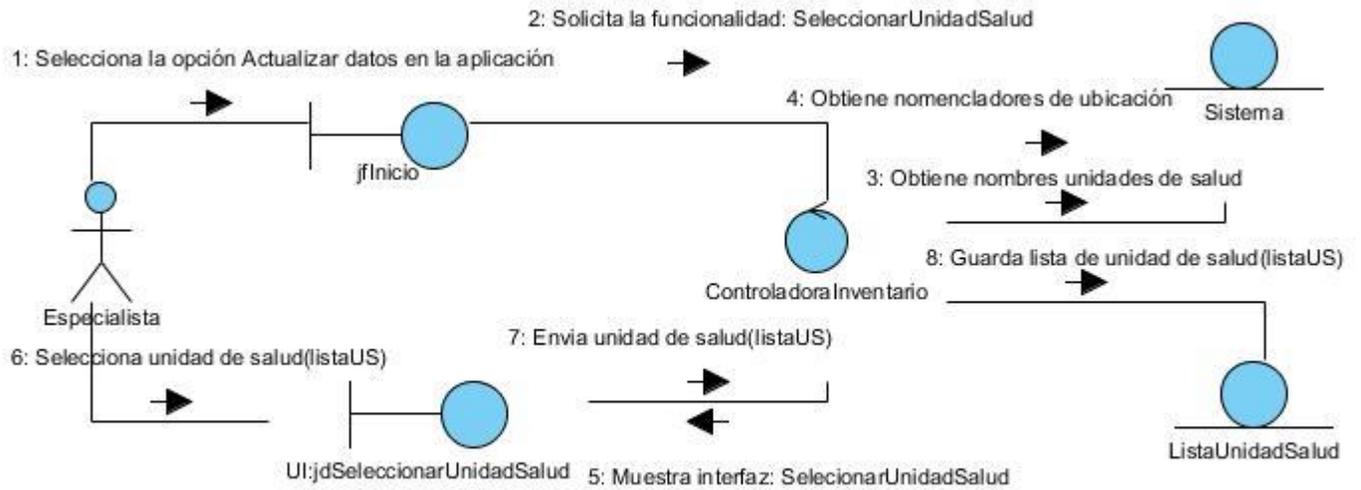


Figura # 13: Diagrama de comunicación. Caso de uso: Seleccionar unidad de salud.

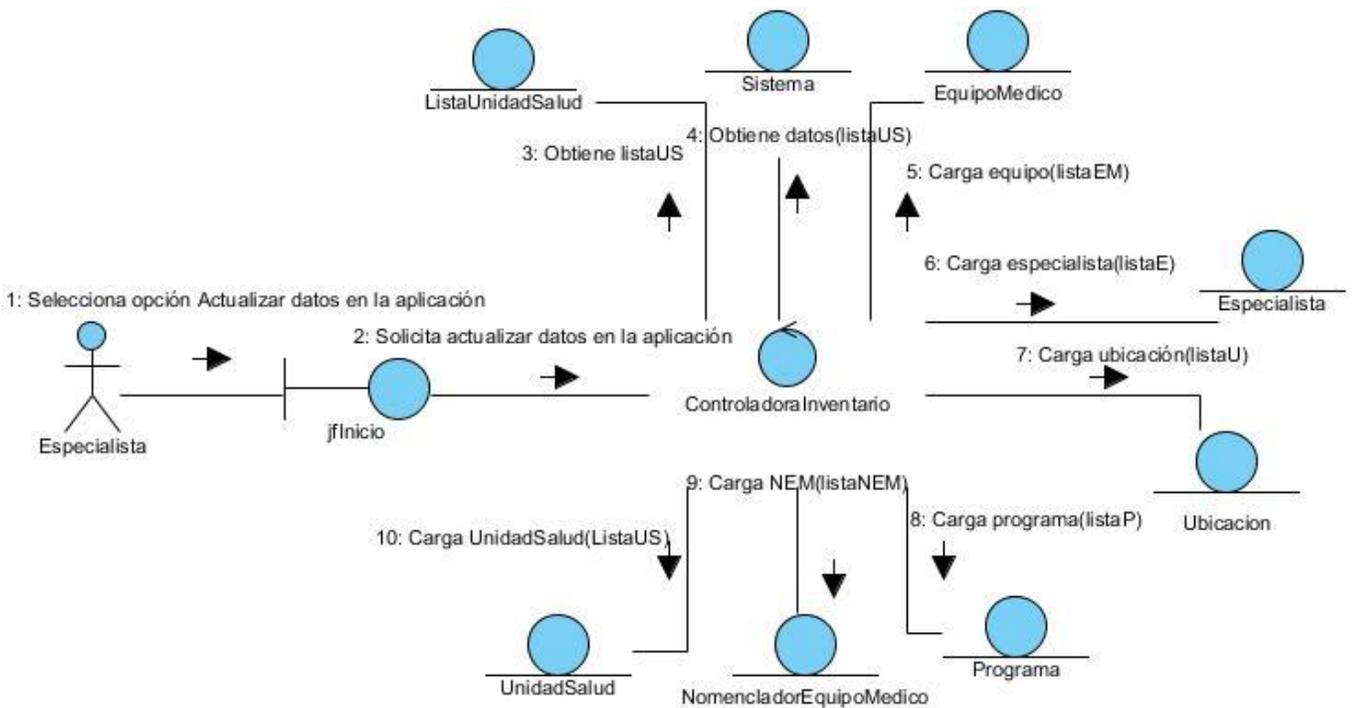


Figura # 14: Diagrama de comunicación. Caso de uso: Actualizar datos en la aplicación.

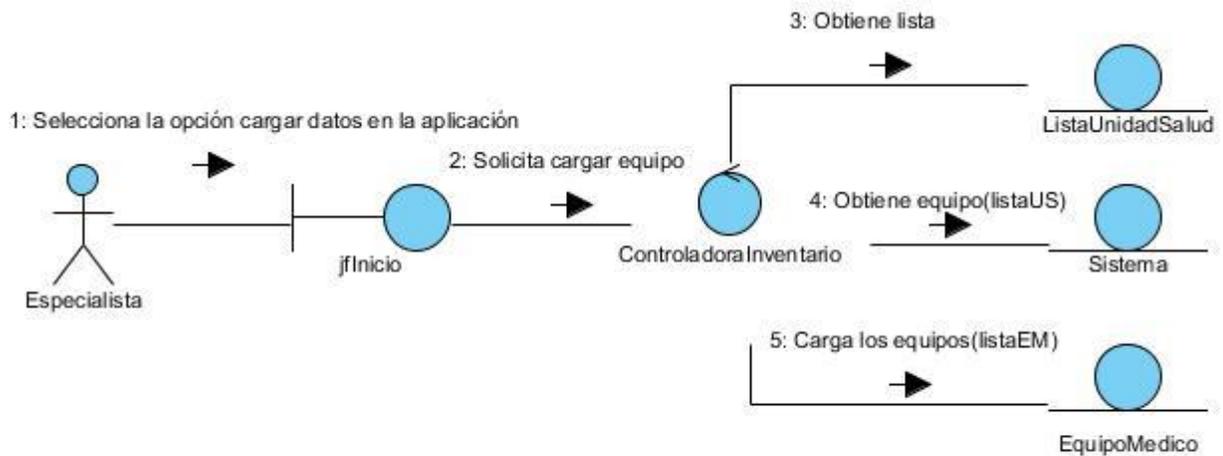


Figura # 15: Diagrama de comunicación. Caso de uso: Cargar equipo médico.

3.4. Diseño

Modelo de Diseño

Durante el flujo de trabajo de diseño, se modela el sistema de manera que soporte todos los requerimientos, incluyendo a diferencia del análisis, a los requerimientos no funcionales. El modelo de diseño se puede utilizar para visualizar la implementación y para soportar las técnicas de programación gráfica de la aplicación. Este modelo describe la realización física de los casos de uso centrándose en cómo los requisitos funcionales y no funcionales, junto con otras restricciones relacionadas con el entorno de implementación, tienen impacto en el sistema a considerar. Además, sirve de abstracción de la implementación del sistema y es, de ese modo, utilizado como una entrada fundamental de las actividades de implementación.

Una clase de diseño es una abstracción de una clase o construcción similar en la implementación del sistema. El lenguaje utilizado para especificar estas clases es el mismo que el lenguaje de programación que se va a utilizar para el desarrollo del software, las relaciones de estas con otras clases a menudo tienen un significado directo cuando la clase es implementada. Los diagramas de clase de diseño contienen las clases, sus objetos y también los subsistemas que contienen las clases de diseño.

Diagramas de clases del diseño

En busca de concebir un diagrama de clases del diseño con poca complejidad de interpretación y mayor organización; fueron explicados anteriormente los elementos relacionados con el modelo, la vista y el controlador que se definen en el patrón Modelo-Vista-Controlador, reflejado en los siguientes diagramas.

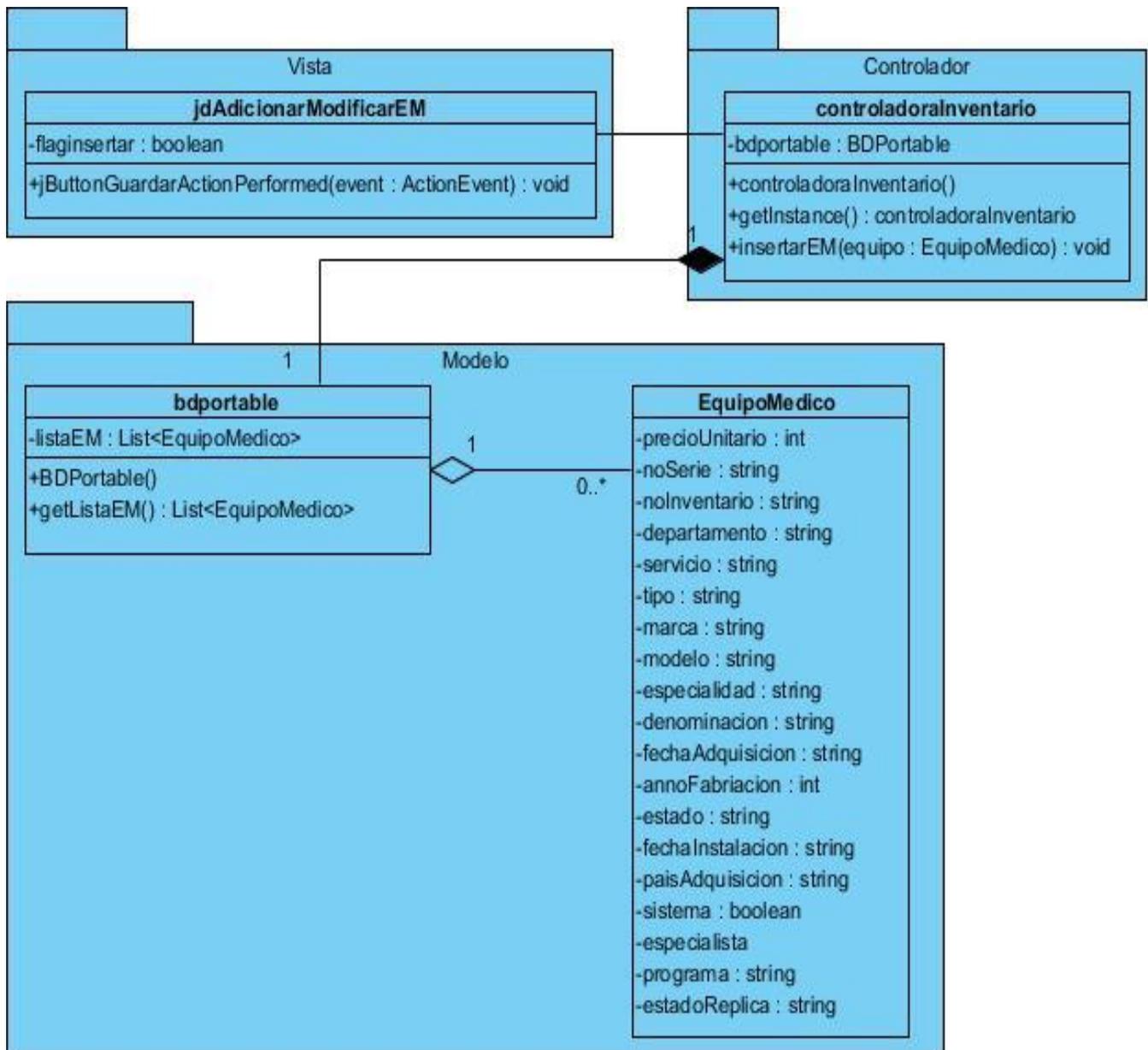


Figura # 16: Diagrama de clases de diseño. Caso de uso: Insertar equipo médico.

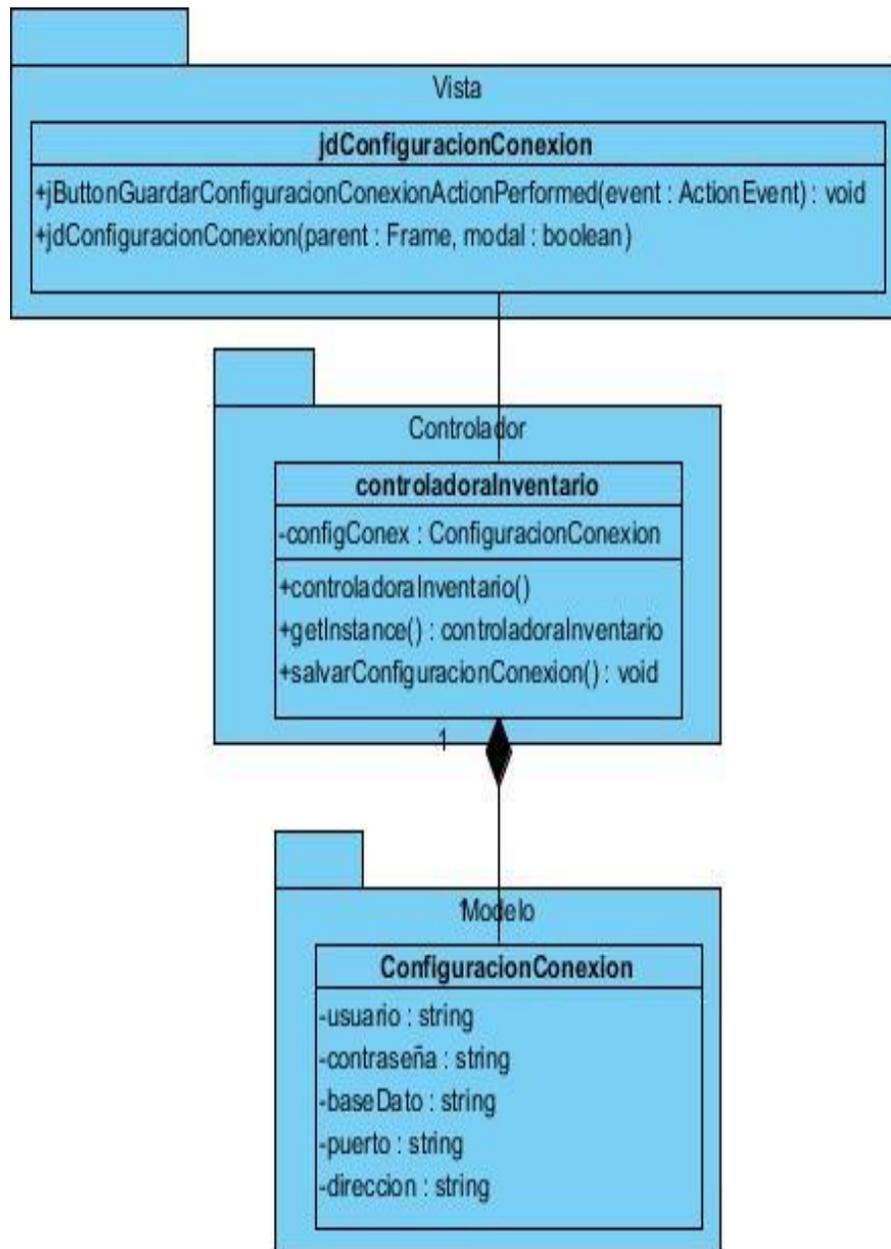


Figura # 17: Diagrama de clases de diseño. Caso de uso: Configurar la conexión.

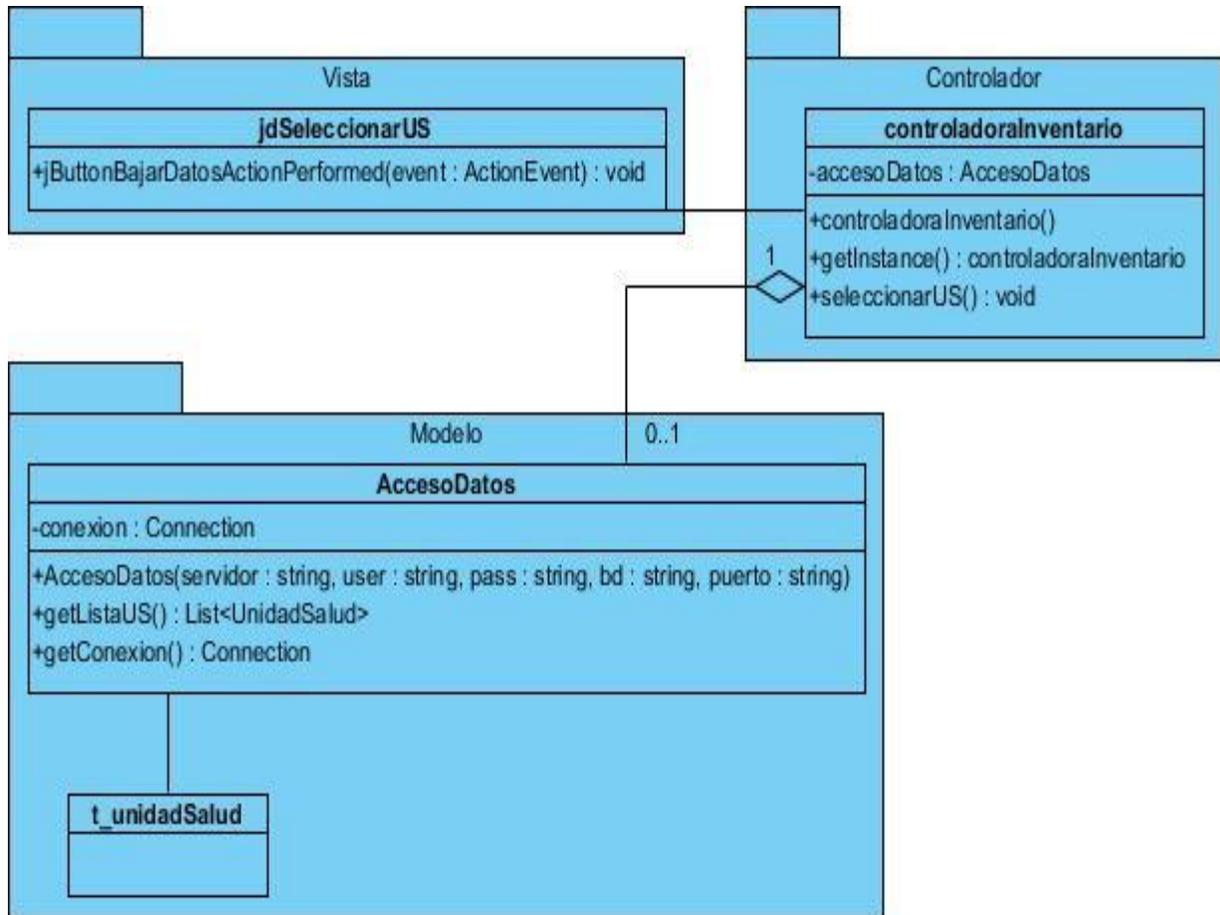


Figura # 19: Diagrama de clases de diseño. Caso de uso: Seleccionar unidad de salud.

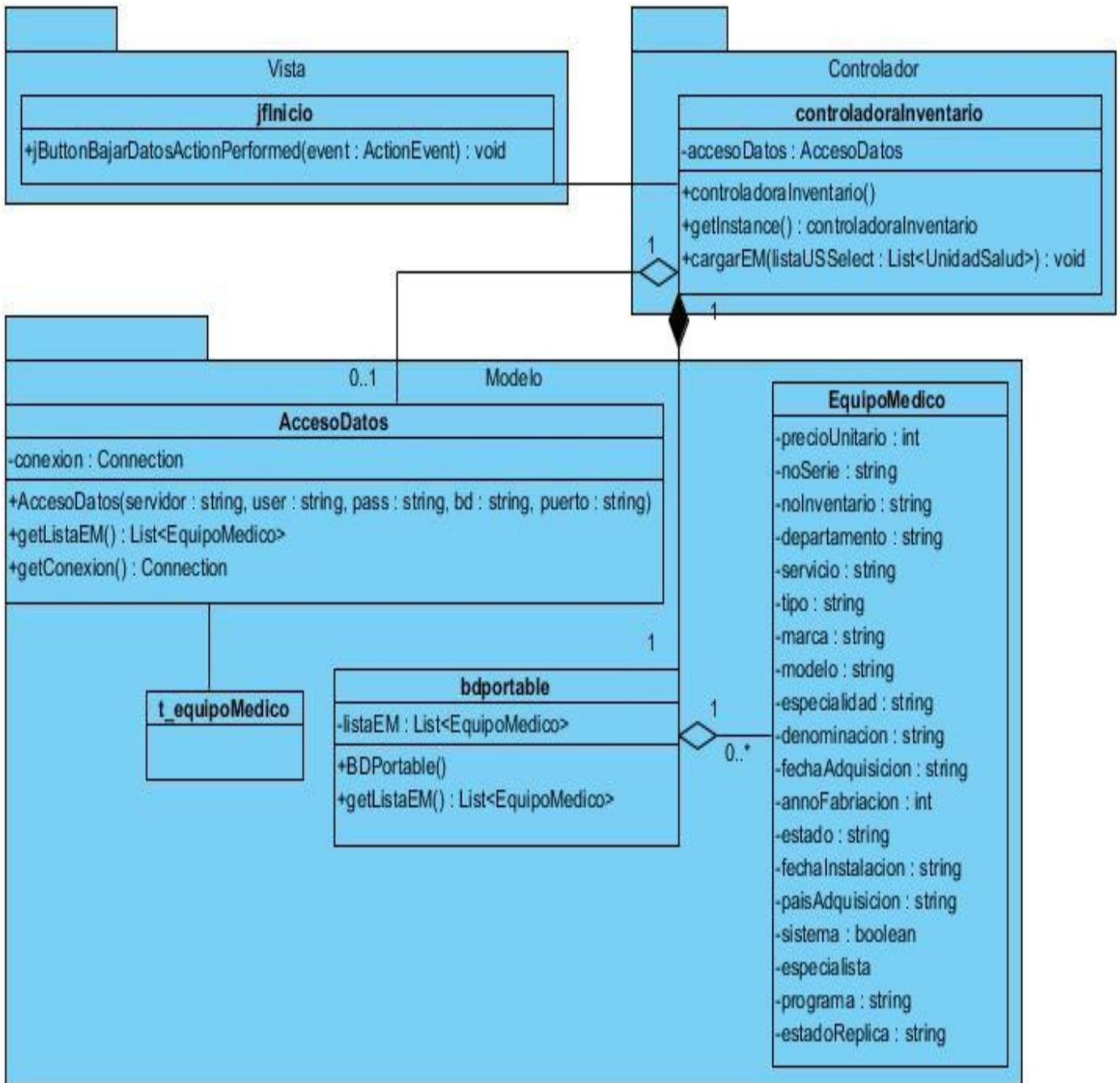


Figura # 20: Diagrama de clases de diseño. Caso de uso: Cargar equipo médico.

Descripción de las clases

Nombre:	controladorInventario
Tipo de clase:	controladora
Descripción General	
Contiene la lógica de la aplicación del sistema, la misma se encuentra compuesta por variables y métodos que a su vez utilizan el modelo.	
Nombre:	bdportable
Tipo de clase:	entidad
Descripción General	
Contiene métodos para leer y escribir la información que gestiona el sistema en un fichero.	
Nombre:	AccesoDatos
Tipo de clase:	controladora
Descripción General	
Contiene los métodos y variables necesarios para obtener la información de la base de datos.	
Nombre:	PDF
Tipo de clase:	controladora
Descripción General	
Contiene el código necesario para generar los distintos reportes en formato (PDF) que brinda el sistema.	
Nombre:	HeaderFooter
Tipo de clase:	controladora
Descripción General	
Contiene métodos para insertar el encabezado y el pie de página en los distintos reportes en formato (PDF) que brinda el sistema.	
Nombre:	MetodosAuxiliares
Tipo de clase:	controladora

Capítulo 3: Análisis y Diseño del sistema

Descripción General	
Contiene métodos estáticos que son utilizados por la clase controladorInventario.	
Nombre:	ConfiguracionConexion
Tipo de clase:	controladora
Descripción General	
Es la representación Orientada a Objetos (OO) de los parámetros necesarios para establecer una conexión con el servidor de base de datos de SIGICEM.	
Nombre:	EntradaSalidaFichero
Tipo de clase:	controladora
Descripción General	
Contiene métodos para leer y escribir la información que gestiona el sistema en un fichero.	
Nombre:	EquipoMedico
Tipo de clase:	entidad
Descripción General	
Es la representación Orientada a Objetos (OO) de los datos de un equipo médico, contiene variables y métodos para acceder o actualizar los mismos.	
Nombre:	Especialista
Tipo de clase:	entidad
Descripción General	
Es la representación Orientada a Objetos (OO) de los datos de un especialista en Electromedicina, contiene variables y métodos para acceder o actualizar los mismos.	
Nombre:	jdAdicionarModificarEM
Tipo de clase:	interfaz
Descripción General	
Es donde se insertar o modifican los equipos médicos, contiene el código de la presentación.	

Tabla 10: Descripción de las clases empleadas en el los diagramas de clases del diseño.

Capítulo 3: Análisis y Diseño del sistema

En este capítulo se ilustraron los elementos más significativos de las fases de análisis y diseño del sistema, obteniéndose los diagramas de clases del análisis, que permiten establecer las relaciones entre las clases interfaz, controladoras y entidad. En la etapa de diseño quedaron definidos los diagramas de comunicación, los que se elaboraron a partir de artefactos de la etapa de análisis. Esto permitió asignar responsabilidades a las clases y razonar sobre el funcionamiento interno del sistema. Se obtuvo el diseño de la interacción de los componentes mediante los diagramas de clases del diseño; logrando finalmente una arquitectura base para el sistema sobre la cual se trabajará en la fase de implementación.

CAPÍTULO 4: IMPLEMENTACIÓN

El objetivo principal de esta disciplina de Implementación es convertir los elementos del diseño en elementos de implementación, dichos elementos son códigos fuentes, ejecutables, binarios, entre otros. De esta disciplina se obtiene un sistema ejecutable estable, constituido de los resultados producidos por los programadores.

4.1. Modelo de Implementación

El modelo de implementación describe cómo los elementos del modelo de diseño se implementan en términos de componentes. Describe también cómo se organizan los componentes de acuerdo con los mecanismos de estructuración y modularización disponibles en el entorno de implementación y en el lenguaje o lenguajes de programación utilizados y cómo dependen los componentes unos de otros²⁰.

Diagrama de componentes

Los diagramas de componentes ilustran los componentes del software que serán usados para construir el sistema. El uso más importante de estos diagramas es mostrar la estructura de alto nivel del modelo de implementación, especificando las dependencias entre tipos de componentes y organiza los subsistemas de implementación en capas.

También se utilizan para mostrar las dependencias de compilación de los ficheros de código, relaciones de derivación entre ficheros de código fuente y ficheros que son resultados de la compilación, dependencias entre elementos de implementación y los correspondientes elementos de diseño que son implementados.

Se utilizan para modelar la vista estática de un sistema. Muestran la organización y las dependencias lógicas entre un conjunto de componentes de software, sean estos componentes de código fuente, librerías, binarios o ejecutables.

²⁰ Acuña, Karenny Brito. *Selección de metodologías de desarrollo para aplicaciones web en la facultad de Informática en la Universidad de Cienfuegos*. [En línea] [Citado el: 5 de abril de 2013.] <http://www.eumed.net/libros-gratis/2009c/584/RUP%20Diseno%20e%20implementacion%20del%20sistema.htm>.

A continuación se especifican las dependencias entre cada uno de los componentes de la arquitectura del sistema.

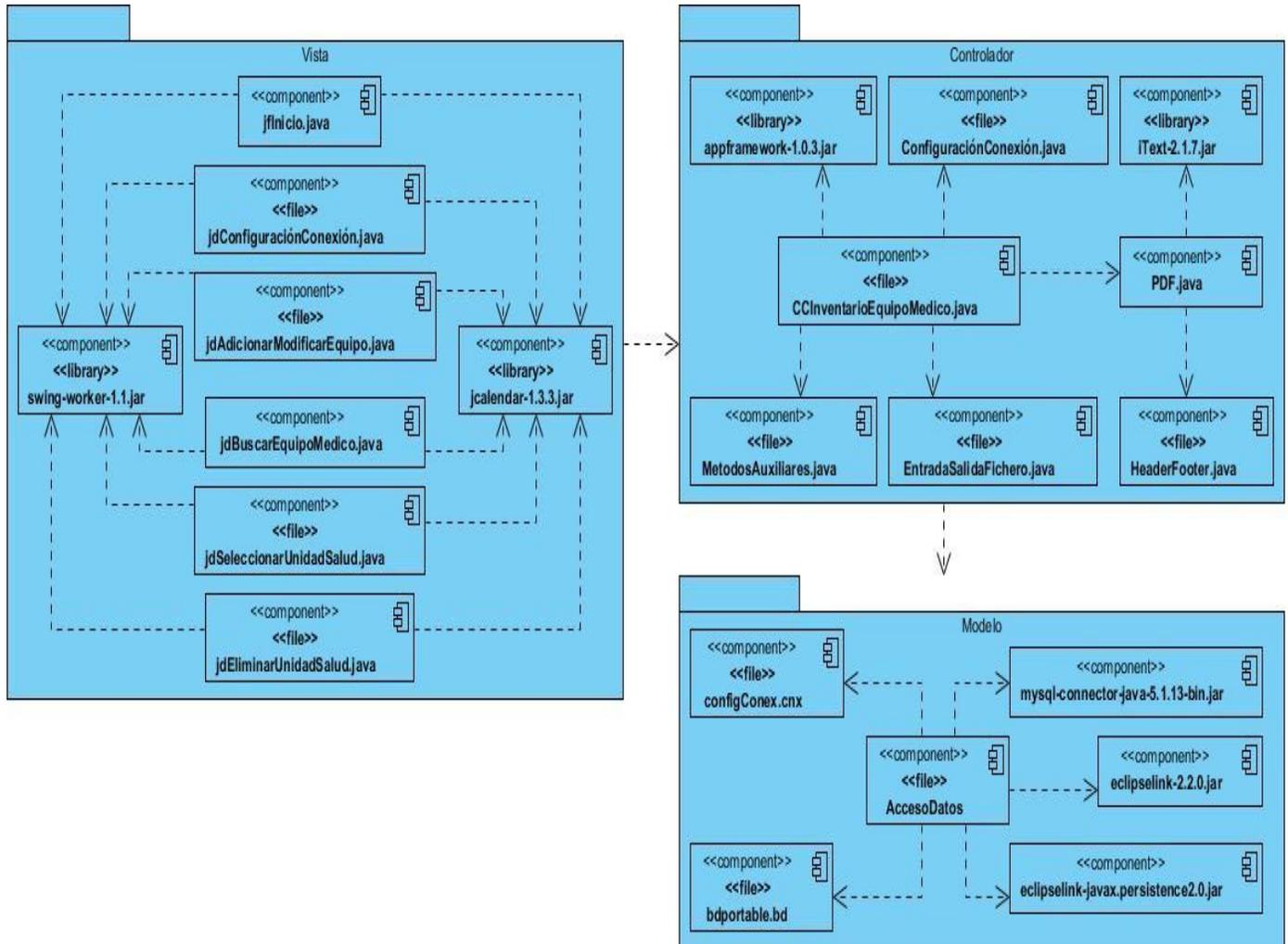


Figura # 21: Diagrama de componentes del Sistema para el levantamiento de inventario de equipos médicos.

Modelo de despliegue

El modelo de despliegue es un modelo de objetos que describe la distribución física del sistema en términos de cómo se distribuye la funcionalidad entre los nodos de computo. El modelo de despliegue se utiliza como entrada fundamental en las actividades de diseño e implementación debido a que la distribución del sistema tiene una influencia principal en su diseño.

En la figura 22 se muestra el diagrama de despliegue del sistema para el levantamiento de inventario de equipos médicos, en el cual la computadora cliente deberá tener instalada la maquina virtual de java para lograr el funcionamiento del sistema y por ende el levantamiento de inventario de equipos médicos por parte del especialista. En caso de que exista la posibilidad de establecer una conexión con el servidor de base de datos siguiendo el protocolo TCP/IP el sistema permitirá actualizar o extraer de dicho servidor el o los equipos médicos pertenecientes a una o varias unidades de salud. El sistema genera reportes en formato .pdf, los cuales además de ser guardado pueden imprimirse en caso de que existe alguna impresora en la unidad de salud que se conecte a la computadora por el puerto USB o LPT.

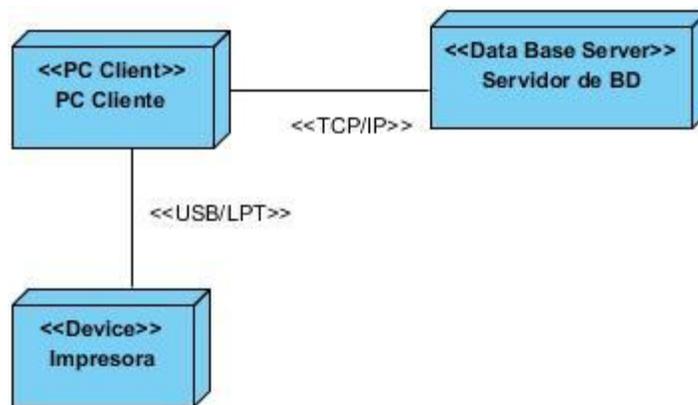


Figura # 22: Diagrama de despliegue del Sistema para el levantamiento de inventario de equipos médicos.

4.2. Tratamiento de errores

El tratamiento de errores lo tenemos presente en toda la elaboración de la aplicación, ya que en un sistema en ocasiones ocurren errores que atentan contra la estabilidad y el correcto funcionamiento del mismo a la hora de introducir, buscar o eliminar datos, por lo que se hace necesario realizar acciones concretas con el objetivo de mitigar dichos errores.

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

4.3. Seguridad e Integridad de la información

La seguridad informática es la disciplina que se ocupa de diseñar las normas, procedimientos, métodos y técnicas, orientados a proveer condiciones seguras y confiables, para el procedimiento de datos en sistemas informáticos. Para lograr sus objetivos, la seguridad informática se fundamenta en tres principios, que debe cumplir todo sistema informático:

- **Confidencialidad:** se refiere a la privacidad de los elementos de información almacenados y procesados en un sistema informático.
- **Integridad:** se refiere a la validez y consistencia de los elementos de información almacenados y procesados en un sistema informático
- **Disponibilidad:** se refiere a la continuidad de acceso a los elementos de información almacenados y procesados en un sistema informático.

Basándose en el principio de integridad, las herramientas de seguridad informática deben asegurar que los procesos de actualización estén sincronizados y no se dupliquen, de forma que todos los elementos del sistema manipulen adecuadamente los mismos datos. Este principio es particularmente importante en sistemas descentralizados, es decir, aquellos en los que diferentes usuarios, computadores y procesos comparten la misma información.²¹

Garantizar la seguridad y la integridad es de vital importancia para cualquier software que gestione información. Es por ello que el acceso al sistema para el levantamiento de inventario de equipos médicos es controlado mediante privilegios de usuarios con nombre de usuario y una contraseña que solo debe conocer el especialista del CICEM que hará uso de la aplicación, garantizando con esto que nadie pueda acceder a la información, ni modificarla sin contar con la autorización establecida. Además los datos a gestionar por el sistema serán guardados en un fichero utilizando un mecanismo de serialización, que convierte una cadena de caracteres en bytes garantizando que el contenido del mismo no pueda ser leído por personas no autorizadas.

²¹ **SlideShare.** *Seguridad Informática. La información como activo estratégico.* [En línea] 2013. [Citado el: 28 de mayo de 2013.]

Para mantener la integridad de los datos que se almacenan en la base de datos, el sistema le asigna a cada equipo un estado, cuando el equipo se inserta por primera vez su estado es nuevo, cuando se edita el estado pasa a ser modificado, y en caso de que ya se haya ingresado a la base de datos el estado es viejo. Esta distinción se realiza con el objetivo de actualizar solo los equipos con estado nuevo o modificado en el servidor durante la sincronización. Luego de actualizados los datos, estos se podrán borrar o no del fichero, en este último caso todos los equipos actualizarán el estado a viejo para que no exista una duplicación de la información en un nuevo proceso de sincronización, ya que los equipos con estado viejo son irrelevantes. Este mecanismo permite ganar en tiempo de ejecución y mayor persistencia de los datos.

Además de anteriormente explicado para lograr la exactitud y fiabilidad de los datos almacenados, se utilizaron los mecanismos de validación que permiten entrar solamente datos correctos y que exista una homogeneidad de la información relacionada con el equipamiento médico. No se permiten dejar datos indispensables para realizar un inventario en blanco, considerándolos como datos obligatorios, así como las cadenas de caracteres y numéricos.

4.4. Estrategias de codificación. Estándares y estilos a utilizar.

Se definen estándares de codificación porque un estilo de programación homogéneo en un proyecto permite que todos los participantes lo puedan entender en menos tiempo y que el código en consecuencia sea mantenible²². Por esta razón para el desarrollo del sistema se utilizarán algunas especificaciones del estándar de codificación del lenguaje Java o *Java Code Conventions*.

Algunos de los estándares definidos son que el sistema a desarrollar debe utilizar como idioma el español, las palabras no se acentuarán, ni se utilizará la letra “ñ” para evitar interpretaciones erróneas por parte de las computadoras, pues se encuentra en dependencia de las configuraciones que posea la misma. También se empleará la notación *CamellCasing* para denotar variables, parámetros y métodos. Esta notación especifica que la palabra de inicio del identificador comienza con minúscula. Si el identificador

²²Calleja, Manuel Arias. Carmen. Estándares de codificación. [En línea] [Citado el: 10 de abril de 2013.] <http://www.cisiad.uned.es/carmen/estilo-codificacion.pdf>.

está compuesto por más de una palabra entonces éstas deben comenzar con mayúsculas, excepto la primera.

Los nombres de las clases utilizadas en la implementación del sistema deben ser sustantivos, cuando son compuestos tendrán la primera letra de cada palabra que lo forma en mayúsculas. Se debe intentar mantener los nombres de las clases simples y descriptivas, además de usar palabras completas, evitar acrónimos y abreviaturas. Los métodos contenidos en dichas clases deben ser verbos y serán denotados siguiendo la notación definida. Los nombres tanto de las clases y los métodos, como de los objetos y atributos deben permitir que con sólo leerlo se conozca el propósito de los mismos²³.

4.5. Solución Implementada

Se obtuvo como resultado de la presente investigación el Sistema para el levantamiento de inventario de equipos médicos por los especialistas del Centro de Ingeniería Clínica y Electromedicina, el cual permitirá llevar un control de los equipos médicos existentes en las unidades de salud que no posean la infraestructura de red necesaria para conectarse a la base de datos del SIGICEM, garantizándose con esto el correcto funcionamiento de los demás procesos relacionados con la gestión tecnológica. El sistema posee varias funcionalidades para cumplir su objetivo entre las cuales se encuentran insertar un equipo médico.

²³ Lucas Vieites. *Convenciones de código para el lenguaje de programación Java*. [En línea] mayo de 2007. [Citado el: 17 de abril de 2013.] http://blog.codexion.com/wp-content/uploads/2010/02/Convenciones_Codigo_Java.pdf

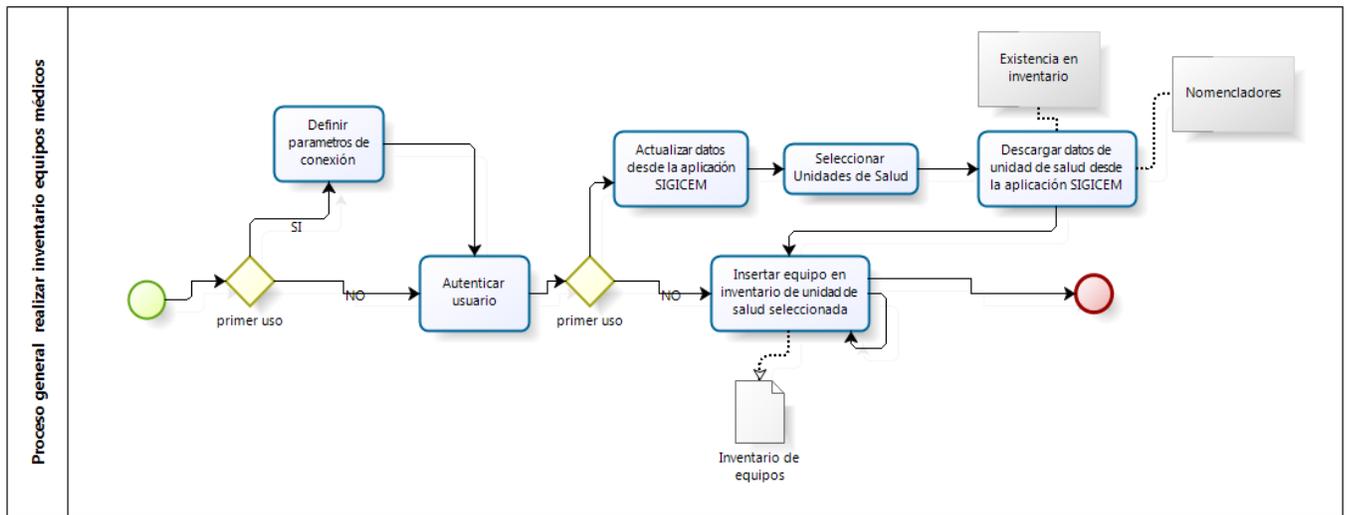


Figura # 23: Diagrama del flujo Insertar equipo médico.

Para poder realizar esta funcionalidad del sistema, el especialista debe primeramente definir los parámetros de configuración necesarios para establecer posteriormente la conexión con el servidor.



Figura # 24: Interfaz de usuario Configuración de conexión.

Capítulo 4: Implementación

Después de configurada la conexión el especialista puede acceder a la aplicación autenticándose, garantizando con esto la seguridad de la misma. Una vez dentro de la aplicación el especialista puede acceder a la opción *Actualizar datos en la aplicación* del menú *Base de Datos*, la cual se encuentra ubicada en la interfaz inicio.

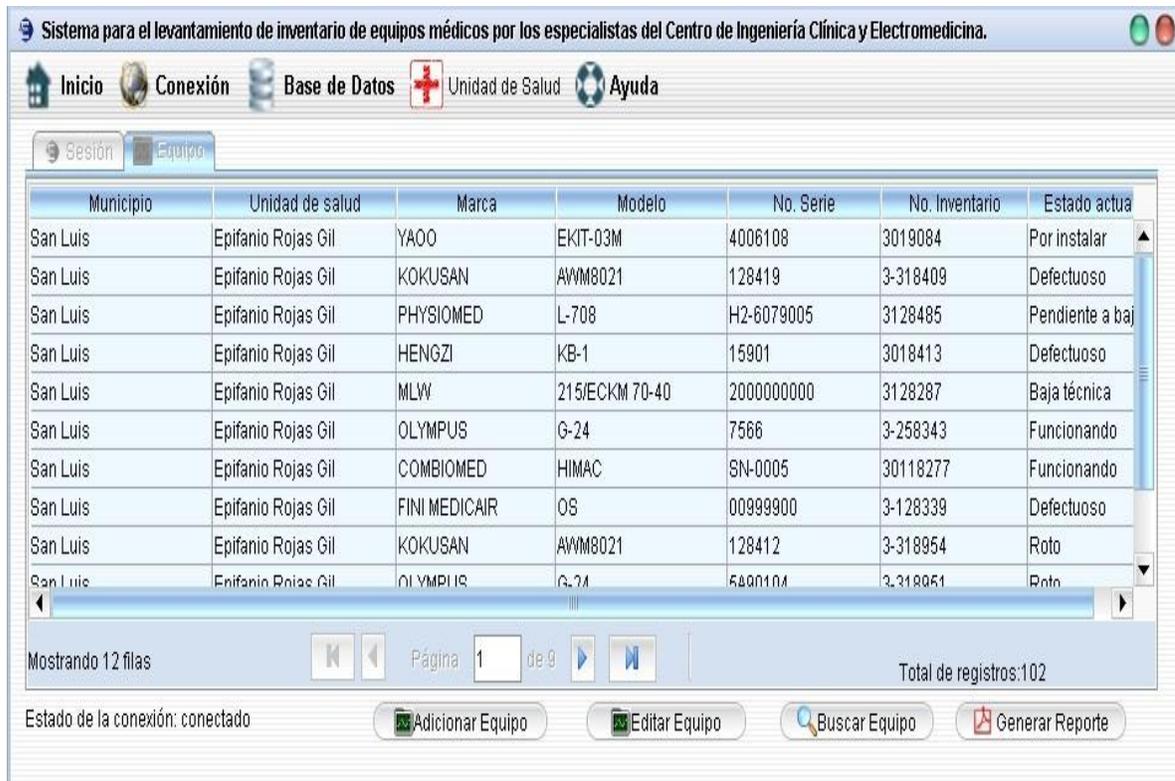


Figura # 25: Interfaz de usuario Inicio.

Es necesario seleccionar una o varias unidades de salud, antes de actualizar en la aplicación, los equipos médicos de estas y los nomencladores necesarios para el correcto funcionamiento de la misma.

País: <<seleccionar>>

Municipio: <<seleccione>>

Region: <<seleccione>>

Localidad: <<seleccione>>

Provincia: <<seleccione>>

Unidad Salud: <<seleccione>>

Nombre_U_Salud	Dirección
Óptica ELAM No 12	Sin Direccionfdd
Optica Sandino	Sin Direccion
U622 Farmacia Principal	Sin Direccion
Farmacia U681 San Luis	Sin Direccion
Epifanio Rojas Gil	Sin Direccion

Figura # 26: Interfaz de usuario Seleccionar unidad de salud.

Posteriormente el especialista pudo insertar un nuevo equipo médico al seleccionar la opción Adicionar equipo de la interfaz Inicio. Dando lugar a una nueva interfaz donde se insertan los datos de los equipos.

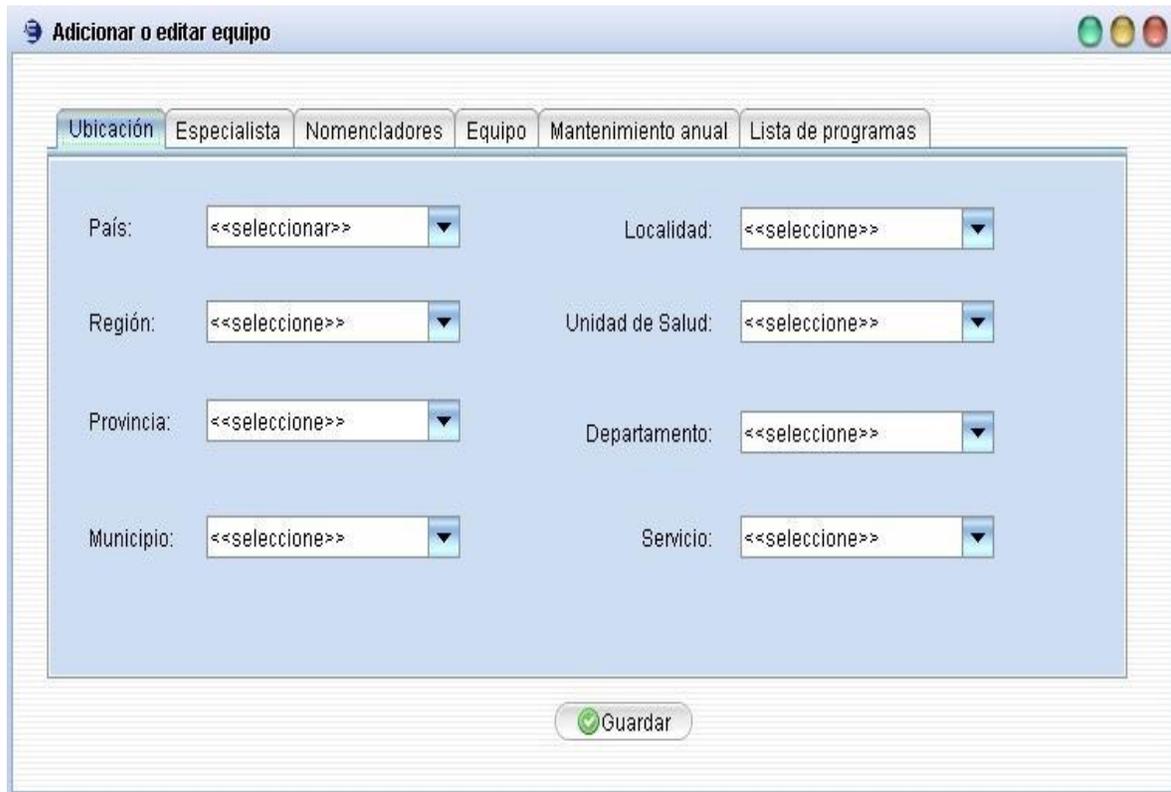


Figura # 27: Interfaz de usuario Adicionar o modificar equipo.

Luego de realizado el inventario se debe actualizar el mismo en la base de datos del SIGICEM al seleccionar, de la interfaz Inicio, la opción *Actualizar datos en el servidor* del menú *Base de Datos*, garantizándose el control de todos los equipos médicos del SNS.

Durante la confección del actual capítulo se abordaron cuestiones pertenecientes a la etapa de implementación. Obteniéndose el diagrama de componentes, donde se detallan las organizaciones y dependencias lógicas de cada uno de los componentes. Además se obtuvo la distribución física que se debe seguir para el despliegue del sistema para el levantamiento de inventario de equipos médicos, especificando cada uno de sus elementos. Con la utilización de los estándares y estilos de codificación para la implementación de los componentes definidos, se logró implementar un sistema que diera solución a la presente investigación. También se mostró una breve explicación del sistema propuesto, lográndose un mejor entendimiento del mismo.

CONCLUSIONES

Con la presente investigación se le dio cumplimiento al objetivo general propuesto, arribando a las siguientes conclusiones:

- El análisis de los procesos asociados a la gestión de inventarios en el Centro de Ingeniería Clínica y Electromedicina, posibilitó obtener una guía para el posterior desarrollo de la solución propuesta.
- La realización de un estudio previo de sistemas de gestión de inventario existentes a nivel internacional y nacional resaltó la necesidad de llevar a cabo el desarrollo del Sistema para el levantamiento de inventario de equipos médicos por los especialistas del Centro de Ingeniería Clínica y Electromedicina.
- Fueron asimiladas y aplicadas las herramientas, tecnologías y arquitectura definidas por el proyecto para el desarrollo de soluciones de escritorio, permitiendo el desarrollo de la propuesta de solución.
- Mediante la confección de los artefactos pertenecientes a las disciplinas de trabajo definidas y guiados por el Proceso Unificado de Desarrollo, fueron descritos cada uno de sus elementos, logrando establecer la documentación necesaria de la investigación.
- La selección de ficheros serializados para el almacenamiento de los datos garantiza la portabilidad del sistema con requerimientos mínimos.
- Se obtuvo la versión funcional del sistema para el levantamiento de inventario de equipos médicos, permitiendo así, llevar un control de los equipos médicos de las unidades de salud que no poseen conectividad con el Sistema de Gestión para Ingeniería Clínica y Electromedicina.

RECOMENDACIÓN

Realizar una nueva versión del sistema para el levantamiento de inventario de equipos médicos por los especialistas del Centro de Ingeniería Clínica y Electromedicina, que pueda ser usada en dispositivos móviles y PDA.

REFERENCIAS BIBLIOGRÁFICAS

1. ABCdatos. [En línea] 17 de 07 de 2009. [Citado el: 25 de 11 de 2012.] <http://www.abcdatos.com/programa/inventarios-gestion.html>.
2. Programas-gratis.net. [En línea] Mediaprogramas , 2013. [Citado el: 2013 de enero de 28.] <http://ok-inventario.programas-gratis.net/>.
3. satnetwork Sistemas Informaticos. [En línea] 2011. [Citado el: 6 de 12 de 2012.] <http://www.satnetwork.com.ar>.
4. Rodas XXI Sistema Integral Económico Administrativo. [En línea] 2002. <http://www.rodasxxi.cu/inventario.php>.
5. DISAIC Casa Consultora. [En línea] 2008. [Citado el: 4 de 12 de 2012.] <http://www.disaic.cu/modules.php?name=content&pa=showpage&pid=818>.
6. SISCONT. (s.a). Tu software contable financiero. [En línea] [Citado el: 14 de 11 de 2012.] <http://www.siscont.com/DESCRIPTIVO%20SISCONT.pdf>.
7. **Miniet, Y. V.** *Definición del ciclo de vida del proyecto ERP*. Habana : s.n., 2009.
8. **Jacobson, Ivar, Rumbaugh, James y Booch, Grady.** *El Proceso Unificado de Desarrollo de Software*. Madrid : Pearson Educacion, 2000. 120.
9. **James Rumbaugh, IvarJacobson y Grady Booch.** *El Lenguaje Unificado de Modelado. Manual de Referencia*. Madrid : Addison Weasley, 2000.
10. Enterate en línea. [En línea] 27 de 11 de 2008. [Citado el: 8 de 12 de 2012.] <http://www.enterate.unam.mx/Articulos/2003/junio/j2ee.htm>.
11. Oracle Corporation. Oracle Technology Network. [En línea] 2011. [Citado el: 8 de 12 de 2012.] <http://download.oracle.com/javase/6/docs/>.

12. **Vélez Serrano, José F, y otros.** *Diseñar y programar, todo es empezar: Una introducción a la Programación Orientada a Objetos usando UML y Java.* Madrid : DYKINSON, S.L. Méndez Valdés, 61 - 28015 Madrid, 2011. pág. 128. 978-84-9982-345-4.
13. **García Peñalvo, Francisco José.** zarza.usa. [En línea] 2008. [Citado el: 5 de 12 de 2012.] http://zarza.usal.es/~fgarcia/doc/tuto2/IV_3.htm.
14. **Lowagie, Bruno.** *iText in Action.* New York : Manning Publications Co, 2007. 1932394796.
15. Oracle Corporation. netbeans.org. [En línea] 2012 de 12 de 5. http://netbeans.org/index_es.html.
16. Canonical Ltd. ubuntu-es. [En línea] ubuntu-es, 4 de 6 de 2011. http://doc.ubuntu-es.org/Sobre_Ubuntu.
17. **Agut, Raúl Monferrer.** *Especificación de Requisitos Software según el estándar de IEEE 830.* Departament d'Informàtica, Universitat Jaume I. 2000-2001.
18. **L. Bass, P. Clements, R. Kazman.** *Software Architecture in Practice.* 2nd Edition. s.l. : Addison Wesley, 2003.
19. ComuSOFT. [En línea] 2011. [Citado el: 22 de abril de 2013.] <http://www.comusoft.com/modelo-vista-controlador-definicion-y-caracteristicas>.
20. **Acuña, Kareny Brito.** Selección de metodologías de desarrollo para aplicaciones web en la facultad de Informática en la Universidad de Cienfuegos. [En línea] [Citado el: 5 de abril de 2013.] <http://www.eumed.net/libros-gratis/2009c/584/RUP%20Diseno%20e%20implementacion%20del%20sistema.htm>.
21. SlideShare. *Seguridad Informática. La información como activo estratégico.* [En línea] 2013. [Citado el: 28 de mayo de 2013.] <http://www.slideshare.net/jemarinoui/seguridad-informtica-1125964>.
22. **Calleja, Manuel Arias.** Carmen. Estándares de codificación. [En línea] [Citado el: 10 de abril de 2013.] <http://www.cisiad.uned.es/carmen/estilo-codificacion.pdf>.

23. **Lucas Vieites.** Convenciones de código para el lenguaje de programación Java. [En línea] mayo de 2007. [Citado el: 17 de abril de 2013.] http://blog.codexion.com/wp-content/uploads/2010/02/Convenciones_Codigo_Java.pdf.

BIBLIOGRAFÍA

ABCdatos. [En línea] 17 de 07 de 2009. [Citado el: 25 de 11 de 2012.] <http://www.abcdatos.com/programa/inventarios-gestion.html>.

Acuña, Kareenny Brito. Selección de metodologías de desarrollo para aplicaciones web en la facultad de Informática en la Universidad de Cienfuegos. [En línea] [Citado el: 5 de abril de 2013.] <http://www.eumed.net/libros-gratis/2009c/584/RUP%20Diseno%20e%20implementacion%20del%20sistema.htm>.

Agut, Raúl Monferrer. Especificación de Requisitos Software según el estándar de IEEE 830. Departament d'Informàtica, Universitat Jaume I. 2000-2001.

ALARCÓN, RAÚL. *DISEÑO ORIENTADO A OBJETOS CON UML*. Madrid : Grupo EIDOS, 2000.

Aspectos Generales del Sistema Nacional de Salud. [En línea] [Citado el: 18 de octubre de 2012.] http://www.sld.cu/sistema_de_salud/aspectos.html.

Calleja, Manuel Arias. Carmen. Estándares de codificación. [En línea] [Citado el: 10 de abril de 2013.] <http://www.cisiad.uned.es/carmen/estilo-codificacion.pdf>.

Canonical Ltd. ubuntu-es. [En línea] ubuntu-es, 4 de 6 de 2011. http://doc.ubuntu-es.org/Sobre_Ubuntu.

ComuSOFT. [En línea] 2011. [Citado el: 22 de abril de 2013.] <http://www.comusoft.com/modelo-vista-controlador-definicion-y-caracteristicas>.

Díaz, Vicente Prieto, y otros. Impacto de las tecnologías de la información y las comunicaciones en la educación y nuevos paradigmas del enfoque educativo. [En línea] [Citado el: 8 de octubre de 2012.] http://bvs.sld.cu/revistas/ems/vol25_1_11/ems09111.htm.

Díaz, Vicente Prieto, y otros. Impacto de las tecnologías de la información y las comunicaciones en la educación y nuevos paradigmas del enfoque educativo. [En línea] [Citado el: 8 de octubre de 2012.] http://bvs.sld.cu/revistas/ems/vol25_1_11/ems09111.htm.

DISAIC Casa Consultora. [En línea] 2008. [Citado el: 4 de 12 de 2012.] <http://www.disaic.cu/modules.php?name=content&pa=showpage&pid=818>.

Enterate en línea. [En línea] 27 de 11 de 2008. [Citado el: 8 de 12 de 2012.] <http://www.enterate.unam.mx/Articulos/2003/junio/j2ee.htm>.

ERIKA CAMACHO, FABIO CARDESO y GABRIEL NUÑEZ. ARQUITECTURAS DE SOFTWARE. Guía de estudio. [En línea] 2004. [Citado el: 10 de 4 de 2013.] <http://prof.usb.ve/lmendoza/Documentos/PS-6116/Guia%20Arquitectura%20v.2.pdf>.

Gamma, Erich, Helm, Richard, Johnson, Ralph and Vlissides, John. Design Patterns. Elements of Reusable Object. s.l. : Addison-Wesley, 1995.

García Peñalvo, Francisco José. zarza.usa. [En línea] 2008. [Citado el: 5 de 12 de 2012.] http://zarza.usal.es/~fgarcia/doc/tuto2/IV_3.htm.

García, Olga. El impacto de las TIC en el mundo de hoy | GestioPolis. [En línea] 26 de enero de 2011. [Citado el: 2012 de octubre de 2012.] <http://www.gestiopolis.com/administracion-estrategia-2/impacto-de-las-tic-en-el-mundo-de-hoy.htm>.

González, Rolando Alfredo Hernández León y Sayda Coello. El Proceso de Investigación Científica. Hernández León, Rolando Alfredo. Ciudad de La Habana : Editorial Universitaria, 2011.

Indira Lilled Laurencio Fuentes, Oscar Naranjo González. Modelado de una aplicación web para el módulo de inventario del sistema Condor. La Habana : s.n., 2007.

Jacobson, Ivar, Booch, Grady y Rumbaugh, James. El Proceso Unificado de Desarrollo de Software. España : Addison Wesley.

James Rumbaugh, IvarJacobson y Grady Booch. El Lenguaje Unificado de Modelado. Manual de Referencia. Madrid : Addison Weasley, 2000.

L. Bass, P. Clements, R. Kazman. Software Architecture in Practice. 2nd Edition. s.l. : Addison Wesley, 2003.

L'IV Com Sàrl, Villars-sous-Yens. Introducción a la gestión de inventarios de equipo médico. Suiza : Organización Mundial de Salud, febrero 2012.

Las tecnologías de información y comunicación y la gestión del conocimiento en el sector salud*. Ruiz, Susana Beatriz Llanusa. 3, Ciudad de la Habana : Ciencias Médicas, 2005, Vol. 31.

Lowagie, Bruno. iText in Action. New York : Manning Publications Co, 2007. 1932394796.

Mastermagazine. [En línea] [Citado el: 29 de noviembre de 2012.] <http://www.mastermagazine.info/termino/4394.php>.

Miniet, Y. V. Definición del ciclo de vida del proyecto ERP . Habana : s.n., 2009.

Oracle Corporation. netbeans.org. [En línea] 2012 de 12 de 5. http://netbeans.org/index_es.html.

Oracle Corporation. Oracle Technology Network. [En línea] 2011. [Citado el: 8 de 12 de 2012.] <http://download.oracle.com/javase/6/docs/>.

Orallo, Enrique Hernández. *El Lenguaje Unificado de Modelado (UML)*.

Pressman, Roger. ***Ingeniería del Software: Un enfoque práctico***. Quinta Edición. Madrid, 2002.

Programas-gratis.net. [En línea] Mediaprogramas , 2013. [Citado el: 2013 de enero de 28.] <http://ok-inventario.programas-gratis.net/>.

Programas-gratis.net. [En línea] Mediaprogramas , 2013. [Citado el: 2013 de enero de 28.] <http://ok-inventario.programas-gratis.net/>.

Rodas XXI Sistema Integral Económico Administrativo. [En línea] 2002. <http://www.rodasxxi.cu/inventario.php>.

satnetwork Sistemas Informaticos. [En línea] 2011. [Citado el: 6 de 12 de 2012.] <http://www.satnetwork.com.ar>.

SISCONT. (s.a). Tu software contable financiero. [En línea] [Citado el: 14 de 11 de 2012.] <http://www.siscont.com/DESCRIPTIVO%20SISCONT.pdf>.

Sistema de gestión para electromedicina(SIGEM). [En línea] 2006. [Citado el: 1 de noviembre de 2012.] <http://www.hab2007.sld.cu/Members/felix/sistema-de-gestion-para-electromedicina/>.

SlideShare. Seguridad Informática. La información como activo estratégico. [En línea] 2013. [Citado el: 28 de mayo de 2013.] <http://www.slideshare.net/jemarinoi/seguridad-informtica-1125964>.

Vélez Serrano, José F, y otros. Diseñar y programar, todo es empezar: Una introducción a la Programación Orientada a Objetos usando UML y Java. Madrid : DYKINSON, S.L. Méndez Valdés, 61 - 28015 Madrid, 2011. pág. 128. 978-84-9982-345-4.

Visual Paradigm. [En línea] [Citado el: 5 de 12 de 2012.] http://www.visual-paradigm.com/support/documents/vpumluserguide/12/13/5963_aboutvisualp.htm.

GLOSARIO DE TÉRMINOS

Conectividad: capacidad de dos o más elementos, hardware o software, para trabajar conjuntamente y transmitirse datos e información en un entorno informático heterogéneo.

Validación de los requisitos: Proceso de confirmación, por parte de los usuarios o del cliente, de que los requisitos especificados son válidos, consistentes, completos.

Verificación de los requisitos: Proceso de comprobación de que los requisitos realmente cubren las necesidades del cliente.

Especificación de requisitos: proceso de redacción o registro de los requisitos. Se pueden utilizar tanto el lenguaje natural, como modelos gráficos.

Requisitos: Descripción de las necesidades o deseos de un producto.

UML: (Unified Modeling Language o Lenguaje Unificado de Modelado). Lenguaje de programación gráfico para el modelado de proyectos de software orientados a objetos.

Software: es el conjunto de programas, procedimientos y documentación asociada a la operación de un sistema informático.

CICEM: Centro de Ingeniería Clínica y Electromedicina.

Proyecto software: conjunto de actividades coordinadas cronológicamente para alcanzar un subconjunto de objetivos a partir de la definición de un subconjunto de necesidades.

IDE: (Integrated Development Environment o Entorno de Desarrollo Integrado). Conjunto de herramientas para el programador, que suelen incluir en una misma suite, un buen editor de código, administrador de proyectos y archivos, enlace transparente a compiladores y debuggers, e integración con sistemas controladores de versiones o repositorios.

CICEM: Centro de Ingeniería Clínica y Electromedicina.

CESIM: Centro de Informática Médica.

Electromedicina: la rama de la ciencia que se encarga del desarrollo, aplicación, mantenimiento y gestión de los equipos y accesorios médicos. .

Herramientas CASE: Conjunto de aplicaciones informáticas orientadas al incremento de la productividad en el desarrollo de software, las siglas CASE vienen dadas por su nombre en inglés Computer Aided Software Engineering que se conoce como Ingeniería de Software Asistida por Computadoras.

Look and feel: aspecto que muestran las interfaces visuales desarrolladas en java.

JAVA: lenguaje de programación orientado a objetos.

JDK (Java Development Kit): conjunto de herramientas con que cuenta el lenguaje de programación JAVA para el desarrollo de aplicaciones.

JRE (Java Runtime Environment): es una máquina virtual de Java y su función es hacer de intermediario entre una aplicación programada en Java y el sistema operativo que se encuentre usando. De este modo, cualquier aplicación puede funcionar en cualquier sistema operativo que disponga del JRE.

TCP/IP (Transmission Control IP/Internet Protocol): es el conjunto de protocolos que se utilizan para transportar la información en Internet y en la mayoría de redes privadas.

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

SAS: Sistemas de Apoyo a la Salud perteneciente al CESIM.

Software Libre: es la denominación del software que respeta la libertad de los usuarios sobre su producto adquirido y, por tanto, una vez obtenido puede ser usado, copiado, estudiado, modificado, y redistribuido libremente.

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

SNS: Sistema Nacional de Salud. TIC: Tecnologías de la Información y las Comunicaciones.

Modelo del dominio: es el conjunto de clases que un ingeniero de software modela al analizar el problema que desea resolver.

Modelo de aplicación: es un conjunto de clases que se relacionan con el modelo del dominio, que tienen conocimiento de las vistas y que implementan los mecanismos necesarios para notificar a estas últimas sobre los cambios que pudiesen ocurrir en el modelo del dominio.

Inventario: es una relación detallada de los activos que posee una organización o institución

Tecnología médica: conjunto de medicamentos, aparatos, procedimientos médicos y quirúrgicos utilizados en la atención médica y los sistemas organizativos con los que se presta la atención sanitaria

Unidad de salud: institución donde se brinda atención médica a las pacientes.